LNCS 3420

Pascal Lorenz
Petre Dini (Eds.)

Networking – ICN 2005

4th International Conference on Networking
Reunion Island, France, April 2005
Proceedings, Part I

1 Part I

Springer

Lecture Notes in Computer Science 3420

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Pascal Lorenz   Petre Dini (Eds.)

Networking – ICN 2005

4th International Conference on Networking
Reunion Island, France, April 17-21, 2005
Proceedings, Part I

Springer

Volume Editors

Pascal Lorenz
University of Haute Alsace
34 rue du Grillenbreit, 68008 Colmar, France
E-mail: lorenz@ieee.org

Petre Dini
Cisco Systems, Inc.
170 West Tasman Drive, San Jose, CA 95134, USA
E-mail: pdini@cisco.com

Library of Congress Control Number: 2005922556

CR Subject Classification (1998): C.2, K.4.4, H.4.3, H.5.1, H.3, K.6.4-5

ISSN 0302-9743
ISBN 3-540-25339-4 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springeronline.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11407065 06/3142 5 4 3 2 1 0

Preface

The International Conference on Networking (ICN 2005) was the fourth conference in its series aimed at stimulating technical exchange in the emerging and important field of networking. On behalf of the International Advisory Committee, it is our great pleasure to welcome you to the proceedings of the 2005 event.

Networking faces dramatic changes due to the customer-centric view, the venue of the next generation networks paradigm, the push from ubiquitous networking, and the new service models. Despite legacy problems, which researchers and industry are still discovering and improving the state of the art, the horizon has revealed new challenges that some of the authors tackled through their submissions.

In fact ICN 2005 was very well perceived by the international networking community. A total of 651 papers from more than 60 countries were submitted, from which 238 were accepted. Each paper was reviewed by several members of the Technical Program Committee. This year, the Advisory Committee revalidated various accepted papers after the reviews had been incorporated. We perceived a significant improvement in the number of submissions and the quality of the submissions.

The ICN 2005 program covered a variety of research topics that are of current interest, starting with Grid networks, multicasting, TCP optimizations, QoS and security, emergency services, and network resiliency. The Program Committee selected also three tutorials and invited speakers that addressed the latest research results from the international industries and academia, and reports on findings from mobile, satellite, and personal communications related to 3rd- and 4th-generation research projects and standardization.

This year we enriched ICN with a series of papers targeting emergency services and disaster recovery (the AICED section); this emerging topic hopefully will lead to more robust and fault-tolerant systems for preventing technical and human disasters.

We would like to thank the International Advisory Committee members and the referees. Without their support, the program organization of this conference would not have been possible. We are also indebted to many individuals and organizations that made this conference possible (Cisco Systems, Inc., France Telecom, IEEE, IARIA, Region Reunion, University of La Reunion, ARP). In particular, we thank the members of the Organizing Committee for their help in all aspects of the organization of this conference.

We hope that the attendees enjoyed this International Conference on Networking on Reunion Island, and found it a useful forum for the exchange of ideas and results and recent findings. We also hope that the attendees found time to enjoy the island's beautiful countryside and its major cultural attractions.

April 2005

Pascal Lorenz
Petre Dini

International Scientific Committee

Advisory Committee Board

P. Dini (USA) — Cisco Systems, Inc.
P. Lorenz (France) — University of Haute Alsace
G. Parr (UK) — University of Ulster (for the AICED section on emergency services and disaster recovery)

Tutorial Chairs

A. Jamalipour (Australia) — University of Sydney
M. Hur (USA) — Microsoft (for the AICED section on emergency services and disaster recovery)

Program Chairs

M. Freire (Portugal) — University of Beira Interior
H. Debar (France) — France Telecom R&D (for the AICED section on emergency services and disaster recovery)

International Advisory Committee

H. Adeli (USA) — Ohio State University
K. Al-Begain (UK) — University of Glamorgan
P. Anelli (France) — University of La Reunion
E. Barnhart (USA) — Georgia Institute of Technology
J. Ben Othman (France) — University of Versailles
B. Bing (USA) — Georgia Institute of Technology
D. Bonyuet (USA) — Delta Search Labs
J. Brito (Brazil) — INATEL
M. Carli (Italy) — University of Rome TRE
P. Chemouil (France) — France Telecom R&D
M. Devetsikiotis (USA) — North Carolina State University
P. Dini (USA) — Cisco Systems, Inc.
P. Dommel (USA) — Santa Clara University
Y. Donoso (Colombia) — University del Norte
I. Elhanany (USA) — University of Tennessee

A. Finger (Germany) — Dresden University of Technology
M. Freire (Portugal) — University of Beira Interior
E. Fulp (USA) — Wake Forest University
B. Gavish (USA) — Southern Methodist University
F. Granelli (Italy) — University of Trento
H. Guyennet (France) — University of Franche-Comté
Z. Hulicki (Poland) — AGH University of Science and Technology
A. Jamalipour (Australia) — University of Sydney
A.L. Jesus Teixeira (Portugal) — University of Aveiro
D. Khotimsky (USA) — Invento Networks
R. Komiya (Malaysia) — Faculty of Information Technology
S. Kumar (USA) — University of Texas
J.C. Lapeyre (France) — University of Franche-Comte
P. Lorenz (France) — University of Haute Alsace
M.S. Obaida (USA) — Monmouth University
A. Pescape (Italy) — University of Napoli "Federico II"
D. Magoni (France) — University of Strasbourg
Z. Mammeri (France) — University of Toulouse
A. Molinaro (Italy) — University of Calabria
H. Morikawa (Japan) — University of Tokyo
H. Mouftah (Canada) — University of Ottawa
A. Pandharipande (Korea) — Samsung Advanced Institute of Technology
S. Recker (Germany) — IMST GmbH
F. Ricci (USA) — The Catholic University of America
J. Rodrigues (Portugal) — University of Beira Interior
P. Rolin (France) — France Telecom R&D
B. Sarikaya (Canada) — UNBC
J. Soler-Lucas (Denmark) — Research Center COM
S. Soulhi (Canada) — Ericsson, Inc.
V. Uskov (USA) — Bradley University
R. Valadas (Portugal) — University of Aveiro
L. Vasiu (UK) - Wireless IT Research Centre
E. Vazquez (Spain) — Technical University of Madrid
D. Vergados (Greece) — University of the Aegean
S. Yazbeck (USA) — Barry University
V. Zaborovski (Russia) — Saint-Petersburg Politechnical University
A. Zaslavsky (Australia) — Monash University
H.J. Zepernick (Australia) — Western Australian Telecommunications Research Institute

International Committee for the topics related to the emergency services and disaster recovery

M. Barbeau (Canada) — Carleton University
G. Candea (USA) — Stanford University

H. Debar (France) — France Telecom R&D
P. Dini (USA) — Cisco Systems, Inc.
S. Gjessing (Norway) — Simula Research Laboratory
P.-H. Ho (Canada) — University of Waterloo
M. Hur (USA) — Microsoft
N. Kapadia (USA) — Capital One
P. Lorenz (France) — University of Haute Alsace
D. Malagrino (USA) — Cisco Systems, Inc.
M. Moh (USA) — San Jose State University
G. Parr (UK) — University of Ulster
A. Pescapé (Italy) — Università degli Studi di Napoli "Frederico II"
H. Reiser (Germany) — Ludwig Maximilians University Munich
J. Salowey (USA) — Cisco Systems, Inc.
D. Schupke (Germany) — Siemens, AG
F. Serr (USA) — Cisco Systems, Inc.
C. Becker Westphall (Brazil) — Federal University of Santa Catarina
A. Zaslavsky (Australia) — Monash University

Table of Contents – Part I

GRID

Mobile-to-Grid Middleware: An Approach for Breaching the Divide Between Mobile and Grid Environments
Umar Kalim, Hassan Jameel, Ali Sajjad, Sungyoung Lee 1

On the Influence of Network Characteristics on Application Performance in the Grid Environment
Yoshinori Kitatsuji, Satoshi Katsuno, Katsuyuki Yamazaki, Hiroshi Koide, Masato Tsuru, Yuji Oie 9

A Client-Side Workflow Middleware in the Grid
Ying Li, Qiaoming Zhu, Minglu Li, Yue Chen 19

General Architecture of Grid Framework with QoS Implementation
Vit Vrba, Karol Molnar, Lubomir Cvrk 27

Optical Networks (I)

Centralized Versus Distributed Re-provisioning in Optical Mesh Networks
Chadi Assi, Wei Huo, Abdallah Shami 34

The Role of Meshing Degree in Optical Burst Switching Networks Using Signaling Protocols with One-Way Reservation Schemes
Joel J.P.C. Rodrigues, Mário M. Freire, Pascal Lorenz 44

Analytical Model for Cross-Phase Modulation in Multi-span WDM Systems with Arbitrary Modulation Formats
Gernot Göger, Bernhard Spinnler 52

Low-Cost Design Approach to WDM Mesh Networks
Cristiana Gomes, Geraldo Robson Mateus 60

A New Path Protection Algorithm for Meshed Survivable Wavelength-Division-Multiplexing Networks
Lei Guo, Hongfang Yu, Lemin Li 68

Wireless Networks (I)

Application Area Expansion in Quasi-Millimeter Wave Band Fixed Wireless Access System
Shuta Uwano, Ryutaro Ohmoto 76

A Robust Service for Delay Sensitive Applications on a WLAN
Fanilo Harivelo, Pascal Anelli 84

17 GHz Wireless LAN: Performance Analysis of ARQ Based Error Control Schemes
Giuseppe Razzano, Luca Cecconi, Roberto Cusani 92

Performance Analysis of Mac-hs Protocol
Robert Bestak 100

Distributed k-Clustering Algorithms for Random Wireless Multihop Networks
Vlady Ravelomanana 109

QoS (I)

Call Admission Control with SLA Negotiation in QoS-Enabled Networks
Iftekhar Ahmad, Joarder Kamruzzaman, Srinivas Aswathanarayaniah 117

Enhancing QoS Through Alternate Path: An End-to-End Framework
Thierry Rakotoarivelo, Patrick Senac, Aruna Seneviratne, Michel Diaz 125

A Comparison on Bandwidth Requirements of Path Protection Mechanisms
Claus G. Gruber 133

Quality of Service Solutions in Satellite Communication
Mathieu Gineste, Patrick Sénac 144

QoS-Oriented Packet Scheduling Schemes for Multimedia Traffics in OFDMA Systems
Seokjoo Shin, Seungjae Bahng, Insoo Koo, Kiseon Kim 153

Optical Networks (II)

Packet Delay Analysis of Dynamic Bandwidth Allocation Scheme in an Ethernet PON
Chul Geun Park, Dong Hwan Han, Bara Kim 161

Inter-domain Advance Resource Reservation for Slotted Optical Networks
Abdelilah Maach, Abdelhakim Hafid, Jawad Drissi 169

Virtual Source-Based Minimum Interference Path Multicast Routing with Differentiated QoS Guarantees in the Next Generation Optical Internet
Suk-Jin Lee, Kyung-Dong Hong, Chun-Jai Lee, Moon-Kyun Oh, Young-Bu Kim, Jae-Dong Lee, Sung-Un Kim 178

Multiple Failures Restoration by Group Protection in WDM Networks
Chen-Shie Ho, Ing-Yi Chen, Sy-Yen Kuo 186

Wavelength Assignment in Route-Fixed Optical WDM Ring by a Branch-and-Price Algorithm
Heesang Lee, Yun Bae Kim, Seung J. Noh, Sun Hur 194

Wireless Networks (II)

M-MIP: Extended Mobile IP to Maintain Multiple Connections to Overlapping Wireless Access Networks
Christer Åhlund, Robert Brännström, Arkady Zaslavsky 204

Light-Weight WLAN Extension for Predictive Handover in Mobile IPv6
Soohong Park, Pyung Soo Kim 214

Algorithms for Energy-Efficient Broad- and Multi-casting in Wireless Networks
Hiroshi Masuyama, Kazuya Murakami, Toshihiko Sasama 221

Converting SIRCIM Indoor Channel Model into SNR-Based Channel Model
Xiaolei Shi, Mario Hernan Castaneda Garcia, Guido Stromberg 231

CAWAnalyser: Enhancing Wireless Intrusion Response with Runtime Context-Awareness
Choon Hean Gan, Arkady Zaslavsky, Stephen Giles 239

Evaluation of Transport Layer Loss Notification in Wireless Environments
Johan Garcia, Anna Brunstrom ... 247

End-to-End Wireless Performance Simulator: Modeling Methodology and Performance
Sung-Min Oh, Hyun-Jin Lee, Jae-Hyun Kim ... 258

QoS (II)

Client-Controlled QoS Management in Networked Virtual Environments
Patrick Monsieurs, Maarten Wijnants, Wim Lamotte ... 268

UML-Based Approach for Network QoS Specification
Cédric Teyssié, Zoubir Mammeri ... 277

Modeling User-Perceived QoS in Hybrid Broadcast and Telecommunication Networks
Michael Galetzka, Günter Elst, Adolf Finger ... 286

Holistic and Trajectory Approaches for Distributed Non-preemptive FP/DP* Scheduling
Steven Martin, Pascale Minet ... 296

Evaluating Evolutionary IP-Based Transport Services on a Dark Fiber Large-Scale Network Testbed
Francesco Palmieri ... 306

Pareto Optimal Based Partition Framework for Two Additive Constrained Path Selection
Yanxing Zheng, Turgay Korkmaz, Wenhua Dou ... 318

Optical Networks (III)

Joint Path Protection Scheme with Efficient RWA Algorithm in the Next Generation Internet Based on DWDM
Jin-Ho Hwang, Jae-Dong Lee, Jun-Won Lee, Sung-Un Kim ... 326

On Integrated QoS Control in IP/WDM Networks
Wei Wei, Zhongheng Ji, Junjie Yang, Qingji Zeng ... 334

Optical Hybrid Switching Using Flow-Level Service Classification for IP Differentiated Service
Gyu Myoung Lee, Jun Kyun Choi ... 342

Delay Constraint Dynamic Bandwidth Allocation for Differentiated Service in Ethernet Passive Optical Networks
Lin Zhang, Lei Li, Huimin Zhang 350

Wireless Networks (III)

An Architecture for Efficient QoS Support in the IEEE 802.16 Broadband Wireless Access Network
Dong-Hoon Cho, Jung-Hoon Song, Min-Su Kim, Ki-Jun Han 358

A Pragmatic Methodology to Design 4G: From the User to the Technology
Simone Frattasi, Hanane Fathi, Frank Fitzek, Marcos Katz, Ramjee Prasad 366

Integrating WMAN with WWAN for Seamless Services
Jinsung Cho, Dae-Young Kim 374

Towards Mobile Broadband
J. Charles Francis, Johannes Schneider 382

Emulation Based Performance Investigation of FTP File Downloads over UMTS Dedicated Channels
Oumer M. Teyeb, Malek Boussif, Troels B. Sørensen, Jeroen Wigard, Preben E. Mogensen 388

Uni-source and Multi-source m-Ary Tree Algorithms for Best Effort Service in Wireless MAN
Jin Kyung Park, Woo Cheol Shin, Jun Ha, Cheon Won Choi 397

WPAN

High Rate UWB-LDPC Code and Its Soft Initialization
Jia Hou, Moon Ho Lee 406

Cube Connected Cycles Based Bluetooth Scatternet Formation
Marcin Bienkowski, André Brinkmann, Miroslaw Korzeniowski, Orhan Orhan 413

Design of UWB Transmitter and a New Multiple-Access Method for Home Network Environment in UWB Systems
Byung-Lok Cho, Young-Kyu Ahn, Seok-Hoon Hong, Mike Myung-Ok Lee, Hui-Myung Oh, Kwan-Ho Kim, Sarm-Goo Cho 421

Bluetooth Device Manager Connecting a Large Number of Resource-Constraint Devices in a Service-Oriented Bluetooth Network
Hendrik Bohn, Andreas Bobek, Frank Golatowski 430

Sensor Networks (I)

ESCORT: Energy-Efficient Sensor Network Communal Routing Topology Using Signal Quality Metrics
Joel W. Branch, Gilbert G. Chen, Boleslaw K. Szymanski 438

On the Security of Cluster-Based Communication Protocols for Wireless Sensor Networks
Adrian Carlos Ferreira, Marcos Aurélio Vilaça, Leonardo B. Oliveira, Eduardo Habib, Hao Chi Wong, Antonio A. Loureiro 449

An Energy-Efficient Coverage Maintenance Scheme for Distributed Sensor Networks
Min-Su Kim, Taeyoung Byun, Jung-Pil Ryu, Sungho Hwang, Ki-Jun Han 459

A Cluster-Based Energy Balancing Scheme in Heterogeneous Wireless Sensor Networks
Jing Ai, Damla Turgut, Ladislau Bölöni 467

An Optimal Node Scheduling for Flat Wireless Sensor Networks
Fabíola Guerra Nakamura, Frederico Paiva Quintão, Gustavo Campos Menezes, Geraldo Robson Mateus 475

Traffic Control (I)

A Congestion Control Scheme Based on the Periodic Buffer Information in Multiple Beam Satellite Networks
Seungcheon Kim 483

Real-Time Network Traffic Prediction Based on a Multiscale Decomposition
Guoqiang Mao 492

Provisioning VPN over Shared Network Infrastructure
Quanshi Xia 500

Potential Risks of Deploying Large Scale Overlay Networks
Maoke Chen, Xing Li 508

Utility-Based Buffer Management for Networks
Cedric Angelo M. Festin, Søren-Aksel Sørensen 518

Communication Architectures

Design and Implementation of a Multifunction, Modular and Extensible Proxy Server
Simone Tellini, Renzo Davoli 527

Pass Down Class-LRU Caching Algorithm for WWW Proxies
Rachid El Abdouni Khayari 535

Delay Estimation Method for N-tier Architecture
Shinji Kikuchi, Ken Yokoyama, Akira Takeyama 544

A New Price Mechanism Inducing Peers to Achieve Optimal Welfare
Ke Zhu, Pei-dong Zhu, Xi-cheng Lu 554

Sensor Networks (II)

A Study of Reconnecting the Partitioned Wireless Sensor Networks
Qing Ye, Liang Cheng 561

Application-Driven Node Management in Multihop Wireless Sensor Networks
Flávia Delicato, Fabio Protti, José Ferreira de Rezende, Luiz Rust, Luci Pirmez 569

Power Management Protocols for Regular Wireless Sensor Networks
Chih-Pin Liao, Jang-Ping Sheu, Chih-Shun Hsu 577

Information Fusion for Data Dissemination in Self-Organizing Wireless Sensor Networks
Eduardo Freire Nakamura, Carlos Mauricio S. Figueiredo, Antonio Alfredo F. Loureiro 585

An Efficient Protocol for Setting Up a Data Dissemination Path in Wireless Sensor Networks
Dongkyun Kim, Gi-Chul Yoo 594

Traffic Control (II)

Active Traffic Monitoring for Heterogeneous Environments
Hélder Veiga, Teresa Pinho, José Luis Oliveira, Rui Valadas, Paulo Salvador, António Nogueira 603

Primary/Secondary Path Generation Problem: Reformulation, Solutions and Comparisons
Quanshi Xia, Helmut Simonis 611

A Discrete-Time HOL Priority Queue with Multiple Traffic Classes
Joris Walraevens, Bart Steyaert, Marc Moeneclaey, Herwig Bruneel 620

SCTP over High Speed Wide Area Networks
Dhinaharan Nagamalai, Seoung-Hyeon Lee, Won-Goo Lee, Jae-Kwang Lee 628

Improving a Local Search Technique for Network Optimization Using Inexact Forecasts
Gilberto Flores Lucio, Martin J. Reed, Ian D. Henning 635

Distributed Addressing and Routing Architecture for Internet Overlays
Damien Magoni, Pascal Lorenz 646

Audio and Video Communications

On Achieving Efficiency and Fairness in Video Transportation
Yan Bai, Yul Chu, Mabo Robert Ito 654

Quality Adapted Backlight Scaling (QABS) for Video Streaming to Mobile Handheld Devices
Liang Cheng, Stefano Bossi, Shivajit Mohapatra, Magda El Zarki, Nalini Venkatasubramanian, Nikil Dutt 662

Video Flow Adaptation for Light Clients on an Active Network
David Fuin, Eric Garcia, Hervé Guyennet 672

Frequency Cross-Coupling Using the Session Initiation Protocol
Christoph Kurth, Wolfgang Kampichler, Karl Michael Göschka 680

IP, ISDN, and ATM Infrastructures for Synchronous Teleteaching - An Application Oriented Technology Assessment
Mustafa Soy, Freimut Bodendorf 690

Sensor Networks (III)

Two Energy-Efficient Routing Algorithms for Wireless Sensor Networks
Hung Le Xuan, Young-koo Lee, Sungyoung Lee 698

An Energy Constrained Multi-hop Clustering Algorithm for Wireless Sensor Networks
Navin Kumar Sharma, Mukesh Kumar 706

Maximizing System Value Among Interested Packets While Satisfying Time and Energy Constraints
Shu Lei, Sungyoung Lee, Wu Xiaoling, Yang Jie 714

An Optimal Coverage Scheme for Wireless Sensor Network
Hui Tian, Hong Shen 722

Routing Protocols Based on Super Cluster Header in Wireless Sensor Network
Jae-hwan Noh, Byeong-jik Lee, Nam-koo Ha, Ki-jun Han 731

Traffic Control (III)

An Automatic and Generic Early-Bird System for Internet Backbone Based on Traffic Anomaly Detection
RongJie Gu, PuLiu Yan, Tao Zou, Chengcheng Guo 740

On Network Model Division Method Based on Link-to-Link Traffic Intensity for Accelerating Parallel Distributed Simulation
Hiroyuki Ohsaki, Shinpei Yoshida, Makoto Imase 749

Network Traffic Sampling Model on Packet Identification
Cheng Guang, Gong Jian, Ding Wei 758

An Admission Control and Deployment Optimization Algorithm for an Implemented Distributed Bandwidth Broker in a Simulation Environment
Christos Bouras, Dimitris Primpas 766

Impact of Traffic Load on SCTP Failovers in SIGTRAN
Karl-Johan Grinnemo, Anna Brunstrom 774

A Novel Method of Network Burst Traffic Real-Time Prediction Based on Decomposition
Xinyu Yang, Yi Shi, Ming Zeng, Rui Zhao 784

Differentiated Services

An EJB-Based Platform for Policy-Based QoS Management of DiffServ Enabled Next Generation Networks
Si-Ho Cha, WoongChul Choi, Kuk-Hyun Cho 794

Determining Differentiated Services Network Pricing Through Auctions
Weilai Yang, Henry L. Owen, Douglas M. Blough 802

A Congestion Control Scheme for Supporting Differentiated Service in Mobile Ad Hoc Networks
Jin-Nyun Kim, Kyung-Jun Kim, Ki-Jun Han 810

Models and Analysis of TCC/AQM Schemes over DiffServ Networks
Jahwan Koo, Jitae Shin, Seongjin Ahn, Jinwook Chung 818

Switching

Choice of Inner Switching Mechanisms in Terabit Router
Huaxi Gu, Zhiliang Qiu, Zengji Liu, Guochang Kang, Kun Wang, Feng Hong 826

Effect of Unbalanced Bursty Traffic on Memory-Sharing Schemes for Internet Switching Architecture
Alvaro Munoz, Sanjeev Kumar 834

New Layouts for Multi-stage Interconnection Networks
Ibrahim Cahit, Ahmet Adalier 842

Packet Scheduling Across Networks of Switches
Kevin Ross, Nicholas Bambos 849

New Round-Robin Scheduling Algorithm for Combined Input-Crosspoint Buffered Switch
Igor Radusinovic, Zoran Veljovic 857

Scheduling Algorithms for Input Queued Switches Using Local Search Technique
Yanfeng Zheng, Simin He, Shutao Sun, Wen Gao 865

Streaming

Multimedia Streaming in Home Environments
Manfred Weihs 873

Joint Buffer Management and Scheduling for Wireless Video Streaming
Günther Liebl, Hrvoje Jenkac, Thomas Stockhammer, Christian Buchner 882

Performance Analysis of a Video Streaming Buffer
Dieter Fiems, Stijn De Vuyst, Herwig Bruneel 892

Feedback Control Using State Prediction and Channel Modeling Using Lower Layer Information for Scalable Multimedia Streaming Service
Kwang O. Ko, Doug Young Suh, Young Soo Kim, Jin Sang Kim 901

Low Delay Multiflow Block Interleavers for Real-Time Audio Streaming
Juan J. Ramos-Muñoz, Juan M. Lopez-Soler 909

A Bandwidth Allocation Algorithm Based on Historical QoS Metric for Adaptive Video Streaming
Ling Guo, YuanChun Shi, Wei Duan 917

Author Index 927

Table of Contents – Part II

MIMO

Decoding Consideration for Space Time Coded MIMO Channel with Constant Amplitude Multi-code System
Jia Hou, Moon Ho Lee, Ju Yong Park, Jeong Su Kim 1

MIMO Frequency Hopping OFDM-CDMA: A Novel Uplink System for B3G Cellular Networks
Laurent Cariou, Jean-Francois Helard 8

Transient Capacity Evaluation of UWB Ad Hoc Network with MIMO
Cheol Y. Jeon, Yeong M. Jang 18

Chip-by-Chip Iterative Multiuser Detection for VBLAST Coded Multiple-Input Multiple-Output Systems
Ke Deng, Qinye Yin, Yiwen Zhang, Ming Luo 26

MPLS

The Performance Analysis of Two-Class Priority Queueing in MPLS-Enabled IP Network
Yun-Lung Chen, Chienhua Chen 34

Constraint Based LSP Handover (CBLH) in MPLS Networks
Praveen Kumar, Niranjan Dhanakoti, Srividya Gopalan, V. Sridhar .. 42

Optimizing Inter-domain Multicast Through DINloop with GMPLS
Huaqun Guo, Lek Heng Ngoh, Wai Choong Wong 50

A Fast Path Recovery Mechanism for MPLS Networks
Jenhui Chen, Chung-Ching Chiou, Shih-Lin Wu 58

A Study of Billing Schemes in an Experimental Next Generation Network
P.S. Barreto, G. Amvame-Nze, C.V. Silva, J.S.S. Oliveira, H.P. de Carvalho, H. Abdalla, A.M. Soares, R. Puttini 66

Overlay Logging: An IP Traceback Scheme in MPLS Network
Wen Luo, Jianping Wu, Ke Xu 75

Ad Hoc Networks (I)

Monitoring End-to-End Connectivity in Mobile Ad-Hoc Networks
Remi Badonnel, Radu State, Olivier Festor 83

Multi-path Routing Using Local Virtual Infrastructure for Large-Scale Mobile Ad-Hoc Networks: Stochastic Optimization Approach
Wonjong Noh, Sunshin An 91

Candidate Discovery for Connected Mobile Ad Hoc Networks
Sebastian Speicher, Clemens Cap 99

A Fault-Tolerant Permutation Routing Algorithm in Mobile Ad-Hoc Networks
Djibo Karimou, Jean Frédéric Myoupo 107

Policy-Based Dynamic Reconfiguration of Mobile Ad Hoc Networks
Marcos A. de Siqueira, Fabricio L. Figueiredo, Flavia M. F. Rocha, Jose A. Martins, Marcel C. de Castro 116

TCP (I)

V-TCP: A Novel TCP Enhancement Technique
Dhinaharan Nagamalai, Beatrice Cynthia Dhinakaran, Byoung-Sun Choi, Jae-Kwang Lee 125

Optimizing TCP Retransmission Timeout
Alex Kesselman, Yishay Mansour 133

Stable Accurate Rapid Bandwidth Estimate for Improving TCP over Wireless Networks
Le Tuan Anh, Choong Seon Hong 141

Performance Analysis of TCP Variants over Time-Space-Labeled Optical Burst Switched Networks
Ziyu Shao, Ting Tong, Jia Jia Liao, Zhengbin Li, Ziyu Wang, Anshi Xu 149

Routing (I)

IPv4 Auto-Configuration of Multi-router Zeroconf Networks with Unique Subnets
Cuneyt Akinlar, A. Udaya Shankar 156

K-Shortest Paths Q-Routing: A New QoS Routing Algorithm in Telecommunication Networks
S. Hoceini, A. Mellouk, Y. Amirat 164

Applicability of Resilient Routing Layers for *k*-Fault Network Recovery
Tarik Čičić, Audun Fosselie Hansen, Stein Gjessing, Olav Lysne 173

Network-Tree Routing Model for Large Scale Networks: Theories and Algorithms
Guozhen Tan, Dong Li, Xiaohui Ping, Ningning Han, Yi Liu 184

Failover for Mobile Routers: A Vision of Resilient Ambience
Eranga Perera, Aruna Seneviratne, Roksana Boreli, Michael Eyrich, Michael Wolf, Tim Leinmüller 192

Quality of Service Routing Network and Performance Evaluation
Lin Shen, Yong Cui, Ming-wei Xu, Ke Xu 202

Ad Hoc Networks (II)

A Partition Prediction Algorithm for Group Mobility in Ad-Hoc Networks
Nam-koo Ha, Byeong-jik Lee, Kyung-Jun Kim, Ki-Jun Han 210

Routing Cost Versus Network Stability in MANET
Md. Nurul Huda, Shigeki Yamada, Eiji Kamioka 218

Multipath Energy Efficient Routing in Mobile Ad Hoc Network
Shouyi Yin, Xiaokang Lin 226

Performance of Service Location Protocols in MANET Based on Reactive Routing Protocols
Hyun-Gon Seo, Ki-Hyung Kim, Won-Do Jung, Jun-Sung Park, Seung-Hwan Jo, Chang-Min Shin, Seung-Min Park, Heung-Nam Kim 234

A New Scheme for Key Management in Ad Hoc Networks
Guangsong Li, Wenbao Han 242

TCP (II)

Robust TCP (TCP-R) with Explicit Packet Drop Notification (EPDN) for Satellite Networks
Arjuna Sathiaseelan, Tomasz Radzik 250

Adapting TCP Segment Size in Cellular Networks
Jin-Hee Choi, Jin-Ghoo Choi, Chuck Yoo 258

AcTMs (Active ATM Switches) with TAP (Trusted and Active PDU Transfers) in a Multiagent Architecture to Better the Chaotic Nature of TCP Congestion Control
José Luis González-Sánchez, Jordi Domingo-Pascual, João Chambel Vieira 266

AIMD Penalty Shaper to Enforce Assured Service for TCP Flows
Emmanuel Lochin, Pascal Anelli, Serge Fdida 275

Routing (II)

Name-Level Approach for Egress Network Access Control
Shinichi Suzuki, Yasushi Shinjo, Toshio Hirotsu, Kazuhiko Kato, Kozo Itano 284

Efficient Prioritized Service Recovery Using Content-Aware Routing Mechanism in Web Server Cluster
Euisuk Kang, SookHeon Lee, Myong-Soon Park 297

Queue Management Scheme Stabilizing Buffer Utilization in the IP Router
Yusuke Shinohara, Norio Yamagaki, Hideki Tode, Koso Murakami . . . 307

Two Mathematically Equivalent Models of the Unique-Path OSPF Weight Setting Problem
Changyong Zhang, Robert Rodošek 318

Fault Free Shortest Path Routing on the de Bruijn Networks
Ngoc Chi Nguyen, Nhat Minh Dinh Vo, Sungyoung Lee 327

Traffic Control in IP Networks with Multiple Topology Routing
Ljiljana Adamovic, Karol Kowalik, Martin Collier 335

Ad Hoc Networks (III)

Dynamic Path Control Scheme in Mobile Ad Hoc Networks Using On-demand Routing Protocol
Jihoon Lee, Wonjong Noh 343

On the Capacity of Wireless Ad-Hoc Network Basing on Graph Theory
Qin-yun Dai, Xiu-lin Hu, Hong-yi Yu, Jun Zhao 353

Mobile Gateways for Mobile Ad-Hoc Networks with Network Mobility Support
Ryuji Wakikawa, Hiroki Matsutani, Rajeev Koodli, Anders Nilsson, Jun Murai 361

Energy Consumption in Multicast Protocols for Static Ad Hoc Networks
Sangman Moh 369

Weighted Flow Contention Graph and Its Applications in Wireless Ad Hoc Networks
Guo-kai Zeng, Yin-long Xu, Ya-feng Wu, Xi Wang 377

Signal Processing

Automatic Adjustment of Time-Variant Thresholds When Filtering Signals in MR Tomography
Eva Gescheidtova, Radek Kubasek, Zdenek Smekal, Karel Bartusek . . . 384

Analytical Design of Maximally Flat Notch FIR Filters for Communication Purposes
Pavel Zahradnik, Miroslav Vlček, Boris Šimák 392

Iterative Decoding and Carrier Frequency Offset Estimation for a Space-Time Block Code System
Ming Luo, Qinye Yin, Le Ding, Yiwen Zhang 401

Signal Processing for High-Speed Data Communication Using Pure Current Mode Filters
Ivo Lattenberg, Kamil Vrba, David Kubánek 410

Current-Mode VHF High-Quality Analog Filters Suitable for Spectral Network Analysis
Kamil Vrba, Radek Sponar, David Kubánek 417

Control of Digital Audio Signal Processing over Communication Networks
Jiri Schimmel, Petr Sysel 425

Routing (III)

Fully-Distributed and Highly-Parallelized Implementation Model of BGP4 Based on Clustered Routers
Xiao-Zhe Zhang, Pei-dong Zhu, Xi-Cheng Lu 433

A Routing Protocol for Wireless Ad Hoc Sensor Networks: Multi-Path Source Routing Protocol (MPSR)
Mounir Achir, Laurent Ouvry 442

Generalized Secure Routerless Routing
Vince Grolmusz, Zoltán Király 454

A Verified Distance Vector Routing Protocol for Protection of Internet Infrastructure
Liwen He 463

Replay Attacks in Mobile Wireless Ad Hoc Networks: Protecting the OLSR Protocol
Eli Winjum, Anne Marie Hegland, Øivind Kure, Pål Spilling 471

S-Chord: Hybrid Topology Makes Chord Efficient
Hui-shan Liu, Ke Xu, Ming-wei Xu, Yong Cui 480

Mobility

Hierarchical Multi-hop Handoff Architecture for Wireless Network Mobility
Yunkuk Kim, Sangwook Kang, Donghyun Chae, Sunshin An 488

Mobility Adaptation Layer Framework for Heterogeneous Wireless Networks Based on Mobile IPv6
Norbert Jordan, Alexander Poropatich, Joachim Fabini 496

MiSC: A New Availability Remote Storage System for Mobile Appliance
Joo-Ho Kim, Bo-Seok Moon, Myong-Soon Park 504

A Logical Network Topology Design for Mobile Agent Systems
Kazuhiko Kinoshita, Nariyoshi Yamai, Koso Murakami 521

Reduced-State SARSA Featuring Extended Channel Reassignment for Dynamic Channel Allocation in Mobile Cellular Networks
Nimrod Lilith, Kutluyıl Doğançay 531

Call Admission Control for Next Generation Cellular Networks Using on Demand Round Robin Bandwidth Sharing
Kyungkoo Jun, Seokhoon Kang 543

Performance (I)

Performance Evaluation and Improvement of Non-stable Resilient Packet Ring Behavior
Fredrik Davik, Amund Kvalbein, Stein Gjessing 551

Load Distribution Performance of the Reliable Server Pooling Framework
Thomas Dreibholz, Erwin P. Rathgeb, Michael Tüxen 564

Performance of a Hub-Based Network-Centric Application over the Iridium Satellite Network
Margaret M. McMahon, Eric C. Firkin 575

Performance Evaluation of Multichannel Slotted-ALOHA Networks with Buffering
Sebastià Galmés, Ramon Puigjaner 585

Towards a Scalable and Flexible Architecture for Virtual Private Networks
Shashank Khanvilkar, Ashfaq Khokhar 597

Peer-to-Peer (I)

A Simple, Efficient and Flexible Approach to Measure Multi-protocol Peer-to-Peer Traffic
Holger Bleul, Erwin P. Rathgeb 606

Secure Identity and Location Decoupling Using Peer-to-Peer Networks
Stephen Herborn, Tim Hsin-Ting Hu, Roksana Boreli, Aruna Seneviratne 617

Live Streaming on a Peer-to-Peer Overlay: Implementation and Validation
Joaquín Caraballo Moreno, Olivier Fourmaux 625

Distributed Object Location with Queue Management Provision in Peer-to-Peer Content Management Systems
Vassilios M. Stathopoulos, Nikolaos D. Dragios, Nikolas M. Mitrou . . . 634

An Approach to Fair Resource Sharing in Peer-to-Peer Systems
Yongquan Ma, Dongsheng Wang 643

Discovery and Routing in the HEN Heterogeneous Peer-to-Peer Network
Tim Schattkowsky 653

Security (I)

Scalable Group Key Management with Partially Trusted Controllers
Himanshu Khurana, Rafael Bonilla, Adam Slagell, Raja Afandi, Hyung-Seok Hahm, Jim Basney 662

H.323 Client-Independent Security Approach
Lubomir Cvrk, Vaclav Zeman, Dan Komosny 673

Architecture of Distributed Network Processors: Specifics of Application in Information Security Systems
V.S. Zaborovskii, Y.A. Shemanin, A. Rudskoy 681

Active Host Information-Based Abnormal IP Address Detection
Gaeil Ahn, Kiyoung Kim 689

Securing Layer 2 in Local Area Networks
Hayriye Altunbasak, Sven Krasser, Henry L. Owen, Jochen Grimminger, Hans-Peter Huth, Joachim Sokol 699

A Practical and Secure Communication Protocol in the Bounded Storage Model
E. Savaş, Berk Sunar 707

Performance (II)

Measuring Quality of Service Parameters over Heterogeneous IP Networks
A. Pescapé, L. Vollero, G. Iannello, G. Ventre 718

Performance Improvement of Hardware-Based Packet Classification Algorithm
Yaw-Chung Chen, Pi-Chung Wang, Chun-Liang Lee, Chia-Tai Chan 728

Analyzing Performance Data Exchange in Content Delivery Networks
Davide Rossi, Elisa Turrini 737

Passive Calibration of Active Measuring Latency
Jianping Yin, Zhiping Cai, Wentao Zhao, Xianghui Liu 746

Peer-to-Peer (II)

Application-Level Multicast Using DINPeer in P2P Networks
Huaqun Guo, Lek Heng Ngoh, Wai Choong Wong 754

Paradis-Net: A Network Interface for Parallel and Distributed Applications
Guido Malpohl, Florin Isailă 762

Reliable Mobile Ad Hoc P2P Data Sharing
Mee Young Sung, Jong Hyuk Lee, Jong-Seung Park, Seung Sik Choi, Sungtek Kahng 772

The Hybrid Chord Protocol: A Peer-to-Peer Lookup Service for Context-Aware Mobile Applications
Stefan Zöls, Rüdiger Schollmeier, Wolfgang Kellerer, Anthony Tarlano 781

LQPD: An Efficient Long Query Path Driven Replication Strategy in Unstructured P2P Network
Xi-Cheng Lu, Qianbing Zheng, Pei-Dong Zhu, Wei Peng 793

Content Distribution in Heterogenous Video-on-Demand P2P Networks with ARIMA Forecasts
Chris Loeser, Gunnar Schomaker, André Brinkmann, Mario Vodisek, Michael Heidebuer 800

Security (II)

Critical Analysis and New Perspective for Securing Voice Networks
Carole Bassil, Ahmed Serhrouchni, Nicolas Rouhana 810

Architecture of a Server-Aided Signature Service (SASS) for Mobile Networks
Liang Cai, Xiaohu Yang, Chun Chen 819

Password Authenticated Key Exchange for Resource-Constrained Wireless Communications
Duncan S. Wong, Agnes H. Chan, Feng Zhu 827

An Efficient Anonymous Scheme for Mutual Anonymous Communications
Ray-I Chang, Chih-Chun Chu 835

GDS Resource Record: Generalization of the Delegation Signer Model
Gilles Guette, Bernard Cousin, David Fort 844

Secure Initialization Vector Transmission on IP Security
Yoon-Jung Rhee 852

Multicast (I)

Multicast Receiver Mobility over Mobile IP Networks Based on Forwarding Router Discovery
Takeshi Takahashi, Koichi Asatani, Hideyoshi Tominaga 859

Secure Multicast in Micro-Mobility Environments
Ho-Seok Kang, Young-Chul Shim 868

Scalability and Robustness of Virtual Multicast for Synchronous Multimedia Distribution
Petr Holub, Eva Hladká, Ludek Matyska 876

Mobile Multicast Routing Protocol Using Prediction of Dwelling Time of a Mobile Host
Jae Keun Park, Sung Je Hong, Jong Kim 884

A Group Management Protocol for Mobile Multicast
Hidetoshi Ueno, Hideharu Suzuki, Norihiro Ishikawa 892

CDMA

Propagation Path Analysis for Location Selection of Base-Station in the Microcell Mobile Communications
Sun-Kuk Noh, Dong-You Choi, Chang-kyun Park 904

Efficient Radio Resource Management in Integrated WLAN/CDMA Mobile Networks
Fei Yu, Vikram Krishnamurthy 912

A Study on the Cell Sectorization Using the WBTC and NBTC in CDMA Mobile Communication Systems
Dong-You Choi, Sun-Kuk Noh 920

DOA-Matrix Decoder for STBC-MC-CDMA Systems
Yanxing Zeng, Qinye Yin, Le Ding, Jianguo Zhang 928

Erlang Capacity of Voice/Data CDMA Systems with Service Requirements of Blocking Probability and Delay Constraint
Insoo Koo, Jeongrok Yang, Kiseon Kim 936

Security and Network Anomaly Detection

A Simplified Leakage-Resilient Authenticated Key Exchange Protocol with Optimal Memory Size
SeongHan Shin, Kazukuni Kobara, Hideki Imai 944

The Fuzzy Engine for Random Number Generator in Crypto Module
Jinkeun Hong 953

A Packet Marking Scheme for IP Traceback
Haipeng Qu, Purui Su, Dongdai Lin, Dengguo Feng 964

Securing Admission Control in Ubiquitous Computing Environment
Jong-Phil Yang, Kyung Hyune Rhee 972

Detecting the Deviations of Privileged Process Execution
Purui Su, Dequan Li, Haipeng Qu, Dengguo Feng 980

Dynamic Combination of Multiple Host-Based Anomaly Detectors with Broader Detection Coverage and Fewer False Alerts
Zonghua Zhang, Hong Shen 989

Impact of Distributed Denial of Service (DDoS) Attack Due to ARP Storm
Sanjeev Kumar 997

Multicast (II)

Design of Network Management System Employing Secure Multicast SNMP
Deuk-Whee Kwak, JongWon Kim 1003

Multi-rate Congestion Control over IP Multicast
Yuliang Li, Alistair Munro, Dritan Kaleshi 1012

A TCP-Friendly Multicast Protocol Suite for Satellite Networks
Giacomo Morabito, Sergio Palazzo, Antonio Pantò 1023

An Enhanced Multicast Routing Protocol for Mobile Hosts in IP Networks
Seung Jei Yang, Sung Han Park 1031

Analysis of Handover Frequencies for Predictive, Reactive and Proxy Schemes and Their Implications on IPv6 and Multicast Mobility
Thomas C. Schmidt, Matthias Wählisch 1039

802.11 Networks

Design Architectures for 3G and IEEE 802.11 WLAN Integration
F. Siddiqui, S. Zeadally, E. Yaprak 1047

Eliminating the Performance Anomaly of 802.11b
See-hwan Yoo, Jin-Hee Choi, Jae-Hyun Hwang, Chuck Yoo 1055

Energy Efficiency Analysis of IEEE 802.11 DCF with Variable Packet Length
Bo Gao, Yuhang Yang, Huiye Ma 1063

Scheduling MPEG-4 Video Streams Through the 802.11e Enhanced Distributed Channel Access
Michael Ditze, Kay Klobedanz, Guido Kämper, Peter Altenbernd 1071

IEEE 802.11b WLAN Performance with Variable Transmission Rates: In View of High Level Throughput
Namgi Kim, Sunwoong Choi, Hyunsoo Yoon 1080

Emergency, Disaster, Resiliency

Some Principles Incorporating Topology Dependencies for Designing Survivable WDM Optical Networks
Sungwoo Tak 1088

Resilient Routing Layers for Network Disaster Planning
Audun Fosselie Hansen, Amund Kvalbein, Tarik Čičić, Stein Gjessing 1097

Design of a Service Discovery Architecture for Mobility-Supported Wired and Wireless Networks
Hyun-Gon Seo, Ki-Hyung Kim 1106

Research on Fuzzy Group Decision Making in Security Risk Assessment
Fang Liu, Kui Dai, Zhiying Wang, Jun Ma 1114

A Resilient Multipath Routing Protocol for Wireless Sensor Networks
Ki-Hyung Kim, Won-Do Jung, Jun-Sung Park, Hyun-Gon Seo, Seung-Hwan Jo, Chang-Min Shin, Seung-Min Park, Heung-Nam Kim 1122

A Multilaterally Secure, Privacy-Friendly Location-Based Service for Disaster Management and Civil Protection
Lothar Fritsch, Tobias Scherner 1130

Survivability-Guaranteed Network Resiliency Methods in DWDM Networks
Jin-Ho Hwang, Won Kim, Jun-Won Lee, Sung-Un Kim 1138

Author Index 1147

Mobile-to-Grid Middleware: An Approach for Breaching the Divide Between Mobile and Grid Environments

Umar Kalim, Hassan Jameel, Ali Sajjad, and Sungyoung Lee

Department of Computer Engineering, Kyung Hee University,
Sochen-ri, Giheung-eup, Yongin-si, Gyeonggi-do, 449-701, South Korea
{umar, hassan, ali, sylee}@oslab.khu.ac.kr

Abstract. In this paper we present an architecture of a middleware layer[1] that enables users of mobile devices to seamlessly and securely access distributed resources in a Grid. It lays the ground work for an application toolkit that addresses issues such as delegation of the job to the Grid service, interaction with heterogeneous mobile devices, support for offline processing, secure communication between the client and the middleware and presentation of results formatted in accordance with the device specification by outsourcing computationally intensive tasks with high storage and network bandwidth demands.

1 Introduction

Grid [1] computing is based on an open set of standards and protocols that enable coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organizations [2]. With Grid computing, organizations can optimize computing and data resources by pooling them for large capacity workloads, share them across networks and enable collaboration. Though the concept of Grid computing is still evolving, yet there have been a number of achievements in the arena of scientific applications [3], [4], [5]. Extending this potential of the Grid to a wider audience, promises increase in productivity, particularly for users of mobile devices who are the prospective users of this technology.

Wireless environments and mobile devices bring different challenges when compared to wired networks and workstations. Although mobile devices promote mobile communication and flexible usage, yet they bring along problems such as unpredictable network quality, lower trust, limited resources (power, network bandwidth etc) and extended periods of disconnections [6]. If such resource limited mobile devices could access and utilize the Grid's resources then they could implicitly obtain results from resource intensive tasks never thought of before.

[1] This research work has been partially supported by Korea Ministry of Information and Communications' ITRC Program joint with Information and Communications University.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 1–8, 2005.

The classical client server approach assumes that the location of computational and storage resources is known [7]. This approach has to evolve in order to provide transparent access to distributed resources. Also considering the limitations [8] of interaction among mobile devices and grid nodes as well as the complexity of the Grid protocols, there is an emerging consensus [8] to develop a middleware layer which will mediate and manage access to distributed resources. Besides the clear separation among the key functionality, the introduction of a middleware layer can offer potential technical advantages. Among them are reduced communication cost, reduced network bandwidth usage, the possibility of using remote interfaces and the support for off-line computation.

In this paper we present an architecture for a middleware (Section 3), enabling heterogeneous mobile devices access to Grid services and implement an application toolkit that acts as a gateway to the Grid. This middleware provides support for seamless delegation of jobs to the Grid, secure communication between the client and the Grid (the same level of security that RSA provides by using smaller key sizes), offline processing, adaptation to network connectivity issues and presentation of results in a form that is in keeping with the resources available at the client device.

2 Mobile-to-Grid Middleware

Considering the constraints of mobile devices and how operations take place within a Grid environment [1], demands for computational as well as network bandwidth resources are intense. These demands for resources make it difficult for the developers to implement practical applications for the users of mobile devices. The problems [8] of mobile and wireless environments aggravate the dilemma. Hence there is a need for a middleware which could operate on behalf of the client and interact with the Grid services in such a manner that the client application is only required to participate primarily at only two instances; firstly before submitting the job and secondly when collecting the results so that the client application is not obliged to steer the process.

There are a number of areas that need to be addressed, namely, job delegation to the Grid, management of the request, handling of disconnections in the wireless environment, the security aspects between the client and the middleware layer, formalization of results depending upon the client's device specification and managing all this information efficiently etc.

To instantiate a job, the client must be able to instruct the Grid service. However, direct interaction with the Grid service results in exhausting demands for storage and computational resources on the client device. Similarly if the request to a Grid service requires continuous steering from the client, this puts strenuous demands on the computational resources and network bandwidth.

If the job submitted by the user is expected to complete in a long duration of time, conventionally the user is bound to maintain an active connection to obtain the results. Also if the network connection goes down (due to power loss, being out of range etc) the user would lose his connection and would have to start mediating with the Grid service from scratch which would result in loss of precious time and resources.

The Grid security infrastructure is based on public key scheme mainly deployed using the RSA algorithm [9]. However key sizes in the RSA scheme are large and thus computationally heavy on handheld devices such as PDA's, mobile phone's etc [10]. Also a user accessing the Grid services must use credentials which are compliant with the X.509 proxy credentials of GSI [9]. Transferring user credentials to access the Grid services without creating a security hazard and demanding relatively large computational and bandwidth resources is another issue that needs to be addressed.

The introduction of a middleware layer allows the user to reduce computational activities at the client device, minimize usage of network bandwidth, save battery power, permit offline processing etc. This is achieved as the middleware acts as a broker and interacts with the Grid on behalf of the client.

3 Detailed Architecture

The primary steps that may occur while the client device accesses the Grid services may be explained as follows. Firstly the client application discovers and connects with the middleware. Then the device, after authentication, submits its device specification along with the job request. The middleware then locates the relevant Grid service and after authorization forwards the request. The client may then request some status information (regarding the job or the service). If the client wishes to disconnect (and collect the results later), the middleware would facilitate a soft state registration and to which would later help in the reintegration. After disconnection all the requests are served locally (with the cached information). Requests that result in updates at the middleware service are logged for execution at reconnection. Upon reconnection pending instructions are executed and information updates at the client end are made to maintain consistency. Considering these steps the details of the modules involved (shown in Figure 1) are mentioned below.

3.1 Discovery Service

The discovery of the middleware by mobile devices is managed by employing a UDDI registry [11], [12]. Once the middleware service is deployed and registered, other applications/devices would be able to discover and invoke it using the API in the UDDI specification [11] which is defined in XML, wrapped in a SOAP [7] envelop and sent over HTTP.

3.2 Communication Interface with the Client Application

The interface advertised to the client application is the communication layer between the mobile device and the middleware. This layer enables the middleware to operate as a web service and communicate via the SOAP framework [13].

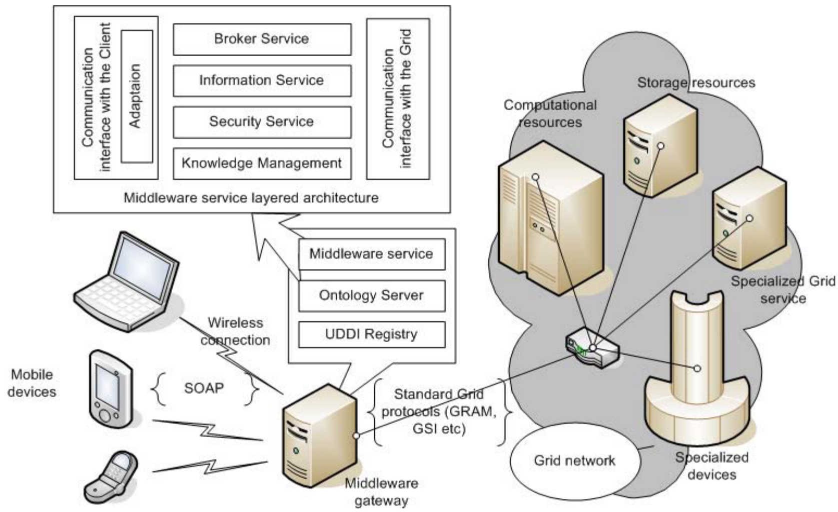


Fig. 1. Deployment model and the architecture

Adaption to Disconnected Operations. The advertisement of the mobile-to-Grid middleware as a web service permits the development of the architecture in a manner that does not make it mandatory for the client application to remain connected to the middleware at all times while the request is being served.

We focus on providing software support for offline processing at the client device. For this we consider two kinds of disconnections; intentional disconnection, where the user decides to discontinue the wireless connection and unintentional disconnection, which might occur due to variation in bandwidth, noise, lack of power etc. This is made possible by pre-fetching information or meta-data only from the middleware service. This facilitates in locally serving the client application at the device. However, requests that would result in updates at the middleware service are logged (so that they may be executed upon reconnection).

To establish the mode of operation for the client application, a connection monitor is used to determine the network bandwidth and consequently the connection state (connected or disconnected). Moreover, during execution, checkpoints are maintained at the client and the middleware in order to optimize reintegration after disconnection.

3.3 Communication Interface with the Grid

The communication interface with the Grid provides access to the Grid services by creating wrappers for the API advertised by the Grid. These wrappers include standard Grid protocols such as GRAM [14], MDS [15], GSI [16] etc which are mandatory for any client application trying to communicate with the Grid services. This enables the middleware to communicate with the Grid, in order to accomplish the job assigned by the client.

3.4 Broker Service

The broker service deals with initiating the job request and steering it on behalf of the client application. Firstly the client application places a request for a job submission. After determining the availability of the Grid service and authorization of the client, the middleware downloads the code (from the mobile device or from a location specified by the client e.g. an FTP/web server). Once the code is available, the broker service submits a "createService" request on the GRAM's Master Managed Job Factory Service (via the wrapper) which is received by the Redirector [14]. The application code (controlled by the middleware) then interacts with the newly created instance of the service to accomplish the task. The rest of the process including creating a Virtual Host Environment (VHE) process and submitting the job to a scheduling system is done by GRAM. Subsequent requests by the client code to the broker service are redirected through the GRAM's Redirector.

The Status monitor (a subset of the broker service) interacts with GRAM's wrapper to submit FindServiceData requests in order to determine the status of the job. The Status monitor service then communicates with the Knowledge Management module to store the results. The mobile client may reconnect and ask for the (intermediate/final) results of its job from the status monitor service.

3.5 Knowledge Management

The knowledge management layer of the system is used to manage the relevant information regarding both the client and Grid applications and services. The main function of this layer is to connect the client and Grid seamlessly as well as to introduce the capability of taking intelligent decisions, based on the information available to the system.

Also, the results to be presented to the client are formatted (or scaled down) here considering the device profiles maintained at the ontology server.

3.6 Security

The Grid Security Infrastructure is based on public key scheme mainly deployed using the RSA algorithm [9]. However key sizes in the RSA scheme are large and thus computationally heavy on handheld devices such as PDA's, mobile phone's, smart phones etc [10]. We employ the Web Services Security Model [17] to provide secure mobile access to the Grid. This web services model supports multiple cryptographic technologies.

The Elliptic Curve Cryptography based public key scheme can be used in conjunction with Advanced Encryption Standard for access to the Grid. This provides the same level of security as RSA while the key sizes are a smaller [10].

Communication between the user and middleware is based on security policies specified in the user profile. According to this policy different levels of security can be used. e.g. some users might just require authentication, and need not want privacy or integrity of messages. Both ECC and AES have smaller key sizes as compared to RSA [10] which means faster computation, low memory, bandwidth and power consumption with high level of security. It may be noted that we emphasize on providing security on Application layer, which also gives us the flexibility to change the security mechanism if the need arises.

4 Information Service

This module interacts with the wrapper of the GLOBUS toolkit's API for information services (MDS [15]). It facilitates the client application by managing the process of determining which services and resources are available in the Grid (the description of the services as well as resource monitoring such as CPU load, free memory etc. Detailed information about grid nodes (which is made available by MDS) is also shared on explicit request by the client.

5 Multiple Instances of the Middleware Gateway

In case multiple instances of the middleware gateway are introduced for scalability, some problematic scenarios might arise. Consider a client that accesses the Grid via gateway M_1, but disconnects after submitting the job. If the client later reconnects at gateway M_2 and inquires about its job status, the system would be unable to respond if the middleware is not capable of sharing information with other instances. This can be achieved in the following manner.

We define a Middleware Directory Listing which maintains the ordered pairs (ID, URI) which will be used for the identification of the middleware instance. Also, we define an X service as a module of the middleware which facilitates the communication between any two middleware instances. After reintegration of the client at M_2, client C sends the ID of the middleware instance, where the job was submitted (i.e. M_1), to the X service. The X service determines that the ID is not that of M_2. The X service then checks the Middleware Directory Listing to find the URI corresponding to M_1. The X service then requests the job-ID submitted by C. Upon a successful response the X service communicates with the X service of M_1 using the URI retrieved. After mutual authentication, X-M_2 sends the job-ID along with the clients request for fetching the (intermediate/final) results to X-M_1. If the job is complete, the compiled results are forwarded to client. In case the job isn't complete yet, the client continues to interact with middleware service X-M_1 (where the job was submitted). Note that X-M_2 acts as a broker for communication between C and M_1. Also, if the C decides to disconnect and later reconnect at a third middleware instance M_3, then M_3 will act as a broker and communicate with M_1 on behalf of C. As all the processing of information is done at the middleware where the job was submitted, the other instances would only act as message forwarding agents.

6 Related Work

Various efforts have been made to solve the problem of mobile-to-Grid middleware. Signal [18] proposes a mobile proxy-based architecture that can execute jobs submitted to mobile devices, so in-effect making a grid of mobile devices. A proxy interacts with the Globus Toolkit's MDS to communicate resource availability in the nodes it represents. The proxy server and mobile device communicate via SOAP and authenticate each other via the generic security service (GSS) API. The proxy server analyzes code and checks for resource allocation through the monitoring and discovery service (MDS). After the proxy server determines resource availability, the adaptation middleware layer component in the server sends the job request to remote locations. Because of this distributed execution, the mobile device consumes little power and uses bandwidth effectively. Also their efforts are more inclined towards QoS issues such as management of allocated resources, support for QoS guarantees at application, middleware and network layer and support of resource and service discoveries based on QoS properties.

In [19] a mobile agent paradigm is used to develop a middleware to allow mobile users' access to the Grid and it focus's on providing this access transparently and keeping the mobile host connected to the service. Though they have to improve upon the system's security, fault tolerance and QoS, their architecture is sufficiently scalable. GridBlocks [20] builds a Grid application framework with standardized interfaces facilitating the creation of end user services. They advocate the use of propriety protocol communication protocol and state that SOAP usage on mobile devices maybe 2-3 times slower as compared to a proprietary protocol. For security, they are inclined towards the MIDP specification version 2 which includes security features on Transport layer.

7 Future Work

Some devices may not be able to efficiently process SOAP messages. Therefore we intend to provide multi-protocol support in order to extend the same facilities to such devices. However our first and foremost goal is to complete the implementation of the architecture using Java's support for web services and devices using 802.11b wireless interface cards. Our main focus will be on handling security, providing support for offline processing, presentation of results depending upon the device specification and interfacing with the Grid nodes. Along with this implementation we intend to validate our approach by using non-Markovian stochastic Petri nets to analyze the model.

8 Conclusion

In this paper we identified the potential of enabling mobile devices access to the Grid. We focused on providing solutions related to distributed computing in wireless environments, particularly when mobile devices intend to interact with grid services. An architecture for a middleware is presented which facilitates implicit interaction of mobile devices with grid services. This middleware is based on the web services communication paradigm. It handles secure communication between the client and the middleware service, provides software support for offline processing, manages the presentation of results to heterogeneous devices (i.e. considering the device specification) and deals with the delegation of job requests from the client to the Grid.

References

1. Foster, I., et al.: The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration, Proc. 4th Global Grid Forum, Open Grid Services Infrastructure working group, Global Grid Forum. (2002), http://www.gridforum.org/Meetings/ggf4/default.htm.
2. Foster, I. et al.: The Anatomy of the Grid: Enabling Scalable Virtual Organizations, Int'l J. Supercomputer Applications, vol. 15, no. 3, (2001), pp.200-222.
3. EU DataGrid. http://eu-datagrid.web.cern.ch/eu-datagrid/
4. Particle Physics Data Grid. http://www.ppdg.net
5. BioGrid. http://www.biogrid.jp
6. Forman, G. and Zahorjan, J.: The Challenges of Mobile Computing, IEEE Computer, vol. 27, no. 4, (April 1994).
7. Tanenbaum, A. S., et. al.: Distributed Systems Principles and Paradigms. Prentice Hall, 1st edition, pp. 42, 43. ISBN 81-7808-789-8.
8. Wen, Y.: Mobile Grid. http://pompone.cs.ucsb.edu/ wenye/majorexam/writeup.pdf
9. Von Welch, Foster, I., Carl Kesselman, et al.: X.509 Proxy Certificates for dynamic delegation. Proceedings of the 3rd Annual PKI R&D Workshop, (2004).
10. Vipul Gupta, Sumit Gupta, et al.: Performance Analysis of Elliptic Curve Cryptography for SSL. Proceedings of ACM Workshop on Wireless Security - WiSe 2002 pp. 87-94, Atlanta, GA, USA, (September 2002), ACM Press.
11. Hoschek, H.: Web service discovery processing steps. http://www-itg.lbl.gov/ hoschek/publications/icwi2002.pdf
12. UDDI specification. www.oasis-open.org/committees/uddi-spec/doc/tcspecs.htm
13. SOAP Framework: W3C Simple Object Access Protocol ver 1.1, World Wide Web Consortium recommendation, (8 May 2000); www.w3.org/TR/SOAP/
14. GT3 GRAM Architecture, www-unix.globus.org/developer/gram-architecture.html
15. Czajkowski, K. et al.: Grid Information Services for Distributed Resource Sharing. Proceedings of the Tenth IEEE International Symposium on High-Performance Distributed Computing (HPDC-10), IEEE Press, (August 2001)
16. Welch, V. Siebenlist, F. Foster, I. et al.: Security for grid services. HPDC, 2003.
17. Della-Libera Giovanni, et al., Security in a Web Services World; A Proposed Architecture and Roadmap, (2002), A joint security whitepaper from IBM Corporation and Microsoft Corporation. (April 7, 2002), Version 1.0
18. Hwang, J. Aravamudham, P.: Middleware Services for P2P Computing in Wireless Grid Networks. IEEE Internet Computing vol. 8, no. 4, (July/August 2004), pp. 40-46
19. Bruneo, D. et al.: Communication Paradigms for Mobile Grid Users. Proceedings 10th IEEE International Symposium in High-Performance Distributed Computing.
20. Gridblocks project (CERN) http://gridblocks.sourceforge.net/docs.htm

On the Influence of Network Characteristics on Application Performance in the Grid Environment

Yoshinori Kitatsuji[1], Satoshi Katsuno[2], Katsuyuki Yamazaki[2], Hiroshi Koide[3], Masato Tsuru[3], and Yuji Oie[3]

[1] NICT Kitakyushu JGN2 Research Center, 3–8–1 Asano, Kokurakita–ku, Kitakyushu–shi, Fukuoka, Japan
kitaji@kyushu.jgn2.jp

[2] KDDI R&D Laboratories, Inc., 2–1–15 Ohara Kamifukuoka-shi, Saitama, Japan
{katsuno, yamazaki}@kddilabs.jp

[3] Kyushu Institute of Technology, 680–4, Kawatsu Iizuka–shi, Fukuoka, Japan
koide@ai.kyutech.ac.jp
{tsuru, oie}@cse.kyutech.ac.jp

Abstract. In the Grid computing, it is a key issue how limited network resources are effectively shared by communications of various applications in order to improve the application-level performance, e.g., to reduce the completion time of each application and/or a set of applications. In fact, the communication of an application changes the condition of network resources, which may, in turn, affect the communications in other applications, and thus may deteriorate their performance. In this paper, we examine the characteristics of traffic generated by some typical grid applications, and how the round-trip time and the bottleneck bandwidth affect the application-level performance (i.e., completion time) of these applications. Our experiments show that the impact of network conditions on the application performance and the impact of application traffic on the network conditions are considerably different depending on the application. Those results suggest an effective network resource allocation should take network-related properties of individual applications into consideration.

1 Introduction

Along with the deployment of high-performance off-the-shelf computers and high-speed wide-area networks, large-scale distributed computing environments are growing at an amazing speed. In such environments, massive computations are performed using a large number of computers connected over WAN (Wide Area Networks). Such a form of distributed computing (the grid computing) dynamically involves a number of heterogeneous computing resources (e.g., CPU,

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 9–18, 2005.

memory, storage, application program, data, etc.) and heterogeneous network resources for connecting them, across geographically dispersed organizations. [1][2] The fundamental challenge in the grid computing is to perform multiple distributed applications sharing the limited and/or heterogeneous computing and network resources so that they achieve a good performance (e.g., the completion time of each distinct application and/or that of a set of related applications). In particular, as the amount of data handled by the distributed applications has recently increased, the time required to transmit data increases, and thus, begins to strongly affect the application performance. The effective share of the network resources is therefore of significant importance.

There are numerous studies on the traffic engineering to improve the network resource utilization. Elwaid *et al.* propose a decentralized method to balance flows over multiple paths based upon the traffic load of the paths obtained by an active end-to-end measurement along the paths.[3] Guven *et al.* propose that routers passively measure loss and available bandwidth of connected links by the measurement entities for re-balancing flows over multiple paths.[4] Although both propose re-balancing traffic over multiple paths, they consider neither the traffic patterns generated by applications nor the application-level performance.

Suppose that the network-related properties of each application can be obtained based on either a test run beforehand or the first run in a series of repeated runs. The network-related properties of an application can have two aspects: how each application performance is affected by the condition of the network resources and, inversely, how the traffic generated by each application affects the network resource condition. In such a scenario, it is expected that those distributed applications can be scheduled to effectively share the network resources on the internal-network as well as the computing resources on the end-nodes, by taking such network-related properties of applications into account. In this paper, therefore, we investigate if the applications show the peculiar characters on the performance related to network to meet this issue, especially focusing on the sensitivities of some typical distributed applications in the completion time to the condition of network resources. The results of our preliminary experiment show a non-trivial tension between the performance of applications and the condition of network resources, which can be utilized to achieve an effective share of the network resources to execute multiple applications in parallel.

2 Application Features and Distributed Computing Experiment Environment

We here describe the target applications employed in our study and the network environment on which those applications run.

2.1 Features of Target Applications

In this section we describe applications employed in the study: N Queen, Jigsaw Puzzle and Task Scheduling, that can run on the grid environment.

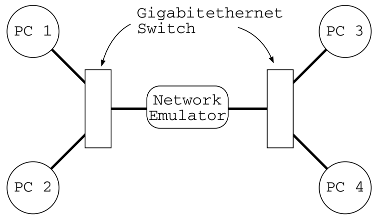


Fig. 1. Network Configuration

- *N Queen* solves placing N Queens on N by N grid such that none of them shares a common row, column or diagonal.
- *Jigsaw Puzzle* solves a jigsaw puzzle for computers, which was originally from Problem C in the 2001 ACM International College Programming Contest, Asia Preliminaries (Hakodate) [5] and has been expanded to take the rotation and the size of a piece into account.
- *Task Scheduling* is a task scheduling program subjecting the standard task graph archives [6]. Its algorithm is to assign a task to available computers based upon its priority. A task is given higher priority as a task flow (the critical path), to which the task belongs, requires the longer time. In selecting computers, the available memory is evaluated if the computer has enough memory to perform task.[7]

The type of distributed processing in N Queen and Jigsaw Puzzle is classified into Task-Framing and Task Scheduling is into Work Flow.[8] In the Task-Framing processing, a master distributes small tasks composing the target problem among a farm of multiple slave processors and gathers the partial results in order to produce the final result of the computation. Huge data transfers are expected to occur on the communications taking place only between the master and the slaves. In the Work Flow processing on the other hand, the target problem is divided up into multiple pipelined stages and/or steps. Various amounts of data are expected to be asynchronously exchanged on each pair of processors.

2.2 Network Configuration for Experiments

We employ the network configuration in Figure 1 for experiments described in Section 3. Distributed processing for each application described in Section 2.1 is performed on computers PC1 through PC4 with starting up from PC1. The basic features of all the computers are: Xeon 3.06 GHz CPU, 2 GBytes memory, Intel (R) PRO/1000 NIC and PCI-X bus. The speed of all links is 1 Gbit/s.

For the Task-Framing applications, the master process runs on PC1, and the slave processes run on all the computers including PC1. A network emulator [9] is employed between PC1, 2 and PC3, 4 to insert latency and to shape a bottleneck link in packet forwarding between two switches. The measurements are performed by capturing all the packets sent to/received from each of the computers. The average round-trip times (RTTs) are, respectively, 0.141 and

0.331 millisecond between PC1 and PC2, and between PC1 and PC3 without any latency inserted.

3 Analysis on Network-Related Characteristics of the Applications

In this section, we investigate network-related characteristics of the targeted applications listed in Table 1. Note that, in Figures 3, 4 and 5, we only illustrate the results in the cases of the 16 × 16 grids for N Queen, the 36 × 36 pieces for Jigsaw Puzzle, and the 500 tasks for Task Scheduling because the alternative cases have tended to show the results quite similar to the above cases.

3.1 Features of Traffic

We investigate the features of the application traffic, e.g., the amount of transferred data, fluctuations of throughput and flows composing the application traffic. We run each of the applications in the network described in Section 2.2 with the sufficient network resources: the average RTTs are 0.114 and 0.331 millisecond; the bandwidth of links is 1 Gbit/s.

Table 1 shows the total amount of transferred data and the average throughput on PC3, and the completion time for each of the applications. The values are the average on 20 experiments. All the applications take longer time to complete their processing and transmit a larger amount of data as the scales of their problems become larger. The relations between the amount of transmitted data and the completion time heavily depend on its applications. N Queen increases an amount of data 5 times and its completion time increases 2.5 times. Jigsaw Puzzle increases its completion time 10 times while the amount of data significantly increases, more than 1000 times. Moreover, Task Scheduling roughly unchanges its completion time while the amount of data increases 1.5 times. In Jigsaw Puzzle and Task Scheduling, the reason that the completion time doesn't increase as much as data must be that the average throughput increases to convey the larger amount of data.

Table 1. Traffic features of the each application

Application	Incoming Traffic to PC3		Outgoing Traffic from PC3		Completion Time
	Total Amount of Data	Average Throughput	Total Amount of Data	Average Throughput	
N Queen, 15×15 grd	0.926 MB	0.878 Mbit/s	39.2 MB	37.2 Mbit/s	40.78 s
N Queen, 16×16 grd	4.07 MB	1.45 Mbit/s	210 MB	75.4 Mbit/s	191.33 s
Jigsaw Puzzle, 4×4 pc	0.0780 MB	0.0443 Mbit/s	0.117 MB	0.0666 Mbit/s	13.70 s
Jigsaw Puzzle, 36×36 pc	101 MB	4.56 Mbit/s	194 MB	8.79 Mbit/s	176.48 s
Task Scheduling, 300 tsk	116 MB	7.68 Mbit/s	132 MB	8.78 Mbit/s	131.83 s
Task Scheduling, 500 tsk	163 MB	10.1 Mbit/s	188 MB	11.6 Mbit/s	132.52 s

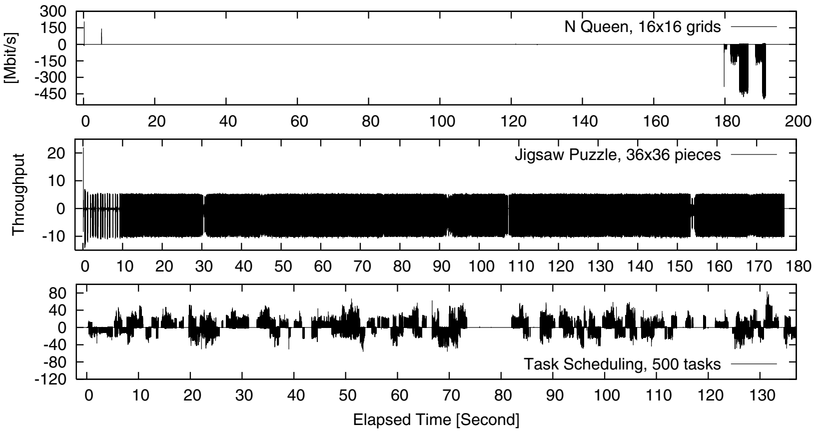
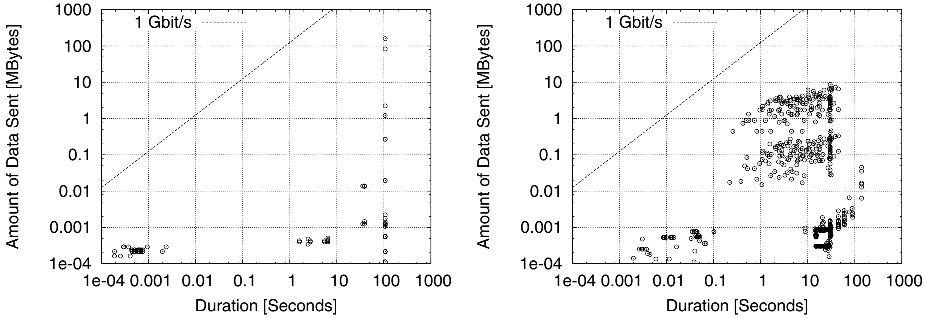


Fig. 2. Fluctuation of throughput in each of the applications. The X axis is the elapsed time in seconds after the applications start. The Y axis shows the throughput of 10 millisecond average in Mbit/s. Positive values on the Y axis indicate the traffic incoming to PC3 while negative values indicate the outgoing traffic

Figure 2 shows the fluctuation of the throughput of data transferred to/from PC3 for one instance of the 20 experiments described in Table 1. Note that both PC2 and PC4 showed the traffic pattern similar to PC3 for all the applications. We found that the patterns of the fluctuation across the scales of the problems were similar for all the applications while those across the applications were completely different. N Queen sends a huge amount of data near to the end of the process from the slave to the master (outgoing from PC3 to PC1). Jigsaw Puzzle continuously exchanges a small amount of data in a stable rate through its processing. Task Scheduling intermittently exchanges a large amount of data through its processing.

In addition, we analyze the feature of flows composing traffic generated by each of the applications. Since all the applications employ only TCP for their task communications, we define a flow as a set of packets transferred in a TCP connection, by direction, beginning with a SYN flag and terminating with a FIN flag. It is seen that in all the applications, multiple TCP connections often established in parallel through their processing.

Figure 3 shows the amount of transferred data and the duration on each of the flows in N Queen and Task Scheduling for one instance of the 20 experiments. Jigsaw Puzzle shows the features similar to N Queen described in Figure 3 (a). The amount of transferred data in flows and their durations differ depending on the type of the distribution process. For N Queen in Figure 3 (a), the flows are small in number, and some of them last for less than ten milliseconds, some last for around ten seconds, and the others last from the beginning to the end. The long-lived flows transmit a various amount of data from 100 bytes to more than 100 Mbytes. The huge amount of data transferred in the N Queen mast be carried by such the long-lived flows. For Task Scheduling in Figure 3 (b),



(a)N Queen (b)Task Scheduling

Fig. 3. Plots of the duration and amount of data transferred in each flow. The line is the boundary of plots which is equivalent to 1 Gbit/s

flows are large in number and diverse in length (duration time), which transmit various amounts of data.

3.2 Impact of Expanding Round-Trip Time

We investigate the influence of a long RTT on the completion time of each application. In our experiments, 1 to 32 milliseconds latencies are inserted into passing traffic for each way by the network emulator employed in Figure 1. The link bandwidth is configured to a sufficient value of 1 Gbit/s to focus only on the impact of the RTT to the application performance. The socket buffer of 16KB length is employed in N Queen and Jigsaw Puzzle, and 64KB and

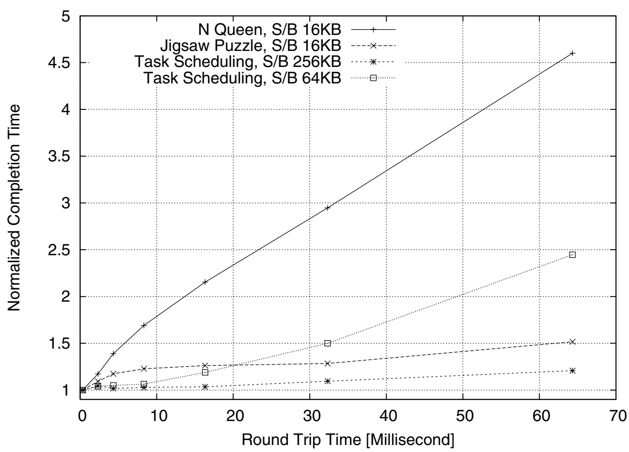


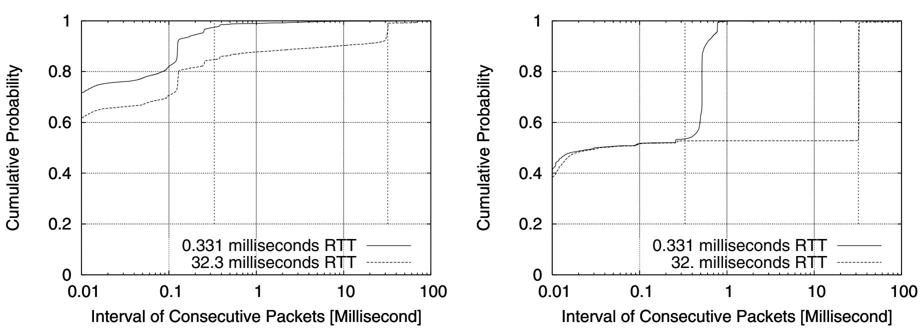
Fig. 4. Completion time influenced by RTT. The application-level performance deteriorates as the RTT increases

256KB are in Task Scheduling. Figure 4 shows the completion times of the each application influenced by the RTT. The completion times are normalized by that without any latency inserted. Each completion time is the average of results on 20 experiments.

Note that, for Jigsaw Puzzle, the completion time differs depending on which couple of communications between the master and slaves are influenced by the RTT: e.g., PC1–PC2 and PC1–PC3, PC1–PC2 and PC1–PC4, or PC1–PC3 and PC1–PC4. In Figure 4, we employ the worst case that the completion times are most influenced by the long RTT.

For all the applications, the application-level performance deteriorates as the RTT increases. For Task Scheduling, enlargement of the maximum size of sending window in TCP connections is very effective in reducing its completion time, which must come from the fact that the average sending window size should be equal to the product of the average throughput and the RTT in successively sending a large amount of data. N Queen is remarkably affected by the RTTs increase while Jigsaw Puzzle is not. To pursue the reason of the difference between the influence of the RTT on N Queen and that on Jigsaw Puzzle, we analyze intervals of consecutive packets in flows generated between PC1 and PC3.

Figure 5 shows the distribution of intervals of consecutive packets in flows generated between PC1 and PC3 for N Queen and Jigsaw Puzzle. For N Queen in Figure 5 (a), the distribution of packet intervals in case with a short RTT (0.331 milliseconds) shows that almost all the intervals are less than the RTT. This implies that almost all the packets are sent in a successive manner, not in an interactive manner. Two peak values of packet intervals are indicated; one is about 0.012 milliseconds corresponding to back-to-back data packets of 1500 bytes (more than 70 % of intervals), the other is about 0.1 millisecond. On the other hand, in case with the RTT (32.3 milliseconds), while there exist two similar peak values of intervals, the number of intervals in the most bursty case (i.e., back-to-back packets) decreases to 60 % and the intervals around the RTT increases instead. This indicates that a long RTT prevents the rapid growth of the sending window size, resulting in the decrease of the throughput.



(a)N Queen (b)Jigsaw Puzzle

Fig. 5. Distribution of packet intervals

For Jigsaw Puzzle in Figure 5 (b), the distribution of intervals also shows two peak values; one, which is more than 40 %, is roughly close to that of back-to-back packets of 1500 bytes (0.012 millisecond), the other is roughly close to the RTT (0.331 millisecond) where the packets may be sent in an interactive manner. In case of 32.3-millisecond RTT, the number of packets in the bursty case is unchanged, while the peak value corresponding to the RTT moves to the new RTT. The reason that the RTT affects only packets in an interactive manner is that a small amount of data, which doesn't exceed by the size of the current sending window, are frequently sent when Jigsaw Puzzle sends lamp data.

The information on such the sensitivity of the application performance for a long RTT will help us in path allocation for different applications in multi-path environments.

3.3 Impact of Limiting Bandwidth of Bottleneck Link

We investigate the influence of limiting bandwidth of the bottleneck link on the completion time of each application. In the experiments, the bandwidth of the bottleneck link varies from 80 Kbit/s to 1Gbit/s by using the network emulator in Figure 1. The RTT is the original short value without any additional latency to focus only on the impact of bandwidth restriction to the application performance. We employ the socket buffer of 16KB length in N Queen and Jigsaw Puzzle, and 256KB in Task Scheduling. We found that Task Scheduling with 64KB socket buffer showed the similar performance characteristics to that with 256KB on the restriction of the bottleneck link.

Figure 6 shows, the completion time roughly unchanges (within 1.1 normalized completion time) even if the bandwidth is reduced before reaching some upper-threshold for each application: e.g., 200 Mbit/s for N Queen with 16 × 16 grids, 70 Mbit/s for Jigsaw Puzzle with 36 × 36 pieces, and 200 Mbit/s for

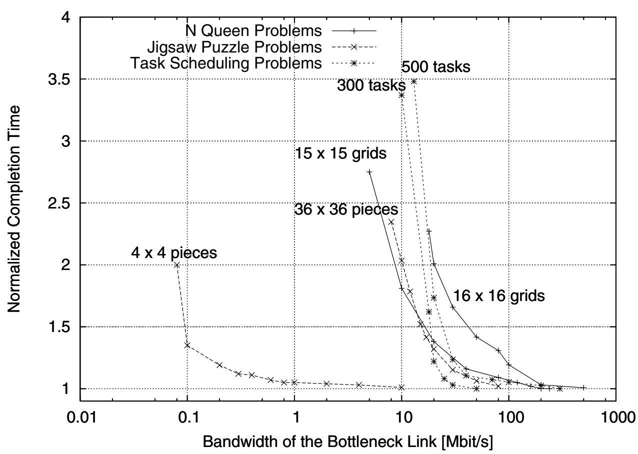


Fig. 6. Completion Time Influenced by Narrow Bottleneck Link

Task Scheduling with 500 tasks. Furthermore, the completion time abruptly increases (exceeding 1.2 normalized completion time) if the bandwidth is reduced after reaching some lower-threshold for each applications: e.g., 100 Mbit/s for N Queen with 16 $\times$ 16 grids, 30 Mbit/s for Jigsaw Puzzle with 36 $\times$ 36 pieces and 40 Mbit/s for Task Scheduling with 500 tasks. The order of such the thresholds across the applications (N Queen > Task Scheduling > Jigsaw Puzzle) is the same as that of the average throughputs generated by them in Table 1.

The information on such the thresholds of limiting bandwidth with respect to the application performance will help us to determine how a limited amount of network bandwidths should be allocated to an application.

4 Conclusion

In this paper, we have investigated the network-related characteristics of some typical distributed applications, focusing on the influence of the application traffic on the condition of network resources and inversely, that of the condition of network resources on the application-level performance.

We first have analyzed the characteristics of traffic generated by the applications that were classified into the Task-Framing or the Work Flow type distributed processing. It was found that all the applications increased their completion time and the amounts of transmitted data as the scale of their problem became larger while the relations between the amount of transmitted data and the completion time heavily depend on its applications. In addition from the analysis of flows, the communications of Task Scheduling (Work Flow type) consisted of a large number of flows with various durations and throughputs.

N Queen and Jigsaw Puzzle (Task-Framing type) had a relatively small number of flows, less than a hundred, and their flows lasted either for short durations less than ten milliseconds, for moderate durations around ten seconds, or for the application processing from the beginning to the end.

We secondly have analyzed how the performance of each application was affected by the condition of network resources. It has been shown that the sensitivity of the application completion time to RTT differed strongly depending on the applications. Note that the capability of enlarging the window in TCP connections could mitigate the performance degradation caused by a long RTT in an application sending a large amount of data in the successive manner. Then it has also been shown that the application completion time abruptly increased if the bottleneck bandwidth was limited to a value less than some threshold, which differed strongly depending on the applications.

Our future goal is to develop an application-aware network resource allocation by using the information on network-related properties of applications. Such kind of traffic engineering will determine which applications should be run simultaneously and which end-to-end path (among alternatives) should be allocated to an individual application, based on how the performance of each of the applications is affected by the condition of network resources and on how the traffic generated by these applications affects the condition of network resources. Our experiments suggest that an efficient network resource allocation is feasible by using such information from the view point of the application-level performance.

Acknowledgement

This work was supported in part by National Institute of Information and Communications Technology under the JGN II R&D Project, and in part by the Ministry of Education, Culture, Sports, Science and Technology, Japan, Grant-in-Aid for Scientific Research on Priority Areas (16016271).

References

1. I. Foster and C. Kesselman: The GRID Blueprint for a New Computing Infrastructure. Morgan Kaufmann Publishers, (1998)
2. I. Foster, C. Kesselman, and S. Tuecke: The Anatomy of the Grid: Enabling Scalable Virtual Organizations. International Journal of Supercomputer Applications, 15(3), (2001) 200–222
3. A. Elwalid, C. Jin, S. Low, I. Widjaja: MATE: MPLS adaptive traffic engineering. Proc. of the Infocom Anchorage (2001) 1300–1309
4. T. Guven, C. Kommareddy, R. La, M. Shayman and S. Bhattacharjee: Measurement Based Optimal Multi-path Routing. Proc. of the Infocom, Hong Kong, March, (2004)
5. ICPC Asia Regional Contest, Japan Hakodate. http://www.fun.ac.jp/icpc/
6. Kasahara Laoratory, Waseda University. http://www.kasahara.elec.waseda.ac.jp/schedule
7. H. Koide, and Y. Oie: A New Task Scheduling Method for Distributed Programs which Require Memory Management in Grids. Proc. Proposed SAINT2004 Workshop 8: High Performance Grid Computing and Networking, Tokyo (2004) 666–673
8. R. Buyya: High Performance Cluster Computing: Programming and Applications. Volume 2, Prentice Hall PTR (1999)
9. Empirix. http://www.empirix.com/

A Client-Side Workflow Middleware in the Grid[1]

Ying Li[1,2], Qiaoming Zhu[1], Minglu Li[2], and Yue Chen[1]

[1] Computer Science and Technology School, Soochow University, Suzhou 215006, China
[2] Department of Computer Science and Engineering,
Shanghai Jiao Tong University, Shanghai 200030, China
{ingli, qmzhu, cheny}@suda.edu.cn
{liying, li-ml}@cs.sjtu.edu.cn

Abstract. Grid computing is becoming a mainstream technology for sharing large-scale resources, accomplishing collaborative tasks and integrating distributed systems. With the development of the Grid technology, the Grid will provide the fundamental infrastructure not only for e-Science but also for e-Business, e-Government and e-Life. The workflow management system in the Grid is important to support such Grid applications. This paper proposes a framework of client-side workflow middleware, puts the emphasis on the transaction management and service discovery in workflow. The transaction in the workflow includes atomic transaction and compensation transaction. The coordination of these transactions in workflow is introduced in detail.

1 Introduction

Grid based computational infrastructure is an ideal computing platform to meet large-scale resources and heterogeneity system [1]. Grid computing is becoming a mainstream technology for sharing large-scale resources, accomplishing collaborative tasks and integrating distributed systems [2].

In the past few years, the Computational Grid and Data Grid have received much attention, the server side applications, toolkits, middlewares are more available for the Grid, however the client side middleware for end Grid user is less concerned. For the workflow management system in Grid, much attention is put to e-science fields; little work has been done for the client-side workflow management system, which seems more close to our daily life than the Computational Grid.

Although the Grid society has no clear definition for the information Grid, knowledge Grid, but the conception of such Grid is now widely accepted. The client-side workflow management middleware plays an important role in Grid: the e-business, e-government in Information Grid need the workflow to coordinate varieties tasks and activities, the Grid user needs a way to compose existing Grid Services into a new business services. In this paper, a client-side workflow management middleware is proposed in the Grid.

[1] This paper is supported by 973 project (No.2002CB312002)of China, ChinaGrid Program of MOE of China, and grand project of the Science and Technology Commission of Shanghai Municipality (No. 03dz15026, No. 03dz15027 and No. 03dz15028).

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 19–26, 2005.

2 Background and Related Works

According to the Workflow Management Coalition, workflow is concerned with the automation of procedures where information and tasks are passed between participants according to a defined set of rules to achieve, or contribute to, an overall business goal [3]. The traditional workflow management system (WFMS) is mainly concerned with the automation of administrative and production business processes. These processes coordinate well-defined activities that execute in isolation, i.e. synchronize only at their start and terminate states [4]. Currently a lot of workflow models were put forward to meet the requirement of the distributed, cross enterprise application integration (EAI).

Recent years, Web Services became an important technique to serve e-business. The service based workflow protocol was designed such as Web Services Flow Language (WSFL) [5], Web Services for Business Process Design (XLANG) [6], Business Process Execution Language for Web Services(BPEL4WS)[7].

The Workflow Framework for Grid Services (GSFL) was put forward in article [8]. The Grid Services Flow Language is an XML based language that allows the specification of workflow descriptions for Grid services in the OGSA [1] framework. It has been defined using XML Schemas.

3 The Requirement of Client-Side Workflow in Grid

In article [7] the author analyzed the existing Web Services workflow specification and put forward the GSFL to avoid some disadvantages in Web Services workflow. However, GSFL puts emphases on the effectiveness of exchanging large amount of data in Grid, the research is aimed at the server side workflow. According to the workflow in GSFL, we believe that in Client-side workflow environment, the following additional condition must be taken into account:

- Transaction support

  Traditionally, transactions have held ACID properties. However, in Grid service environment, the coordination behaviors provided by a traditional transaction mechanism are sufficient if an application function can be represented as short-lived transaction performing stable state changes to the system. But they are less well suited for structuring "long-lived" application functions, and less capability to handle the coordination of multiple services. Long-lived transactions may reduce the concurrency in the system to an unacceptable level by holding on resources for a long time. If such a transaction aborts, much valuable work already performed in the workflow must be undone. Therefore, in a loosely coupled and service autonomic gird service workflow environment, it is inevitable that more relaxed forms of transactions, which don't strictly abide to the ACID properties, will be required.
- Grid workflow schedule and coordinate

  Compared with traditional workflow schedule where each service is statically created, the scheduling of workflow engine in Grid is more complex. Except the static scheduling which workflow engine allocates the resources according to the process description statically binding the resources, there exists dynamic scheduling, which engine dynamically allocates the resources at runtime.

- Easy composed and used by end user

  Currently the research of the Grid workflow is mainly in scientific field [10] [11]. The workflow system could be beneficial for modeling and managing scientific processes of experimental investigation, evidence accumulation and result assimilation [11]. Processes themselves can then be reused, shared, adapted and annotated with interpretations of quality, provenance and security.

With the growth of the e-business in the Grid and the development of information Grid, user-composed or client-side workflow system must be taken into account. In the testbed of ShanghaiGrid [12], end users need a tool to published their Grid applications or compose existing Grid Services into a new business services to extend its business values. Under such circumstance, the Grid Computing Environment (GCE) must provide a way to let end user easily compose workflow, so client workflow management system is more important. Users, organizations and other roles in the Grid need a facility to assemble the Grid Services to form business logic or a new Grid Services in a workflow; such workflow is a dynamic process and maybe change frequently. The benefits of client-side Workflow in Grid include flexibility, adaptability and reusability.

The research is to design a client-side workflow management middleware based on the GCE. It should include such high level feature: Grid Transaction support, Grid based services discovery, Rule-based workflow, GUI-enabled workflow composing tools.

4 Agent Based Transaction Management in Grid Workflow

Based on our previous research on the transaction model in Web Services and Grid [13][14], Fig.1 shows the transaction architecture based on agent in the Grid workflow system. This architecture can handle two types of transaction in the Grid workflow system, which can be appointed by users or administrator:

- Atomic transaction (AT). AT is used to coordinate activities having short-lived application and executed within limited trust domains.
- Compensation transaction (CT). CT is used to coordinate activities having long-lived application. In order to improve the concurrency, a compensation model must be applied in CT.

4.1 Atomic Transaction (AT) Coordination in the Workflow

AT holds the ACID properties. In some condition, especially in short-lived application, AT will provide a more efficient way to accomplish workflow; meanwhile it has consistent failure and recovery semantics. The AT coordination in the workflow includes following steps:

- Initiation: the Agent in workflow engine creates the transaction management and sends the Coordination Context (CC) message which includes necessary information

to create a transaction such as transaction type AT, transaction identifier, coordinator address and expires to the Grid Services participate in the workflow. The lifecycle of CC will extinct till the transaction finished.
- Preparation: the coordinator send prepare information to the Grid Services' agent, and each service will first attempt to reserve all the needed resources. If successes, agent returns the success message to the coordinator, otherwise, returns the failure message.

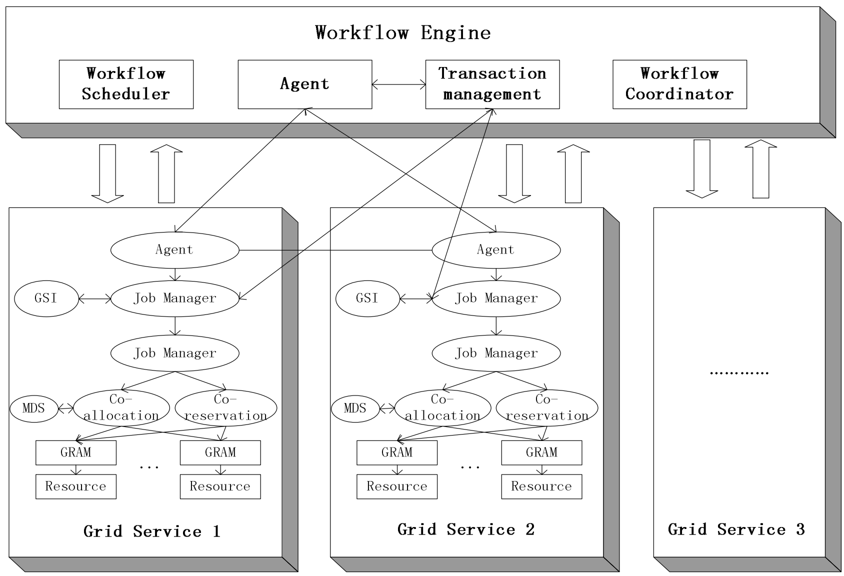


Fig. 1. Transaction architecture for Grid workflow

- Execution: P_i (i=1..n) is the ordered execution activities participate in the workflow, P_1 is the start activity and P_n is the end activity. R_i(i=1..n-1) is the rule to determine if the return value of P_i suits certain condition. At certain time point, $P_1 \ldots P_{k-1}$($1<=k-1<n$) are already executed, and P_K is to be started. The P_k first allocates the resource it reserved, then executes, records the transaction in log in order to recover later from possible failure. After execution, P_k remains uncommitted status and returns the results to the workflow engine; the engine compares the results with R_K. If the results of P_K satisfy the R_K, then P_{K+1} will execute next. Otherwise, the workflow stops executing P_{K+1}. When workflow Engine receives the results from P_k, it sends confirmed message to P_k , if P_k does not receives the confirmed message in a certain period of time, it will resend the results for N times. Meanwhile, the workflow Engine can query P_k status if there is no results after certain time.
- Committion: within the expiration time, if P_n(the last activity in workflow) is successfully executed and returns the results to workflow engine, the workflow is about to commit all the transactions. Otherwise cancel all the transaction, roll back any Pi to previous states. The commit step is like the traditional two-phrase commit.

If P_k has sub-workflow, it will apply above mechanism recursively.

4.2 Compensation Transaction Coordination in the Workflow

AC applied in workflow system has some disadvantages: it reserves the resources all the activities needed until the transaction finished. In long-lived application, it is impossible to use AC as a workflow model. Fault-tolerance and compensation are required to support such application, the steps are similar to the AC in the workflow, but have following important differentiations:

- The Grid Services reserves and allocates resources only when it's invoked.
- In execution step, every activity has a timestamp. At certain time point, P_1… P_{k-1} ($1<=k-1<n$) are already executed, and P_K is to be started. Task Manager(TM) starts the P_k through agent and wait the response from P_k . P_k reserves and allocates the resource, then executes, records the transaction in log in order to recover later from possible failure. After execution, P_k immediately commits, if successful commits, it generates corresponding compensation transaction and returns TM with Committed message, which contain the results, to the Workflow Engine. Otherwise it automatically rollbacks operations taken previously and returns Aborted messages. After sending commit information, P_k waits for confirm information from TM. If workflow engine receives committed message it compares the results with R_K. If the results of P_K satisfy the R_K, then P_{K+1} will execute next. Otherwise, the workflow stops executing P_{K+1}, sending abort message to P_i ($1<=i<=k$) which has already been committed. After that, the TM sends confirm information to P_k . If In certain time P_k does not receive confirm message, it will resend for N times, if there still has no response from TM, P_k automatically recovery using compensation.
- Compensation generation: based on pre-defined rules stored in rule database in GCE, Database event, Transaction Event, human anticipate pattern and system environment, the Agent generates compensation.
- Recovery: The compensation option is used to recover the committed transaction for committed activities, meanwhile rollback option is used to the abort the transaction before commit.

4.3 Non-compensation Transaction Coordination in Workflow

For non-compensation transaction existing in workflow, the coordination of the transaction became more complex. Currently, if any activity in workflow is non-compensation transaction, the workflow engine treats that workflow as atomic transaction workflow to avoid the complexity of workflow schedule.

5 The Design of the Workflow Management System on the Grid

5.1 The Framework of the Workflow Management System on the Grid

Fig.2 shows the framework of the workflow, it includes:

- Workflow Client Tools (WCT).

WCT gives a GUI based interface to let uses or agents to assemble Grid Services in which information was retrieved from UDDI.

- Workflow Repository Services (WRS)
  The user defined persistence workflow is also regarded resource and can be reused by others. The composed workflow information is put in UDDI and stores it permanently in the Workflow Repository Services. The WRS provides inserting, deleting, updating, or querying service to be invoked by the Grid users.
- Workflow Monitor Services (WMS)
  The WMS can monitor the execution of the workflow return the status.
- Workflow admin Services (WAS)
  The WAS mainly includes role management, audit management and so on.
- Workflow Engine Services (WES)
  The WES is the core services in the workflow management system. The Workflow Engine is responsible for creating, assigning, controlling the activities (tasks), and deciding in each moment the next action to be performed [15]. It invokes the Grid Services as the tasks. The transaction management is discussed in section 3.

The framework that the Grid based workflow management system is like tradition one [16,17,18], but all functions should be wrapped with Services.

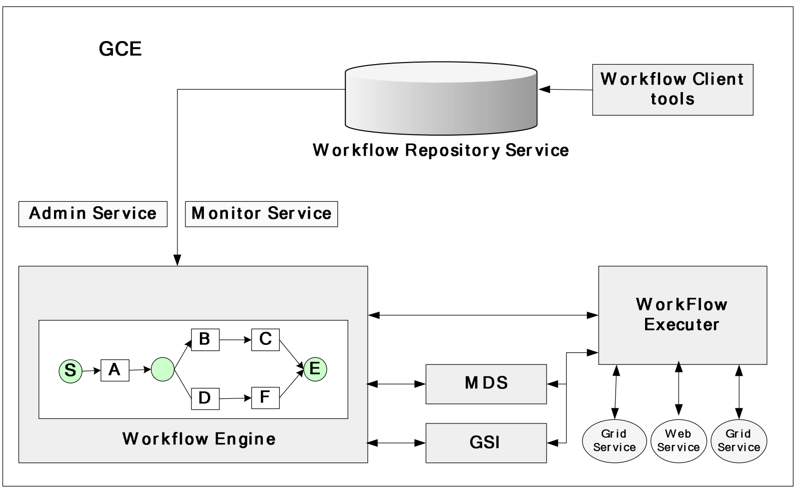


Fig. 2. The framework of the workflow management service

5.2 Service Discovery of the Workflow

As mentioned before, the user-defined workflow can be regarded as grid services in the GCE, and can be regarded as a sequence of Grid Services running under one transaction. In GCE we assume that each Factory and Grid service instance must register its GSH (Grid Service Handle) with Registry. Mapping relationship between GSH (Grid Service Reference) to GSR is registered in HandleMapper. And the Factory Service should be persistence service which means the factory service will automatically create when its container starts up.

Grid user (Client) looks up Grid Services through the UDDI according to the services descriptions, the UDDI returns the Grid Services' URLs. Fuzzy search arithmetic of the schema is designed that if there does not exist entire match, the UDDI will return a list of Grid Services to the user and let user choose one.

GCE judges if the URL is a workflow-based Grid services. If it is, the Agent creates the WES and WRS instance and WES gets the workflow information from the WRS.

The WES picks up the first task in the tasks-list and gets its URL, using this URL the WES discovers the GSR of a Grid service instance. If the Grid Service instance has been created and in valid status, the WSE looks up the Registry and gets GSR according to the GSH by using querying HandleMapper. If GSR does not exist, the WSE finds the GSH of Factory in Registry, creates a Grid service using the factory interface, which gets the GSH and GSR and then registers them in the Registry.

When WES gets the GSR, it has enough information to bind it then receives its results. And then using that result as input to do the same step mentioned above until the entire task in task list have been done.

6 Conclusions and Future Work

In this paper, we analysis the requirement of client-side workflow in Grid and a client-side workflow management middleware is purposed based on our pervious work on ShanghaiGrid testbed about transaction , workflow and so on; the transaction and service discovery in workflow is discussed in detail. The Client side workflow Engine Service we proposed is a lightweight engine. It does not contain performance evaluation, dynamic scheduling, fault torrent and so on. And the workflow management system is a centralized system. Such matter will be taken into account to improve the workflow management system. The none-compensation transaction in workflow should be further studied to improve the coordinate of schedule.

References

1. I. Foster, C. Kesselman, J. M. Nick and S. Tuecke. The Physiology of the Grid-An Open Grid Services Architecture for Distributed Systems Integration. June, 2002.
2. I. Foster, C. Kesselman, J. Nick, S. Tuecke. Grid Services for Distributed System Integration. Computer, 35(6), 2002.
3. Object Management Group, "Workflow Management Facility Specification," Version 1.2
4. Daniela Grigori, Francois Charoy, Claude Godart , LORIA, INRIA Lorraine. Flexible Data Management and Execution to Support Cooperative Workflow: the COO approach, Third International Symposium on Cooperative Database Systems for Advanced Applications
5. http://www-3.ibm.com/software/solutions/webservices/pdf/WSFL.pdf
6. http://www.gotdotnet.com/team/xml_wsspecs/xlang-c/default.htm
7. http://www-106.ibm.com/developerworks/library/ws-bpel/
8. S. Krishnan, P. Wagstrom, and G. V. Laszewski. GSFL: A Workflow Framework for Grid Services. Argonne National Laboratory. Aug 2002.

9. D. J. Taylor, "How big can an atomic action be?", Proceedings of the 5th Symposium on Reliability in Distributed Software and Database Systems, LosAngeles, January 1986,
10. H. P. Bivens, "Grid Workflow", Grid Computing Environments Working Group,2001.
11. Junwei Cao, Stephen A. Jarvis, Subhash Saini and Graham R. Nudd, GridFlow: Workflow Management for Grid Computing, CCGrid 2003.
12. Ruonan Rao, Baiyan Li, Minglu Li, and Jinyuan You The Delivery and Accounting Middleware in the ShanghaiGrid, Proceeding of the Second International Workshop on Grid and Cooperative Computing (LNCS 2975), Shanghai. December, 2003.
13. Feilong Tang, Minglu Li, Jian Cao, and Qianni Deng. Coordinating Business Transaction for Grid Service. Proceeding of the Second International Workshop on Grid and Cooperative Computing (LNCS 2975), Shanghai. December, 2003.
14. Tang Feilong, Li Minglu, Cao Jian, Rao Ruonan and Qian Qi.Technology of Enterprise Application Integration Based on Web Services. Proceeding of GCC 2002. December, 2002.
15. J. A. Espinosa, A. S. Pulido IB (Integrated Business) :A Workflow based Integration Approach, Proceedings of the 35th Hawaii International Conference on System Sciences – 2002.
16. Khalid Belhajjame, etc . A flexible workflow model for process-oriented applications. Proceedings of the Second International Conference on Web Information Systems Engineering (WISE'02).
17. A.Goh, et.al. ECA Rule-based Support for Workflows, Artificial Intelligence in Engineering,15(2001), pp37-46.
18. Jian Cao Minglu Li Shensheng Zhang Qianni Den, Composing Web Services based on Agent and Workflow, Proceeding of the Second International Workshop on Grid and Cooperative Computing.

General Architecture of Grid Framework with QoS Implementation

Vit Vrba, Karol Molnar, and Lubomir Cvrk

Brno University of Technology, Department of Telecommunications,
Brno, Purkynova 118, The Czech Republic
{vrbav, molnar, cvrk}@feec.vutbr.cz

Abstract. The sophisticated distribution of computing some difficult mathematical tasks between geographically scattered computers, connected into the so-called grids, offers to users the potential of using a processing power several times higher than that of a supercomputer or a cluster at a much lower price. A crucial problem of recent applications requiring the grid-based solution is that they are designed to solve one specific problem. The aim of this paper is to propose a universal Grid framework that will use a sophisticated aggregation method for distributed data processing. This framework should realize all repeating programmer operations automatically – e.g. data transfer and validation, security, authorization, etc. The user of this system should only insert desired algorithm and data for processing and the system will be able to solve any parallelized task automatically without additional programming.

1 Introduction

Currently, there is an increasing number of tasks which cannot be processed by common computers. The demands of research and science define new algorithmized tasks continually and these tasks cannot be solved by the computing power of a single workstation. It is necessarily to use expensive supercomputers or aggregate computing power for solving these tasks.

The aggregation of computing power can be realized in two different ways. The first one, historically older and very expensive solution, is to develop a multiprocessing supercomputer. This centralized system, with exactly defined requests on the function of each component, covers complex problems from parallel use of processors to programming the specialized software. This solution is proprietary in most cases and there are a lot of financial and technological barriers that disallow effective usage.

The second method is aggregating the power of a few servers into clusters, when it is possible to solve the task on multiple computing units in parallel. This is the most widely used method today.

1.1 Grid Networks

The idea of using clusters was later generalized in constructing Grid networks, which can aggregate the power of not only servers but common workstations too. The Grid

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 27–33, 2005.

is in principle cluster secured to be able to work in the unprotected space of the Internet. The whole system consists of geographically scattered computers, which are being interconnected dynamically towards being able to offer unified information about its free capacity (e.g. computing cycles, free disk space, etc.).

Today there are a lot of subjects supporting grid research in the hardware area (HW, IBM), for the middleware or the business database grids (Oracle). On the basis of these experiences there were written recommendations for a grid development called “Standard Open Grid Services Architecture”.

1.2 Current Grid Problems

A huge problem of current applications using the Grid technology is the fact that they are programmed to solve one specific task. For a different task it is necessary to program the whole system again, including e.g. network communication, user authentication, encryption, etc. It is also impossible to use an algorithm already programmed for a single computer, because it is not optimized for parallel computing. Another difficulty is the fact that tools for grid-usage monitoring, accounting or even for security have not been developed. These disadvantages are very problematic mainly when the grid is used by several departments or even companies.

The idea of aggregation power of common computers connected over a communication network is not an original. There exist some systems in computer networks that are focused on processing distributed computing. But each of them has some troubles. Solutions like SETI@home, Folding@home, Genome@home, breaking the DES algorithm are programmed for computing one specific task. On the other hand there are projects like MOSIX or BEOWULF that offer more flexibility from the viewpoint of algorithm but they can be used for only one specific operation system, they require a complex support and very often they require the whole computing power of the machine.

The main disadvantage of both mentioned solutions is the impossibility to *solve any computing task without large additional costs*. But these aspects offer the biggest potential of distributed tasks.

2 Universal Grid Framework Architecture

The reasons for the above specification are clear – design an open architecture enabling users to substantially speed-up the development work by reducing it to a mere implementation of a concrete algorithm. This universal architecture has to ensure the aggregation of the power of common workstations for solving generally any parallelized tasks.

This architecture will not work as a stand alone system. It will be only a general application framework where it is very simple to input algorithm and data for processing a particular task. This framework ensures an automatic distribution of the algorithm from server (node) to workstations (points or execution units) and consequently collects the processed outputs, sorts them and presents results.

Demands on Universal Grid Framework Architecture

We determine these important demands on the planned architecture:

- Internet-based clustering of points,
- federation of clusters to create hierarchical, cooperative grids,
- web services interface supporting a grid job model for cross-platform interoperability,
- one or more interoperable nodes in the network,
- client installation on unlimited number of points,
- transparent access to distributed resources,
- framework independence from the processing task,
- high effectivity of using the distribution power,
- definition of maximal using computing capacity of workstation,
- definition of maximal storage limit of workstation,
- secured and verified communication between node and points,
- elaborated verification of the authenticity of computed data,
- Quality of Services in Grid network should be implemented.

Advantages of a Grid Based on the Framework

- facilitation of algorithm implementation,
- faster and cheaper grid building,
- easy development,
- secured communication realized automatically,
- user verification realized automatically,
- decentralized data saving (distributed backups, archiving, data stores, etc.).

2.1 Concept of Grid Framework

Due to our demands we could not use the existing Grid open source projects (eg. Globus Project) or grid solutions offered by JAVA or .NET. Te main argument for refusing these projects is speed – interpreted languages are slow or at least they are not as fast as compiled code is. And what is more – only JAVA and .NET frameworks occupy some computing power and need at least Pentium III with 256MB RAM for their smooth working.

We had to design our own concept of Grid framework that covers the server and client specification, communication node-point protocol and methods for computing parallel threads.

This Grid framework respects OGSA (Open Grid Service Architecture) and uses also Web Service technologies – SOAP for communication and WWW for monitoring the execution units. As can be seen in Figure 1, it consists of 3-layer architecture described below.

Execution Unit

Execution unit is a final client application that receives the algorithm and data from the node. The workstation then computes the task and sends the result back to the node. Our client was written in Microsoft C++ compiled with Intel C++ Compiler. This 3MB application is super-effective for processing computing tasks, easily installable and configurable (also remotely) and has extended options of setting the

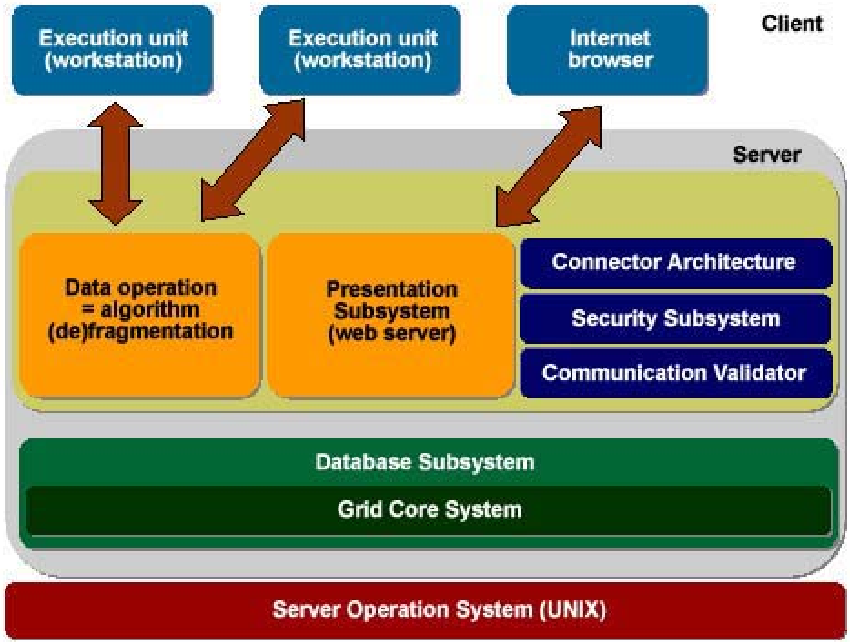


Fig. 1. Grid framework 3-layer architecture

user rights. It can be set up whether users can start or stop the computing process, change the capacity of processor, memory or storage, etc. At the present, the client application is available for the Windows platform just now but we are preparing also the UNIX version.

Grid Core System
This daemon ensures subsystem communications for the server part of framework. The grid server was written in JAVA for the UNIX operation system.

Database Subsystem
Our framework is not dependent on a single type of DBMS (Database Management System). We have found it is an advantage to use relation database systems only for some computing tasks, while sources and results of others tasks meet the object-oriented, hierarchic database systems or even the XML format. Due to this requirement we had to write a universal database driver which defines a generalized database language that is later translated into the native DBMS language (e.g. SQL Data Manipulation Language). Because of huge database capacity we also implemented support for data mining.

Data Operations
This subsystem uses the SOAP technology for data transfer (communication principles are described later) and ensures

- store user algorithm
- store data for proceeding

- store processed data
- deliver the algorithm
- send data to clients
- receive data from clients
- security
- data validation

Presentation Subsystem
Remote server administration is realized through apresentation subsystem. It allows setting server parameters, properties of the grid network and several clients. The administration is accessible via the web and allows monitoring the state of computing task as a whole but also monitoring serveral clients.

3 Communication Principles

A process of data computing must deal with the following issues:

- identifying suitable Grid nodes (that possess the required computing power, memory, storage, etc.),
- transferring the algorithm (input resources and dependent libraries) execution units,
- transferring data to execution units,
- starting the remote job and monitoring its execution,
- transferring the results from the remote execution unit back to the node.

3.1 Quality of Service for Grid Networks

The environment in which an application resides is very important. Grid clients (execution units) will present their requirements at component development-time, assembly-time, deployment-time or execution-time. Quality of Service for Grid could be defined in terms of how well these requirements can be met.

As was said above, our universal architecture respects a widely adopted Open Grid Services Architecture (OGSA) that provides a web-service-based development environment for developing grid applications. Although OGSA relies on Internet standards while making the platform services open for loose coupling, it does not support any QoS issues such as QoS specification, management, and is limited in performance optimization.

Grid users may wish to have fine-grained control of quality of service (QoS) guarantees in the network in order to allow timely data transfer in a distributed application environment. Internet Protocol (IP) based networks, and the Internet itself will be used to allow communication across Grid Systems. The IP and the Internet were never designed to handle QoS-sensitive traffic and so the Internet community must evolve the network and enhance the Internet protocols in order to cater for the needs of these new and demanding applications. Although there exist some solutions for IP protocol (e.g. INTSERV, DIFFSERV) they are not suitable for Grid networks.

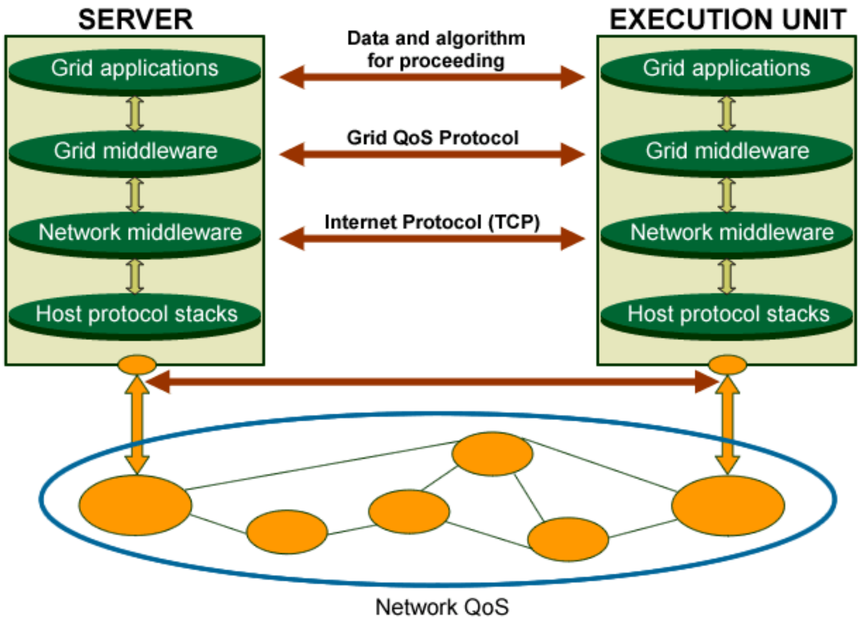


Fig. 2. Grid QoS Protocol architecture

3.2 Concept of Grid QoS Protocol

As we could not use QoS on the network layer of the OSI model for our universal architecture we had to define our own Grid QoS protocol. As shown in Figure 2 this protocol is based on Internet Protocol (application layer) and uses TCP ports. This high-performance network protocol ensures:

- endpoint-A, endpoint-B - the IP addresses of the two end-points of the reservation,
- directionality: whether the reservation is uni-directional or bi-directional,
- resource reservation and allocation of execution unit computing capacity,
- access control, security, accounting & billing,
- admission control, policing, and scheduling,
- encrypting data,
- traffic shaping, bandwidth, buffer management, etc.,
- monitoring support.

4 Conclusion

In the near future, the Computational Grid will become an important and powerful computing platform in both the scientific and commercial distributed computing communities for the execution of large-scale, resource-intensive applications. The goal of our grid-framework solution is to aggregate collections of shared, heterogeneous, and distributed resources to provide computational “power” for parallel application in a fast and cheap way.

Using the Grid Framework can greatly reduce expenses and the implementation time for any subject that plans to solve sophisticated computing tasks. Quality of Services is particularly important in utility Grid networks and it provides a new opportunity to further R&D in networking.

References

1. I. Foster, C. Kesselman, The Grid: Blueprint for a New Computing Infrastructure, Morgan Kaufmann Publishers, ISBN 1558604758
2. I. Foster, C. Kesselman, J. Nick, and S. Tuecke. ”Grid services for distributed system integration”. Computer, 35(6), 2002.
3. M. Baker, R. Buyya and D. Hyde, Cluster Computing: A High-Performance Contender, IEEE Computer, July 1999
4. F. Berman, A. J. G. Hey and G. Fox, Grid Computing: Making The Global Infrastructure a Reality, Wiley, ISBN 0470853190
5. W. P. Petersen and P. Arbenz, Introduction to Parallel Computing, Oxford University Press, 2003
6. T. Mengotti, W. P. Petersen and P. Arbenz, Distributed computing over Internet using a peer to peer network, September 2002
7. R. Staehli; F. Eliassen. QuA: A QoS-Aware Component Architecture, Technical Report Simula 2003-13, Simula Research Laboratory, 2002.
8. J. Walpole, C. Krasic, L. Liu, D. Maier, C. Pu, D. McNamee, and D. Steere, Quality of service semantics for multimedia database systems, Proc. Data Semantics 8: Semantic Issues in Multimedia Systems IFIP TC-2 Working Conference, Jan. 1999, pp. 393-412.

Centralized Versus Distributed Re-provisioning in Optical Mesh Networks

Chadi Assi[1], Wei Huo[1], and Abdallah Shami[2]

[1] Concordia Institute for Information Systems Engineering, Concordia University
[2] Department of Electrical and Computer Engineering,
University of Western Ontario
{assi, w_huo}@ciise.concordia.ca
ashami@eng.uwo.ca

Abstract. Significant progress has been made towards making optical networks 100% restorable in the event of single link failures through protection schemes with preplanned spare capacity. Currently dual failures are considered not uncommon and finding shows that designs offering complete dual failures restorability require more than double the amount of spare capacity. In this paper, we study the impact of re-provisioning on improving the overall network robustness and we compare two different re-provisioning schemes under both centralized and distributed implementations. We show that under distributed implementation, network robustness degrades due to excessive contentions and accordingly we propose a solution to mitigate the impact of contentions. We evaluate the performance of our proposal through simulation experiments.

1 Introduction

Significant progress has been made towards making optical networks resilient in the event of single link failures. Protection schemes with preplanned spare capacity [1] have been extensively studied in optical mesh networks where for each admitted connection two link disjoint paths are provisioned; a primary path with working capacity and a secondary path with protection capacity [2]. Protection capacity can either be dedicated or shared among multiple connections whose primary paths are physically disjoint and in the event of a failure along the primary path, the connection is rerouted to its secondary path [3, 4].

Given the increase in the size and complexity of today's networks, dual failures become increasingly probable. Dual failures can dramatically disrupt the services offered by the network if appropriate precautions are not implemented. Hence, designing recovery algorithms to protect against such failures is a paramount concern. To date, various efforts have already addressed the problem of routing connections under dual failures assumption, and findings show that designs offering complete dual failures restorability require more than double the amount of spare capacity [5, 13].

In order to avoid this excessive deployment of extra spare capacity in the network, capacity *re-configuration* after the occurrence of and recovery from the first failure has been proposed [6-10]. After the occurrence of the first failure, all failed connec-

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 34–43, 2005.

tions are restored from their working into their protection paths. Hence, upon complete recovery, shared protection capacity along active protection routes can no longer be shared. As a result some of the connections in the network will become unprotected and therefore will increases the network vulnerability to a subsequent failure.

Re-provisioning provides a mechanism by which one can find and allocate new protection capacities for these newly unprotected connections without a priori knowledge of the location of the second failure. A backup re-provisioning algorithm for handling multiple failures is recently presented in [10] and comprehensive studies indicate that re-provisioning can dramatically lower network vulnerability. Similarly, others have considered *pre-emptive* network re-provisioning schemes [8,9] whenever the second failure is assumed to occur after recovery actions are taken for the first failure but *before* the actual failed link itself is restored; overall findings show a notable improvement in the level of network vulnerability as well as recovery ratios.

In this paper we study the benefits of capacity re-provisioning, particularly on improving network robustness in optical shared mesh network. We assume two independent link failures, where the second failure occurs after the first failure is recovered from, but before it is physically repaired. A critical objective for re-provisioning, however, is to reduce the total number of connections that have to be re-provisioned. Here the motivations are twofold: (1) to reduce management overheads in provisioning a large number of connections, and (2) to lower reservation contention between multiple unprotected connections trying to establish backup capacity. The latter may result in increased blocking rates for re-provisioning, which in turn will increase vulnerability to subsequent failure(s). We present and compare the performance of two different re-provisioning schemes under both centralized and distributed implementations. We show that re-provisioning mitigates the impact of double-link failures and dramatically improves the robustness in a network that is designed to achieve 100% restorability under only single link failures. Moreover, we show that the proposed re-provisioning scheme outperforms the conventional scheme; that is under the same failure circumstances, our scheme re-provisions fewer connections than the conventional approach (i.e., reduced overhead and contentions) and protects more (i.e., better robustness). We also show that under distributed implementation the performance of both re-provisioning schemes degrades. This is due to the fact that right after recovery from the first failure, contentions among multiple unprotected connections simultaneously attempting to reserve protection capacity may occur, therefore leaving some of these demands unprotected. We propose a new technique to cope with the adverse effects of contentions by allowing unsuccessful connections to reattempt re-provisioning. We show that reattempting substantially improves the network performance when distributed re-provisioning is implemented. The rest of the paper is organized as follows. In section II we present the problem statement and network re-provisioning in section II. Section III presents the centralized and the distributed implementation of the re-provisioning algorithm. Section IV presents performance evaluation and comparisons and finally we conclude in section V.

2 Network Reprovisioning

2.1 Problem Statement

The objective of this paper is to study the benefits of capacity re-provisioning on improving the robustness of a network that, under normal conditions, is designed to only protect against all single link failures. When a failure occurs, all connections whose working paths are affected by that failure are re-routed on their corresponding protection paths [2, 3, 4]. Connection recovery usually requires source node notification and recovery signaling to configure the protection resources (e.g., wavelengths and cross connect switches) along the backup route [4]. However, since these protection resources may also be shared with other unaffected connections, these connections may become unprotected and vulnerable to the next failure [10]. Fig. 1 shows an illustrative example with three connections (A-H, C-G, and D-F). A working lightpath (*wi*) and a physically disjoint backup lightpath (*bi*) for each connection are initially provisioned. The protection resource wavelength λ1 on link <D-E> is shared between *b1* and *b2* since their corresponding working paths (*w1* and *w2*) are link disjoint. Upon the failure of link <B-F>, all working connections that are routed through that link are re-routed onto their corresponding protection paths. This in turn yields a set of *unprotected* connections, which increases the vulnerability to a second failure.

Overall, to summarize these unprotected connections, one can classify them into three categories:

1) *Indirectly Affected Connections*: During recovery, shared protection resources are activated by the failed connections and this may cause unaffected connections (whose backup lightpaths share these protection resources) to become unprotected. For example, *w2* in Fig. 1 is unprotected since *b1* is activated and *b2* can no longer share spare capacity with *b1*.
2) *Directly Affected Working Connections*: A connection that is re-routed to its backup route is still vulnerable to another failure on its protection route and hence is no longer protected (e.g., *b1* is unprotected since it loses its primary).
3) *Directly Affected Backup Connections*: Connections whose protection routes have failed due to the first failure become unprotected (e.g., *w3* is unprotected since it loses its backup path).

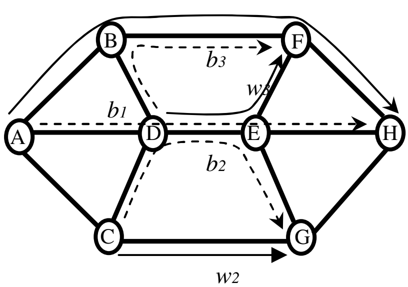


Fig. 1. Sample network and connections

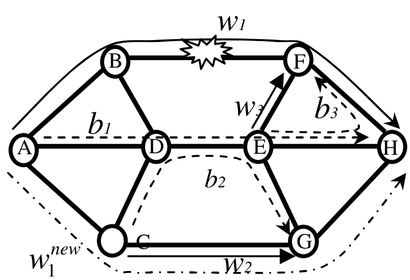


Fig. 2. Example for re-provisioning

Clearly, increased numbers of unprotected connections can increase vulnerability to subsequent failures and lower overall network restorability. To improve the service availability, re-provisioning exploits the available capacity in the network to re-establish new backup paths for unprotected connections in advance of a subsequent failure (and right after the recovery from the first fault).

2.2 Reprovisioning Approach

A re-provisioning algorithm typically takes several inputs including network topology/usage information and a list, U, of unprotected demands or demands that require re-provisioning. The algorithm then tries to establish backup lightpaths for unprotected connections using available capacity. One such scheme has been proposed in [10] and its performance is evaluated here, termed thereafter Scheme I. Here, upon recovery from the first failure, this particular algorithm categorizes all unprotected connections into one of the three categories detailed above and attempts to establish new protection lightpaths for each.

However, we note that when a connection (w_i) is restored onto its backup route (b_i), the shared protection capacity along b_i becomes *temporarily unavailable* for other demands whose backup routes also share that capacity. To improve the restorability of these connections, we present a new, improved scheme, termed thereafter Scheme II. Namely, instead of provisioning new backup capacity for these newly unprotected demands (whose total number may be very large), a new path w_i^{new} is provisioned for the failed lightpath, w_i, that is link-disjoint with b_i. Hence, upon completing the provisioning of w_i^{new}, the traffic is simply reverted back from b_i to w_i^{new}. However, note that protection capacity along b_i may not preserve its sharability status as w_i^{new} could be non link-disjoint with (some) demands whose protection routes share protection capacity with b_i. In such a case, a new pair (w_i^{new}, b_i^{new}) is re-provisioned and traffic is reverted from b_i to w_i^{new}. Finally, if this step is not successful, the algorithm computes the set of unprotected connections resulting from the recovery of w_i and re-provisions them accordingly (similar to scheme I). Note that when wavelength conversion is deployed, only the links along b_i where protection wavelength(s) cannot be shared are identified and new protection wavelength(s) on those links are provisioned. Upon finishing this phase, other unprotected connections in other categories that have not been considered are re-provisioned.

The effectiveness of Scheme II is best shown via an illustrative example in Fig. 2. Here we assume that initially b_1, b_2 and b_3 are all setup using λ_1, and b_1 shares λ_1 on link <D-E> with b_2 and on link <E-H> with b_3. Typically when link <B-F> fails, w_1 is restored to its backup b_1 and as a result, b_2 and b_3 become unavailable since they share protection capacity with b_1. Hence w_2, w_3 and b_1 become unprotected and three new protection paths (or capacity) need to be re-provisioned if Scheme I is applied in order to fully protect the network against a subsequent failure. However in Scheme II, when w_1 is restored to its backup, b_2 and b_3 become only *temporarily unavailable*. Hence if we can find a new working path (w_1^{new}) that is link disjoint with b_1 to carry the failed traffic, then b_2 and b_3 can also become available again. Note that w_1^{new} may not be disjoint with w_2 and/or w_3 (w_2 in this example). Therefore, b_1 cannot share any protection resource with b_2. In a wavelength continuous network, a new backup b_1^{new} (and protection wavelength) that is link-disjoint with w_1^{new} has to be provisioned. In a wavelength convertible network, the conflict links are identified (e.g., <D-E>) and a different wavelength is provisioned along those links (e.g., λ_2 can be assigned to b_1 on link <D-E> leaving the rest of the backup lightpath intact). Note that Scheme II differs from Scheme I in that the number of connections to be re-provisioned upon a failure is dramatically reduced, whereas the number of *temporarily unprotected* connections during the re-provisioning time remains the same. The steps for executing Scheme II algorithm are now detailed:

1) Each demand whose working path, w_i, is affected by the first failure is rerouted to its backup route, b_i, and resources along w_i are released.

2) For each b_i

- **a.** Find w_i^{new} with enough capacity that is link-disjoint with b_i and the primary routes of demands sharing protection capacity with b_i; reserve the working capacity along w_i^{new} and revert the traffic into it from b_i.
- **b.** Otherwise, find w_i^{new} with enough capacity that is link-disjoint with b_i:
  - **i.** If successful, revert traffic to it and find a new protection capacity for b_i.
  - **ii.** Otherwise, compute w_i^{new} and reserve corresponding capacity and revert traffic to it, then compute b_i^{new} to protect w_i^{new}.

3) For each connection that fails in step 2, identify the list of unprotected connections:

- **a.** For each unprotected demand, release protection capacity that is already reserved and no longer useable. Repeat until all demands are processed.
- **b.** Compute a link-disjoint route with the working path of each unprotected demand and allocate protection capacity if available.
- **c.** Reserve capacity and go back to 3b. Stop when all unprotected connections are processed.

3 Centralized and Distributed Reprovisioning

The performance of a re-provisioning scheme strongly depends on the implementation of the underlying algorithm. An algorithm typically can either have a centralized or a distributed implementation [11]. Under a centralized implementation, a central network management system holds the global information of network resources, such as network topology, link states, wavelength usage on each link, sharability information for protection resources, etc., and the corresponding steps of the particular algorithm are executed at this central controller. Here, upon the occurrence of a failure, the network will take the responsibility of recovering the failed connections through a standard signaling recovery protocol [4] and the central controller is informed through an alarm message to initiate the re-provisioning procedure. Upon receiving the alarm, the central controller identifies the list of unprotected connections (if scheme I is deployed) resulting from the recovery of failed connections. For every unprotected connection in the list, a new protection path with available capacity is determined. The controller then configures resources for the unprotected connection by notifying each node along the route. If the controller finds there are not sufficient network resources to protect a connection, the connection is deemed unprotected. After the controller receives acknowledgment from each node, it will send a message notifying the source node of the appropriate changes to its protection path.

Similarly, when scheme II is used, the controller starts by first computing a new working path for each failed connection and assigning a wavelength along the path. If successful, then the controller requests the reservation of the wavelength along the new selected route; upon completion, the recovered traffic is reverted back to the new working connection (see section before for details of algorithm). Clearly, under a centralized management, re-provisioning of unprotected connections is done sequentially in order to avoid contention for capacity. Contention for capacity may lead to increasing the number of unprotected connections in the network, therefore increasing its vulnerability. Alternatively, under distributed implementation of scheme I, the source node of each unprotected demand is responsible for re-provisioning new protection capacity for its connection. An unprotected demand is typically identified by either the node detecting the link failure or by the source node of a failed connection. We deploy a distributed provisioning approach with forward reservation [12]. If at least one node along the route is not successful in reserving the selected wavelength, the reservation fails and the connection is deemed unprotected. Here, unlike the centralized scheme, all unprotected connections attempt to reserve protection capacity simultaneously and therefore contentions [11, 12] may likely occur among connections requesting the reservation of the same resource. Clearly, a connection failing to find new protection capacity will be left unprotected and ultimately increasing the network vulnerability to a subsequent failure. Note that, if the number of unprotected connections resulting from the first failure and simultaneously attempting to re-provision new protection capacity is quite large, contentions over resources is more likely to increase; thereby, leaving a large number of unprotected demands in the network upon re-provisioning. Therefore, to achieve a better network restorability, the effect of contentions will have to be reduced. Similarly discussions hold for scheme II. Contentions are likely to occur among multiple connections simultaneously attempting to provision new capacity (i.e., wavelength resources); therefore, and unlike the centralized scheme, resulting in an increase in the number of unprotected connections after re-provisioning. As we have already mentioned, the impact of contentions among unprotected connections trying to re-provision backup capacity is intensified when the number of connections to be re-provisioned is large. One of the advantages scheme II possesses over scheme I is that the number of connections to be re-provisioned is potentially much smaller; therefore making the impact of contentions on network restorability less severe. Nonetheless, it is still a concern as it will prevalent in section IV. To mitigate the impacts contentions may have on the network restorability, we propose that unprotected connections attempting to re-provision and failing to succeed due to contention, be allowed to reattempt after selecting a different wavelength if possible. The advantage of reattempting is that blocking due to contentions may be reduced whereas the drawbacks are increased network re-provisioning times.

4 Simulation Results

We study the performance of lightpath re-provisioning in a sample core topology [10] consisting of 24 nodes and 86 unidirectional links. Requests are uniformly distributed between all source-destination pairs and arrive according to a Poisson traffic model. Meanwhile, the connection-holding time is exponentially distributed and the number of wavelengths per link is W=64. Table 1 summarizes the performance of re-provisioning under centralized implementation. We compare the conventional scheme (Scheme I) versus the proposed scheme (Scheme II) in terms of total number of demands to be re-provisioned (Ri), total number of successfully re-provisioned demands (SRi), and the total number of unprotected demands after re-provisioning (UAi). We simulate the failure of a unidirectional link and calculate the number of unprotected demands upon the failure (before re-provisioning); note that this number (Ui) is the same for both schemes and it is equal to the number of connections to be re-provisioned in Scheme I (i.e., $U1=U2=R1$). For Scheme II, the number of unprotected connections after re-provisioning and the number of successfully re-provisioned connections are measured to determine the total number of re-provisioned connections (i.e., $R2 = UA2 + SR2$).

Results in Table 1 show that when the load is 500 Erlangs, the total number of unprotected connections resulting from the first failure is 146. Subsequently, upon re-provisioning using scheme I, a total number of 146 connections are re-provisioned and only 9 connections are left unprotected (9:146). This shows that re-provisioning dramatically reduces the network vulnerability by protecting vulnerable demands. On the other hand, scheme II shows that although 146 connections are unprotected before re-provisioning, only 63 connections are re-provisioned and 1 connection is left unprotected out of 146 (1:146). Further, when the load increases, e.g. 1000 Erlangs, the gain figures of scheme I and II are 34:177 and 20:177 accordingly with only 113 connections re-provisioned under scheme II. Clearly, the benefits of re-provisioning are evident herein, as the total number of unprotected connections is significantly reduced. We also observe that Scheme II outperforms Scheme I in two other aspects: (1) the total number of unprotected connections in the network after re-provisioning is much lower. This indicates better network restorability and less vulnerability to another failure; (2) the total number of connections that require re-provisioning upon a failure is lower. This yields a clear advantage as it can substantially lighten network management overheads and reduce contention amongst simultaneously re-routing/reservation attempts (i.e., higher re-provisioning successful rate). Overall, the results show that the proposed Scheme II performs less re-provisioning yet achieves better protection. Similarly, the performance results of re-provisioning under distributed implementation are shown in Table 2. Clearly, the results show the advantages of network re-provisioning in reducing the total number of unprotected demands in the network after the first failure. However, it is important to notice that distributed re-provisioning protects fewer connections than the centralized scheme. This is mainly due to the fact that in a distributed environment, connections contend among each other to reserve protection capacities. For example, when the load is 1000 Erlangs, 94 connections (Table 2) are left unprotected after re-provisioning with scheme I whereas only 34 connections (Table 1) are unprotected if re-provisioning is centralized. This therefore will adversely affect the network robustness in advance of a second failure. Similarly, with scheme II, 33 connections are unprotected after re-provisioning (when the load is 1000 Erlangs) vs. only 20 unprotected connections left in centralized implementation.

Two observations are in order here. We first notice that the total number of re-provisioned connections for scheme II under distributed implementation is larger than that under centralized implementation (e.g., 148 vs. 113 at 1000 Erlangs). This is due to the fact that when an unprotected demand is not successful in steps 2.a and 2.b (see section III), step 3 of the algorithm is executed whereby all unprotected demands resulting from this connection are identified and re-provisioned accordingly. Now,

Table 1. Schemes I vs. II-centralized

load	R1	UA1	SR1	R2	UA2	SR2
100	37	0	37	16	0	16
200	48	0	48	23	0	23
300	72	3	69	34	0	34
400	104	3	101	47	0	47
500	146	9	137	63	1	62
600	152	13	139	74	3	71
700	153	20	133	94	10	84
800	167	19	148	97	7	90
900	171	27	144	97	9	86
1000	177	34	143	113	20	93

Table 2. Scheme I vs. II-distributed

load	R1	UA1	SR1	R2	UA2	SR2
100	37	9	28	17	1	16
200	48	17	31	24	1	23
300	72	30	42	44	7	37
400	104	48	56	55	9	46
500	146	71	75	91	14	77
600	152	83	69	100	25	75
700	153	83	70	120	29	91
800	167	83	84	113	21	92
900	171	90	81	135	36	99
1000	177	94	83	148	33	115

unlike centralized re-provisioning, under distributed implementation our experiments showed that more connections will not be successful with steps 2.a and 2.b and as a result, a lager number of connections will be re-provisioned using step 3. Another important observation is that scheme I is more affected by contentions than scheme II. The justification for this is explained by the fact that under scheme I, all unprotected connections are simultaneously re-provisioned which means higher contentions. Whereas under scheme II, steps 2.a and 2.b start by re-provisioning only the failed connections going through the failed link (i.e., the directly affected connections) and finally resort to step 3. Since the total number of failed connections is much smaller than the number of unprotected connections, contentions will have a lower effect.

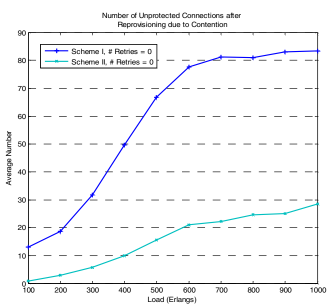


Fig. 3. With no retries

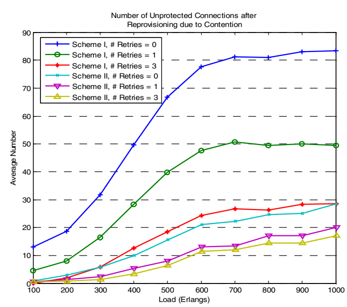


Fig. 4. With retries

Total number of unprotected connections due to contentions

Overall, under distributed re-provisioning, a demand may fail to protect its connection for two reasons: (1) due to unavailable resources or (2) due to contentions with other connections. We measured the impact of contentions on increasing the load in Fig. 3. Clearly, most of connections fail to be protected due to contentions while attempting to reserve capacity. Here, unlike centralized re-provisioning where a central controller maintains an updated database for its resources; in distributed re-provisioning, the blocking due to contentions increases substantially due to the latency in receiving resource updates in time. Therefore, a node attempts to reserve capacity that may already have been reserved by some other connection, leading to blocking due to contentions. Also, Fig. 3 shows the strong impact contentions have on scheme I; this is due to the larger set of unprotected connections attempting to re-provision simultaneously. To minimize the impact of contentions, we propose that a connection that is blocked due to contention be allowed to select a new wavelength and retry its reservation. Fig. 4 shows the improvement of this retry scheme in reducing the number of unprotected connections. The disadvantage of retrying, however, is the increase in the overall re-provisioning time. Our simulations showed that the total network re-provisioning time is kept well under *1 second* when the number of retries is 3.

5 Conclusions

We studied the problem of improving restorability in optical networks for dual, near-simultaneous failures. A novel capacity re-provisioning scheme is introduced in order to reduce the number of unprotected connections after the first failure. The work considers a conventional scheme for lightpath re-provisioning and proposes a new, improved scheme that yields superior performance. We discussed the implementations of re-provisioning under both centralized and distributed control; we showed that under distributed implementations, the restorability of the network degrades due to excessive contentions that occur when simultaneous connections attempt to re-provision. We also showed that the proposed re-provisioning scheme performs much better than a conventional scheme under distributed implementation; that is because the proposed approach reduces contentions since it re-provisions less number of unprotected demands. Finally, we proposed to reduce the impacts of contentions by allowing unsuccessful connections to reattempt re-provisioning.

References

1. W. Grover "Mesh-based Survivable Networks: Options and Strategies for Optical, MPLS, SONET and ATM Networking" prentice hall, 2003.
2. S. Ramamurthy, B. Mukherjee, "Survivable WDM Mesh Networks, Part II - Restoration", *IEEE ICC* 1999.
3. J. Labourdette, "Shared Mesh Restoration in Optical Networks" Proc. OFC'04, Feb. 2004
4. J. Yates, G. Li, "Challenges in Intelligent Transport Network Restoration," Proc. OFC'03, March 2003.

5. M. Clouqueur, W. D. Grover, “Mesh-restorable networks with complete dual failure restorability and with selectively enhanced dual-failure restorability properties”, SPIE OPTICOMM, Boston, MA, July-Aug 2002.
6. D. Schupke, R. Prinz, “Performance of Path Protection and Rerouting for WDM Networks Subject to Dual Failures”, Proc. OFC’03, March 2003.
7. S. Kim, S. Lumetta, “Evaluation of Protection Reconfiguration for Multiple Failures in WDM Mesh Networks”, Proc. OFC’03, March 2003.
8. R. Ramamurthy, A. Akyamac, J-F. Labourdette, S. Chaudhuri, “Pre-Emptive Re-provisioning in Mesh Optical Networks”, Proc. OFC’03, March 2003.
9. P. Charalambous, *et.al.*, “A National Mesh Network Using Optical Cross-Connect Switches”, Proc. OFC’03, March 2003.
10. J. Zhang, K. Zhu, B. Mukherjee, “A Comprehensive Study on Backup Re-provisioning to Remedy the Effect of Multiple-Link Failures in WDM Mesh Networks” *ICC ’04*, Paris, June 2004.
11. L. Shen and B. Ramamurthy, “Centralized vs. Distributed Connection Management Schemes under Different Traffic Patterns in Wavelength-Convertible Optical Networks” Proc. IEEE ICC’02, NY, 2002.
12. Y. Mei, and C. Qiao, “Efficient Distributed Control Protocols for WDM Optical Networks” Proc. ICCCN, September 1997.
13. M. Clouqueur, W.D. Grover, "Availability analysis of span-restorable mesh networks," IEEE JSAC, vol.20, no. 4, May 2002, pp. 810-821.

The Role of Meshing Degree in Optical Burst Switching Networks Using Signaling Protocols with One-Way Reservation Schemes

Joel J.P.C. Rodrigues[1], Mário M. Freire[1], and Pascal Lorenz[2]

[1] Department of Informatics, University of Beira Interior,
Rua Marquês d'Ávila e Bolama,
6201-001 Covilhã, Portugal
{joel, mario}@di.ubi.pt
[2] IUT, University of Haute Alsace,
34, rue du Grillenbreit, 68008 Colmar, France
lorenz@ieee.org

Abstract. This paper discusses performance implications of meshing degree (or nodal degree) for optical burst switching (OBS) mesh networks using signaling protocols with one-way reservation schemes. The analysis is focused on the following topologies: rings, chordal rings with nodal degrees ranging from three to six, mesh-torus, NSFNET, ARPANET and the European Optical Network (EON). It is shown that the largest nodal degree gain, due to the increase of the nodal degree from two to around three, is observed for degree-three chordal ring topology, where as the smallest gain is observed for the ARPANET. For these cases, the magnitude of the nodal degree gain is slightly less than three orders for the degree-three chordal ring and less than one order of magnitude for the ARPANET. On the other hand, when the nodal degree increases from 2 to a value ranging from about four up to six, the nodal degree gain ranges between four and six orders of magnitude for chordal rings. However, EON, which has a nodal degree slightly less than four has the smallest nodal degree gain. The observed gain for this case is less than one order of magnitude. Since burst loss is a key issue in OBS networks, these results clearly show the importance of meshing degree for this kind of networks.

1 Introduction

Optical burst switching (OBS) [1]-[6] has been proposed as an alternative paradigm to overcome the technical limitations of optical packet switching (OPS), namely the lack of optical random access memory and to the problems with synchronization. OBS combines the best of OPS and circuit switching, and it is a technical compromise between wavelength routing (i.e., circuit switching) and optical packet switching, since it does not require optical buffering or packet-level processing and is more efficient than circuit switching if the traffic volume does not require a full wavelength channel. In OBS networks, IP (Internet Protocol) packets are assembled into very large size packets called data bursts. These bursts are transmitted after a burst header

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 44–51, 2005.

packet, with a delay of some offset time. Each burst header packet contains routing and scheduling information and is processed at the electronic level, before the arrival of the corresponding data burst. The burst offset is the interval of time, at the source node, between the transmission of the first bit of the setup message and the transmission of the first bit of the data burst.

According to the length of the burst offset, signaling protocols may be classified into three classes: no reservation, one-way reservation and two-way reservation. In the first class, the burst is sent immediately after the setup message and the offset is only the transmission time of the setup message. This first class is practical only when the switch configuration time and the switch processing time of a setup message are very short. The Tell And Go (TAG) protocol [7] belongs to this class. In signaling protocols with one-way reservation, a burst is sent shortly after the setup message, and the source node does not wait for the acknowledgement sent by the destination node. Therefore, the size of the offset is between transmission time of the setup message and the round-trip delay of the setup message. Different optical burst switching mechanisms may choose different offset values in this range. Just-in-time (JIT) [3], JumpStart [4]-[6], JIT$^+$ [8], just-enough-time (JET) [1] and Horizon [2] are examples of signaling protocols using one-way reservation schemes. The offset in two-way reservation class is the time required to receive an acknowledgement from the destination. The major drawback of this class is the long offset time, which causes the long data delay. Examples of signaling protocols using this class include the Tell And Wait (TAW) protocol [7] and the scheme proposed in [9]. Due to the impairments of no reservation and two-way reservation classes, we concentrate the study in one-way reservation schemes, being considered the following protocols: JIT, JIT$^+$, JumpStart, JET, and Horizon.

A major concern in OBS networks is the contention and burst loss. The two main sources of burst loss are related with the contention on the outgoing data burst channels and on the outgoing control channel. In this paper, we consider bufferless networks and we concentrate on the loss of data bursts in OBS networks.

The reminder of this paper is organized as follows. In section 2, we describe the model of the OBS network under study, and in section 3 we discuss performance implications of the nodal degree for OBS networks with mesh topologies. Main conclusions are presented in section 4.

2 Network Model

In this study, we consider OBS networks with the following mesh topologies: chordal rings with nodal degrees between 3 and 6, mesh-torus with 16 and 20 nodes, the NSFNET with 14-node and 21 links [10], the NSFNET with 16 nodes and 25 links [11], the ARPANET with 20 nodes and 32 links [10], [12], and the European Optical Network (EON) with 19 nodes and 37 links [13]. For comparison purposes bi-directional ring topologies are also considered. These topologies have the following nodal degree: ring: 2.0; degree-three chordal ring: 3.0; degree-four chordal ring: 4.0; degree-five chordal ring: 5.0; degree-six chordal ring: 6.0; mesh-torus: 4.0; NSFNET with 14-node and 21 links: 3.0; the NSFNET with 16 nodes and 25 links: 3.125; the ARPANET with 20 nodes and 32 links: 3.2; and the EON: 3.89.

Chordal rings are a well-known family of regular degree three topologies proposed by Arden and Lee in early eighties for interconnection of multi-computer systems [14]. A chordal ring is basically a bi-directional ring network, in which each node has an additional bi-directional link, called a chord. The number of nodes in a chordal ring is assumed to be even, and nodes are indexed as 0, 1, 2, …, N-1 around the N-node ring. It is also assumed that each odd-numbered node i (i=1, 3, …, N-1) is connected to a node $(i+w) \bmod N$, where w is the chord length, which is assumed to be positive odd. For a given number of nodes there is an optimal chord length that leads to the smallest network diameter. The network diameter is the largest among all of the shortest path lengths between all pairs of nodes, being the length of a path determined by the number of hops. In each node of a chordal ring, we have a link to the previous node, a link to the next node and a chord. Here, we assume that the links to the previous and to the next nodes are replaced by chords. Thus, each node has three chords, instead of one. Let $w1$, $w2$, and $w3$ be the corresponding chord lengths, and N the number of nodes. We represented a general degree three topology by D3T($w1$, $w2$, $w3$). We assumed that each odd-numbered node i (i=1, 3, …, N-1) is connected to the nodes $(i+w1) \bmod N$, $(i+w2) \bmod N$, and $(i+w3) \bmod N$, where the chord lengths, $w1$, $w2$, and $w3$ are assumed to be positive odd, with $w1 \le N-1$, $w2 \le N-1$, and $w3 \le N-1$, and $wi \neq wj$, $\forall i \neq j \wedge 1 \le i,j \le 3$. In this notation, a chordal ring with chord length w is simply represented by D3T(1,N-1,$w3$).

Now, we introduce a general topology for a given nodal degree. We assume that instead of a topology with nodal degree of 3, we have a topology with a nodal degree of n, where n is a positive integer, and instead of having 3 chords we have n chords. We also assume that each odd-numbered node i (i=1,3,…,N-1) is connected to the nodes $(i+w1) \bmod N$, $(i+w2) \bmod N$, …, $(i+wn) \bmod N$, where the chord lengths, $w1$, $w2$, … wn are assumed to be positive odd, with $w1 \le N-1$, $w2 \le N-1$, …, $wn \le N-1$, and $wi \neq wj$, $\forall i \neq j \wedge 1 \le i,j \le n$. Now, we introduce a new notation: a general degree n topology is represented by DnT($w1$, $w2$,…,wn). In this new notation, a chordal ring family with a chord length of $w3$ is represented by D3T(1,N-1,$w3$) and a bi-directional ring is represented by D2T(1,N-1).

We assume that each node of the OBS network supports F+1 wavelength channels per unidirectional link. One wavelength is used for signaling (carries setup messages) and the other F wavelengths carry data bursts. Each OBS node consists of two main components [8]: i) a signaling engine, which implements the OBS signaling protocol and related forwarding and control functions; and ii) an optical cross-connect (OXC), which performs the switching of bursts from input to output. It is assumed that each OXC consists of non-blocking space-division switch fabric, with full conversion capability, but without optical buffers. It is assumed that each OBS node requires [8]: i) an amount of time, $TOXC$, to configure the switch fabric of the OXC in order to set up a connection from an input port to an output port, and requires ii) an amount of time, $Tsetup(X)$ to process the setup message for the signaling protocol X, where X can be JIT, JET, and horizon. It is also considered the offset value of a burst under reservation scheme X, $Toffset(X)$, which depends, among other factors, on the signaling protocol, the number of nodes the burst has already traversed, and if the offset value is used for service differentiation. In this study, it is assumed that [8]: *TOXC* = 10 ms, *Tsetup*(*JIT*)=12.5 μs, *Tsetup*(*JIT*+)=12.5 μs, *Tsetup*(*JumpStart*)=12.5 μs, *Tsetup*(*JET*)=50 μs, *Tsetup*(*Horizon*)=25 μs, the mean burst size, 1/μ, was set to 50 ms, and the burst arrival rate λ, is such that λ/μ=32 (except for figure 3).

3 Performance Assessment

In this section, we make a careful study of the influence of nodal degree on the performance of OBS mesh networks for JIT, JIT^+, JumpStart, JET, and Horizon signaling protocols. Details about the simulator used to produce simulation results can be found in [15]. In chordal ring topologies, different chord lengths can lead to different network diameters, and, therefore, to a different number of hops. One interesting result that we found is concerned with the diameters of the D3T(w_1, w_2, w_3) families, for which w_2=(w_1+2)*mod N* or w_2=(w_1-2)*mod N*. Each family of this kind, i.e. D3T(w_1,(w_1+2)*mod N*, w_3) or D3T(w_1, (w_1-2)*mod N*, w_3), with $1 \le w_1 \le 19$ and $w_1 \ne w_2 \ne w_3$, has a diameter which is a shifted version (with respect to w_3) of the diameter of the chordal ring family (D3T(1, *N*-1, w_3)). For this reason, we concentrate the analysis on chordal ring networks, i. e., DnT(1, 19, w_3, .., w_n).

In order to quantify the benefits due to the increase of nodal degree, we introduce the nodal degree gain, $G_{(n-1)n}(i,j)$, defined as:

$$G_{(n-1),n}(i,j) = \frac{P_i(n-1)}{P_j(n)} \tag{1}$$

where $p_i(n\text{-}1)$ is the burst loss probability in the *i*-th hop of a degree *n*-1 topology and $P_j(n)$ is the burst loss probability in the *j*-th hop of a degree *n* topology, for the same network conditions (same number of data wavelengths per link, same number of nodes, etc), and for the same signaling protocol.

Figures 1 and 2 show, respectively for JIT and JET, the nodal degree gain, in the last hop of each topology, due to the increase of the nodal degree from 2 (D2T(1,19)) to: 3 (D3T(1, 19, 7)), 3.2 (ARPANET), 3.89 (EON – European Optical Network), 4 (D4T(1,19,3,9)), 5 (D5T(1,19,3,7,11)), and 6 (D6T(1,19,3,5,11,15)). Concerning chordal rings, we have chosen among several topologies with smallest diameter the ones that led to the best network performance. As may be seen in those figures, the considered topologies may be sorted from the best performance for the worst performance as: D5T(1,19,3,7,11), D6T(1,19,3,5,11,15), D4T(1,19,3,9), D3T(1, 19, 7), ARPANET, and EON – European Optical Network.

We observed that the performance of the ARPANET is very close to the performance of EON. ARPANET has a nodal degree (3.2) near to the degree-three topology (D3T(1, 19, 7)), and EON has a nodal degree (3.89) near to the degree-four topology (D4T(1,19,3,9)). However, the performance of both ARPANET and EON is worst than the nearest chordal ring degree topology. This results reveals the importance of the way links are connected in the network, since chordal rings and

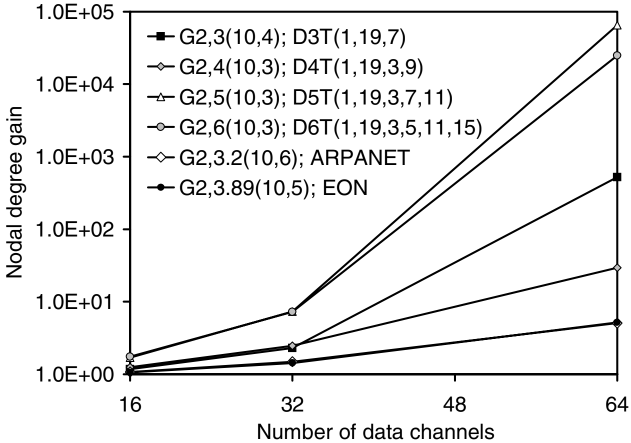


Fig. 1. Nodal degree gain due to the increase of the nodal degree from 2 (D2T(1,19)) to: 3 (D3T(1, 19, 7)), 3.2 (ARPANET), 3.89 (EON – European Optical Network), 4 (D4T(1,19,3,9)), 5 (D5T(1,19,3,7,11)), and 6 (D6T(1,19,3,5,11,15)) as function of the number of data channels, in the last hop of each topology, for JIT signaling protocol; *N*=20

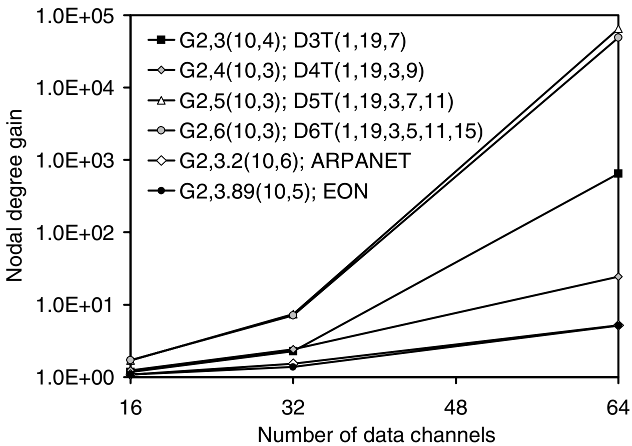


Fig. 2. Nodal degree gain due to the increase of the nodal degree from 2 (D2T(1,19)) to: 3 (D3T(1, 19, 7)), 3.2 (ARPANET), 3.89 (EON – European Optical Network), 4 (D4T(1,19,3,9)), 5 (D5T(1,19,3,7,11)), and 6 (D6T(1,19,3,5,11,15)) as function of the number of data channels, in the last hop of each topology, for JET signaling protocol; *N*=20

ARPANET and EON have similar nodal degrees and therefore a similar number of network links. Results presented in these figures (1 and 2) were obtained for the JIT and JET signaling protocols, and, as may be seen, their performance is very close. This result is confirmed in Fig. 3, that presents the performance comparison of the nodal degree gain for the best (D5T(1,19,3,7,11)) and the worst (EON) of the topologies showed in Figures 1 and 2. Fig. 3 shows the nodal degree gain due to the increase of the nodal degree from 2 (D2T(1,19)) to: 3.89 (EON), and 5 (D5T(1,19,3,7,11)), as a function of λ/μ, in the last hop of each topology, for JIT, JET, Horizon, JIT$^+$, and JumpStart signaling protocols (F=64).

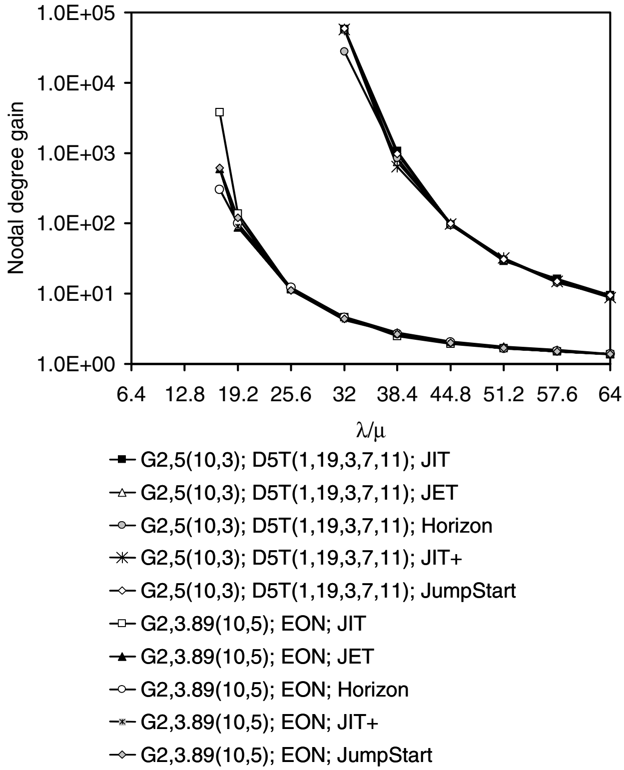


Fig. 3. Nodal degree gain due to the increase of the nodal degree from 2 (D2T(1,19)) to: 3.89 (EON - European Optical Network), and 5 (D5T(1,19,3,7,11)), as a function of λ/μ, in the last hop of each topology, for JIT, JET, Horizon, JIT$^+$, and JumpStart signaling protocols; N=20, F=64

Fig. 4 shows the nodal degree gain, as a function of the nodal degree, due to the increase of the nodal degree from 2 (D2T(1,14)) to 3 (NSFNET (N=14)), from 2 (D2T(1,15)) to: 3 (D3T(1, 15, 5)), 3.125 (NSFNET (N=16)), 4 (D4T(1,15,5,13) and Mesh-Torus (N=16)), and 5 (D5T(1,15,7,3,9)), from 2 (D2T(1,18)) to 3.89 (EON (N=19)), from 2 (D2T(1,19)) to: 3 (D3T(1,19,7)), 3.2 (ARPANET (N=20)), 4 (D4T(1,19,3,9)), 5 (D5T(1,19,3,7,11)), 6 (D6T(1,19,3,5,11,15)), from 2 (D2T(1,24)) to 4 (Mesh-Torus (N=25)), and from 2 (d2T(1,29)) to 6 (D6T(1,29,3,7,11,13)). As may be seen, when the nodal degree increases from 2 to around 3, the largest gain is observed for degree-three chordal rings (a bit less than three orders of magnitude) and the smallest gain is observed for the ARPANET (less than one order of magnitude). When the nodal degree increases from 2 to around 4, the largest gain is observed for degree-four chordal rings (with a gain between four and five orders of magnitude) and the smallest gain is observed for the European Optical Network (with a gain less than one order of magnitude). When the nodal degree increases from 2 to around 5 or 6, the gain is between four or five orders of magnitude, considering that for degree-six chordal ring with 30 nodes, the gain is more then five orders of magnitude. These results clearly show the importance of the way links are connected in OBS networks, since, in this kind of networks, burst loss probability is a key issue.

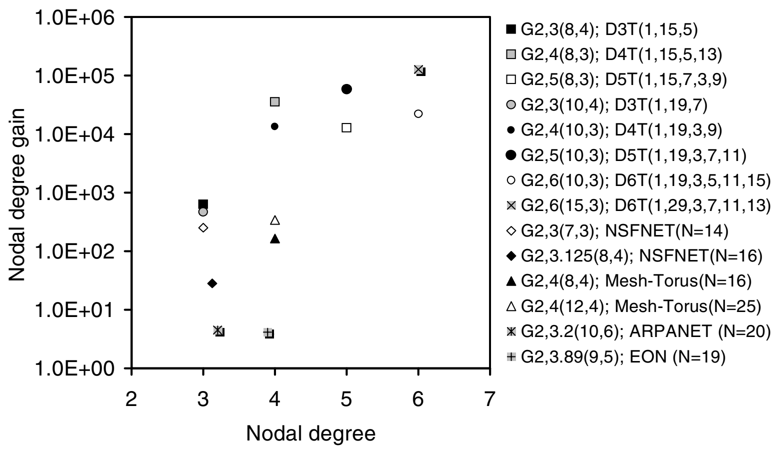


Fig. 4. Nodal degree gain in the last hop of each topology, as a function of the nodal degree, due to the increase of the nodal degree from 2 (D2T(1,14)) to 3 (NSFNET (*N*=14)), from 2 (D2T(1,15)) to: 3 (D3T(1, 15, 5)), 3.125 (NSFNET (*N*=16)), 4 (D4T(1,15,5,13) and Mesh-Torus (*N*=16)), and 5 (D5T(1,15,7,3,9)), from 2 (D2T(1,18)) to 3.89 (EON – European Optical Network (*N*=19)), from 2 (D2T(1,19)) to: 3 (D3T(1,19,7)), 3.2 (ARPANET (N=20)), 4 (D4T(1,19,3,9)), 5 (D5T(1,19,3,7,11)), 6 (D6T(1,19,3,5,11,15)), from 2 (D2T(1,24)) to 4 (Mesh-Torus (N=25)), and from 2 (d2T(1,29)) to 6 (D6T(1,29,3,7,11,13)), for JIT signaling protocol; *F*=64

4 Conclusions

In this paper, we analyzed the influence of nodal degree on the performance of OBS mesh networks with the following topologies: rings, chordal rings, mesh-torus, NSFNET, ARPANET and the EON. It was shown that when the nodal degree increases from 2 to around 3, the largest gain occurs for degree-three chordal rings, being slightly less than three orders of magnitude and the smallest gain occurs for the ARPANET, being the gain less than one order of magnitude. When the nodal degree increases from 2 to around 4, the largest gain is observed for degree-four chordal rings, being between four and five orders of magnitude and the smallest gain is observed for the European Optical Network, being less than one order of magnitude.

Acknowledgements

Part of this work has been supported by the Group of Networks and Multimedia of the Institute of Telecommunications – Covilhã Lab, Portugal, and by the Euro-NGI Network of Excellence of Sixth Framework Programme of EU.

References

1. Qiao, C., Yoo, M.: Optical burst switching (OBS)-A new paradigm for an optical Internet. In Journal of High Speed Networks, Vol. 8, No. 1 (1999) 69-84.
2. Turner, J.S.: Terabit Burst Switching. J. High Speed Networks, Vol. 8, No. 1 (1999) 3-16.
3. Wei, J.Y., McFarland, R.I.: Just-in-time signaling for WDM optical burst switching networks. In Journal of Lightwave Technology, Vol. 18, No. 12 (2000) 2019-2037.
4. Baldine, I., Rouskas, G., Perros, H., Stevenson, D.: JumpStart: A just-in-time signaling architecture for WDM burst-switched networks. Commun. Mag., Vol. 40, No. 2 (2002) 82-89.
5. Zaim, A.H., Baldine, I., Cassada, M., Rouskas, G.N., Perros, H.G., Stevenson, D.: The JumpStart just-in-time signaling protocol: a formal description using EFSM. In Optical Engineering, Vol. 42, No. 2, February (2003) 568-585.
6. Baldine, I., Rouskas, G.N., Perros, H.G., Stevenson, D.: Signaling Support for Multicast and QoS within the JumpStart WDM Burst Switching Architecture. In Optical Networks, Vol. 4, No. 6, November/December (2003).
7. Widjaja, I. Performance Analysis of Burst Admission Control Protocols. IEE Proceeding of Communications, Vol. 142, pp. 7-14, February 1995.
8. Teng, J., Rouskas, G. N.: A Detailed Analysis and Performance Comparison of Wavelength Reservation Schemes for Optical Burst Switched Networks, Photonic Network Communications (to appear).
9. Duser M.; and Bayvel P. Analysis of a Dynamically Wavelength-Routed Optical Burst Switched Network Architecture. J. Lightwave Technol., Vol. 20, No. 4, (2002), 574-585.
10. Sridharan, M., Salapaka, M. V., and, Somani, A. K. A Practical Approach to Operating Survivable WDM Networks, J. Selected Areas in Commun., Vol. 20, No. 1, (2002) 34-46.
11. Ramesh, S., Rouskas, G. N., and Perros, H. G.: Computing blocking probabilities in multiclass wavelength-routing networks with multicast calls, IEEE Journal on Selected Areas in Communications, Vol. 20, No. 1, (2002) 89-96.
12. Nayak, T. K., and Sivarajan, K. N., A New Approach to Dimensioning Optical Networks, IEEE Journal on Selected Areas in Communications, Vol. 20, No. 1, (2002) 134-148.
13. O'Mahony, M. J.: Results from the COST 239 Project: Ultra-high Capacity Optical Transmission Networks, in Proc. European Conf. on Optical Communication (ECOC), Oslo, Norway, Vol. 2, (1996) 2.11-2.18.
14. Arden, B.W., and Lee, H.: Analysis of Chordal Ring Networks. In IEEE Transactions on Computers, Vol. C-30, No. 4 (1981) 291-295.
15. Rodrigues, J.J.P.C., Garcia, N.M., Freire, M.M. and Lorenz, P.: Object-Oriented Modeling and Simulation of Optical Burst Switching Networks, 2004 IEEE Global Telecommunications Conference Workshops (GLOBECOM'2004), Dallas, Texas, Nov. 29- Dec. 3 (2004) 288-292.

Analytical Model for Cross-Phase Modulation in Multi-span WDM Systems with Arbitrary Modulation Formats

Gernot Göger[1] and Bernhard Spinnler[1]

Siemens AG, CT IC 2, Otto-Hahn-Ring 6, D-81739 Munich, Germany

Abstract. Cross-phase modulation (XPM) is a major performance-limiting effect in high capacity wavelength-division-multiplexed (WDM) networks. In this contribution, we present closed expressions for fast and accurate calculation of XPM-induced field distortions. We validate the derived method for various modulation formats and apply it for simultaneous optimization of power and dispersion management. Efficient suppression of self-phase modulation (SPM) and XPM penalties is achieved by a novel dispersion compensation strategy.

1 Introduction

Increasing traffic demand is met by running backbone WDM communication systems with many narrow spaced channels. Simultaneously non-linear multi-channel interaction – particularly XPM – is strongly enhanced. Its impact has to be determined fast and reliably for the applicability of search algorithms for optimum network configurations. Previous studies are confined to XPM penalties in intensity modulated systems [1][2] or dealt with statistical considerations in systems with phase modulation (PM) [3]. In this contribution, we present a generalized method to calculate XPM-induced field distortions in multi-span WDM systems with arbitrary modulation formats and validate its scope. Our method avoids the time-consuming task to solve the multi-channel nonlinear Schrödinger equation (NLSE) by the split-step Fourier (SSF) method.

2 Perturbational Approach to XPM-Induced Field Distortions

The propagation of the complex optical field $E(z,t)$ in a single-mode optical fiber is governed by the NLSE [4]:

$$\partial_z E + \frac{\alpha}{2} E + \beta_1 \partial_t E + i\frac{\beta_2}{2}\partial_t^2 E - i\gamma |E|^2 E = 0. \tag{1}$$

$\beta_m \equiv d^m\beta/d\omega^m$ is the mth coefficent of the frequency expansion of the propagation constant $\beta(\omega)$ at $\omega = \omega_0$, γ is the nonlinear coefficient and α the fiber

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 52–59, 2005.

attenuation. After transformation into the retarded time frame $T = t - z\,\beta_{1,i}$ of channel i and with $E(z,t) = A(z,t)\,e^{-\alpha\,z/2}$ (1) becomes

$$\partial_z A = -i\frac{\beta_{2,i}}{2}\,\partial_T^2 A + i\gamma|A|^2 A. \tag{2}$$

The zeroth order solutions of (2) for a power series expansion $A = \sum_{m=0}^{\infty}\gamma^m A_m$ according to [5] for two channels i and k are[1]

$$\begin{aligned}\tilde{A}_{0,i}(z,\omega) &= \tilde{A}_i(0,\omega)\,e^{i\beta_{2,i}\omega^2 z/2}\\ \tilde{A}_{0,k}(z,\omega) &= \tilde{A}_k(0,\omega)\,e^{i(\beta_{2,k}\omega^2 z/2 - i\omega z d_{i,k})}\end{aligned} \tag{3}$$

where $d_{i,k} = \beta_{1,i} - \beta_{1,k}$ is the group velocity difference between channel i and k. The first order XPM contribution $\tilde{A}_{1,i}(z,\omega)$ (Equation (9) in [5]) can be written as

$$\begin{aligned}\tilde{A}_{1,i}(z,\omega) = \frac{i\gamma}{\pi}\,e^{i\beta_{2,i}\omega^2 z/2}\int_0^z\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-(i\,\beta_{2,i}\omega^2/2+\alpha)\zeta}\\ \tilde{A}_{0,k}(\zeta,\omega_1)\,\tilde{A}_{0,k}^*(\zeta,\omega_2)\,\tilde{A}_{0,i}(\zeta,\omega-\omega_1+\omega_2)\,d\omega_1\,d\omega_2\,d\zeta.\end{aligned} \tag{4}$$

Assuming a continuous wave carrier, i.e. $\tilde{A}_{0,i}(\zeta,\omega) = \bar{A}_i\delta(\omega)$ and inserting (3) into (4) we get

$$\begin{aligned}\tilde{A}_{1,i}(z,\omega) = \frac{i\gamma}{\pi}\,\bar{A}_i\,e^{i\beta_{2,i}\omega^2 z/2}\int_0^z\int_{-\infty}^{\infty} e^{-(i\,\beta_{2,i}\omega^2/2+\alpha)\zeta}\\ e^{i(\beta_{2,k}\omega(2\omega_1-\omega)/2-\omega d_{i,k})\zeta}\,\tilde{A}_k(0,\omega_1)\,\tilde{A}_k^*(0,\omega_1-\omega)\,d\omega_1\,d\zeta.\end{aligned} \tag{5}$$

After setting $a_{i,k}(\omega,\omega_1) = \alpha + i\omega\,d_{i,k} - i\beta_{2,k}\omega(\omega_1-\omega/2) + i\beta_{2,i}\omega^2/2$ and integration

$$\begin{aligned}\tilde{A}_{1,i}(z,\omega) = \frac{i\gamma}{\pi}\,\bar{A}_i\,e^{i\beta_{2,i}\omega^2 z/\,2}\int_{-\infty}^{\infty}\frac{1-e^{-a_{i,k}(\omega,\omega_1)z}}{a_{i,k}(\omega,\omega_1)}\\ \tilde{A}_k(0,\omega_1)\,\tilde{A}_k^*(0,\omega_1-\omega)\,d\omega_1.\end{aligned} \tag{6}$$

Generalization to a N-span system is straightforward. The contribution from span l detected at $L = \sum_{j=1}^{N} L^{(j)}$ can be written as

$$\begin{aligned}\tilde{A}_{1,i}^{(l)}(L,\omega) = \frac{i\gamma_i^{(l)}}{\pi}\,\bar{A}_i\,e^{i\omega^2\sum_{j=l}^{N}\beta_{2,i}^{(j)}L^{(j)}}\int_{-\infty}^{\infty}\frac{1-e^{-a_{i,k}^{(l)}(\omega,\omega_1)L^{(l)}}}{a_{i,k}^{(l)}(\omega,\omega_1)}\\ e^{-i\sum_{j=1}^{l-1}\left(\omega\,d_{i,k}^{(l)}-\beta_{2,k}^{(l)}\omega(\omega_1-\omega/\,2)\right)L^{(l)}}\,g_{k,\mathrm{net}}^{(l)}\\ \tilde{A}_k(0,\omega_1)\,\tilde{A}_k^*(0,\omega_1-\omega)\,d\omega_1.\end{aligned} \tag{7}$$

[1] The Fourier transform sign convention $\tilde{f}(\omega) = \int e^{-i\omega t}f(t)\,dt$ is applied.

With $g_{k,\mathrm{net}}^{(l)} = \prod_{n=1}^{l-1} e^{-L^{(n)}\alpha^{(n)}} g_k^{(n)}$ the net gain of channel k at the start of span l compared to the first span, and $d_{i,k}^{(l)} = \beta_{1,i}^{(l)} - \beta_{1,k}^{(l)}$, $a_{i,k}^{(l)}(\omega,\omega_1) = \alpha^{(l)} + i\omega\, d_{i,k}^{(l)} - i\beta_{2,k}^{(l)}\omega(\omega_1 - \omega/\,2) + i\beta_{2,i}^{(l)}\omega^2/\,2$ as before, summing over all M channels and N spans yields

$$\tilde{A}_{1,i}^{(l)}(L,\omega) = \frac{i}{\pi}\,\bar{A}_i \sum_{k=1,k\neq i}^{M} \sum_{l=1}^{N} \gamma_i^{(l)} g_{k,\mathrm{net}}^{(l)}\, e^{i\omega^2 \sum_{j=l}^{N} \beta_{2,i}^{(j)} L^{(j)}/\,2}$$
$$\int_{-\infty}^{+\infty} e^{-i\sum_{j=1}^{l-1}\left(\omega\, d_{i,k}^{(l)} - \beta_{2,k}^{(l)}\omega(\omega_1 - \omega/\,2)\right) L^{(l)}}\; \frac{1 - e^{-a_{i,k}^{(l)}(\omega,\omega_1) L^{(l)}}}{a_{i,k}^{(l)}(\omega,\omega_1)}$$
$$\tilde{A}_k(\omega_1)\,\tilde{A}_k^*(\omega_1 - \omega)\, d\omega_1 . \tag{8}$$

3 Verification of the Model

To check the accuracy of the model, we first consider a system with five spans each consisting of 100 km SSMF with D = 17.0 ps/(nm km), $\alpha = 0.20$/km and $\gamma = 1.297$/(W·km) and of 15.686 km dispersion compensating fiber (DCF) with

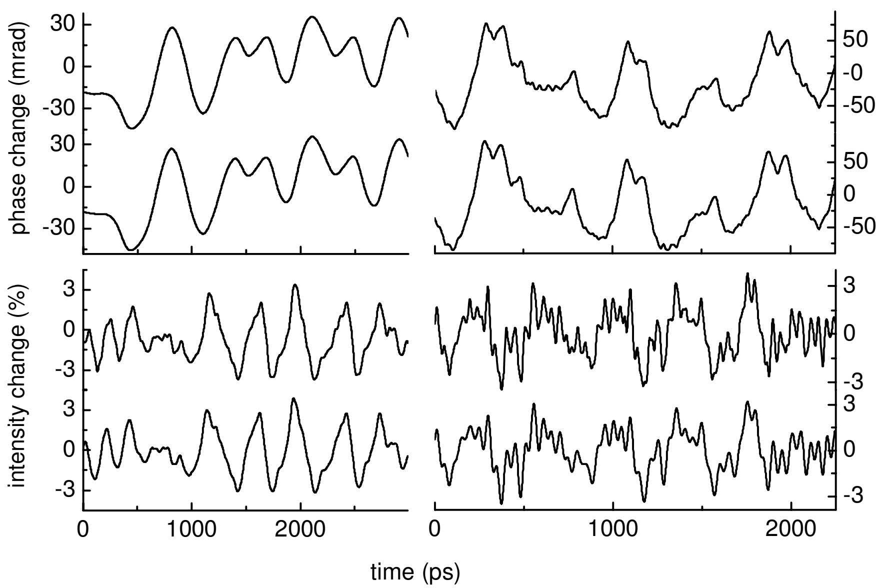


Fig. 1. Temporal XPM-induced phase (top) and intensity distortions (bottom) of the cw probe channel for the model system with NRZ (left) and NRZ-DPSK (right) modulation. SSF simulation (upper curves) and theoretical predictions (lower curves). Constant phase shift due to SPM has been subtracted

D = -102 ps/(nm·km), $\alpha = 0.50$/km, and $\gamma = 2.954$/(W·km) corresponding to a residual dispersion of 100 ps/nm per span. One 10 Gb/s channel at 193.45 THz with NRZ and NRZ-DPSK modulation, respectively is employed. The continuous wave (cw) probe channel is located at 193.4 THz. Per channel launch powers into SSMF amount to 3 dBm for NRZ and 6 dBm for NRZ-DPSK modulation. DCF launch powers are chosen 4 dB lower. As can be seen in Fig.1, theoretical predictions for phase and intensity distortions agree very well with results obtained by the SSF method. Despite the sharp leading and trailing edges of the modulated channel's pulses with roll-off factor 0.5, the curve for phase changes runs rather smoothly due to the low-pass filter characteristics of PM-PM conversion.

In a next step, we test the proposed procedure for various network configurations. All channels are modulated. Since – different from noise-like multichannel interactions – lower order perturbational or Volterra series solutions of the NLSE for single channel propagation turn out not to be satisfactory [6], SPM influence is taken into account by propagating a 32-bit random sequence by the SSF method. XPM-induced field fluctuations according to (8) scaled with $\bar{A}_i = A_{\mathrm{SPM}}(L, t)$ are repeatedly superposed on this sequence. After -680 ps/nm dispersion precompensation, seven 50 GHz spaced channels are transmitted over 15 spans each consisting of 90 km SSMF (here $\alpha = 0.25$ dB) and of 15 km DCF. 10.7 Gb/s NRZ, NRZ-DPSK and NRZ-DQPSK channels with mean per channel launch powers of 3 dBm are employed. Amplifier noise figures of 5.5 dB, optical (Gauss) and electrical (fifth order Bessel) filter bandwidths of 25 and 7.5 GHz are chosen. In case of PM signals, a balanced detector is inserted. Probability density functions are derived according to [7].

For comparison we perform numerical SSF simulations for the corresponding systems with PRBS length of 2^7-1. As for the analytical method ASE is not accounted for on the link but it is added at the end of the link as an equivalent noise process with analytically calculated variance. In order to generate a proper noise statistics the received signal is repeated periodically before adding

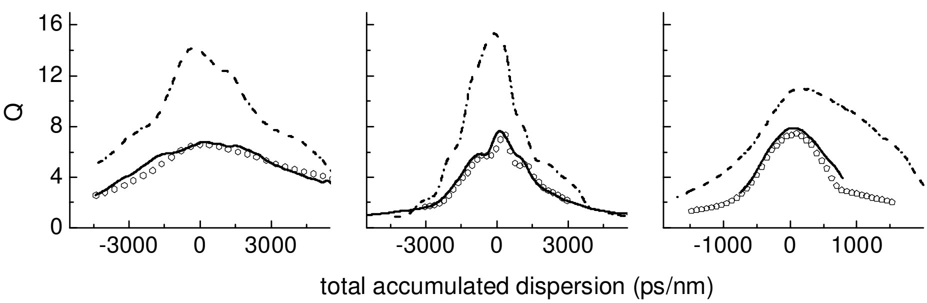


Fig. 2. Q-factor of the channel at 193.4 THz versus total accumulated dispersion for NRZ-DQPSK (left), NRZ-DPSK (middle) and NRZ modulation (right) for full inline dispersion compensation. SPM only (dashed), multi channel SSF calculations (solid) and semi-analytic model (symbols)

noise. The bit-error rate (BER) at the optimum decision threshold is obtained by Monte-Carlo simulations for a set of several (non-optimum) thresholds and subsequent tail extrapolation. The calculated BER for the examined channel at 193.4 THz is mapped for both methods via the standard relationship onto an equivalent Q-factor. Figure 2 shows the good agreement of the proposed model with the SSF method results.

4 Application to Network Optimization Problems

Fast Q-factor assessing algorithms enable the successful search for regions in parameter space where extreme values are located, e.g. maxima of Q or configurations of largest dispersion tolerance. Here we focus on the Q extremum. Firstly, a strictly linear dispersion map is examined for a link consisting of 25 fiber spans. The residual dispersion per span and the per channel launch power are chosen as the degrees of freedom. Dispersion precompensation is set to -300 ps/nm. Postcompensation is adjusted in each configuration for maximum Q. All other parameters correspond to those of the previous section. Again re-

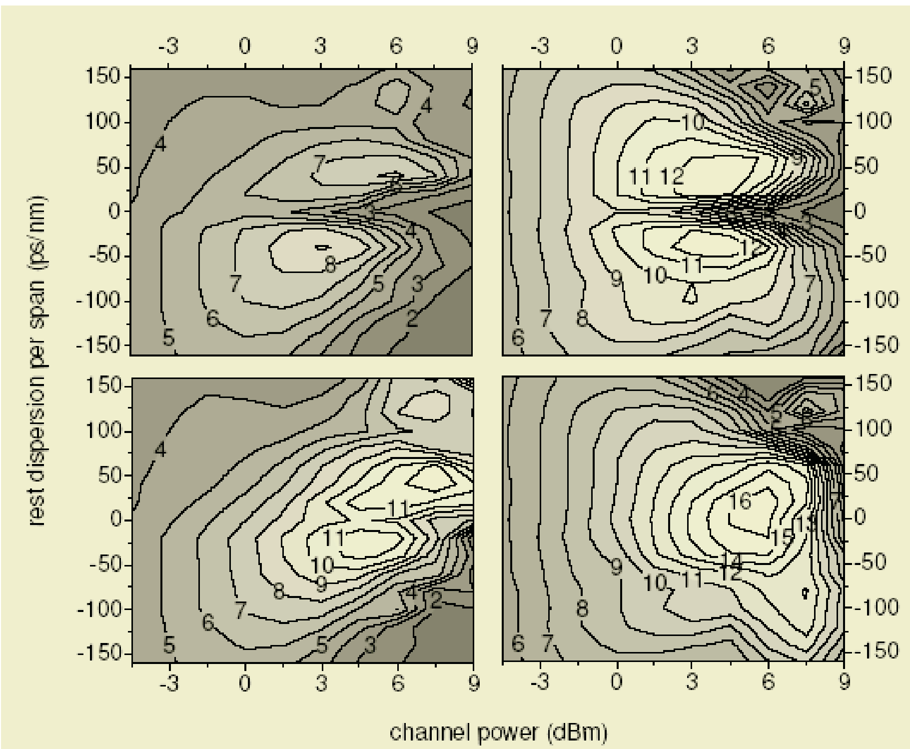


Fig. 3. Maximum Q-factor contour plots versus channel launch power and residual dispersion per span for NRZ (left) and NRZ-DPSK modulation (right). Single (bottom) and multi channel results (top)

sults are taken for the middle channel at 193.4 THz of eight active channels 193.6, 193.55 ... 193.25 THz.

For NRZ single channel transmission, slight dispersion over-/under-compensation of about ±30 ps/nm per span and 7 dBm per channel launch power are favourable. The optimum dispersion values also hold in case of additional XPM interaction (see left half of Fig.3) at 1.5 dB lower channel launch powers. For NRZ-DPSK modulation, the Q-factor maximum at full inline compensation for single channel propagation is more pronounced due to the inherent OSNR gain of the modulation process. The so defined region is sharply divided by the introduction of XPM into two best choice regions around (3.5 dBm, 30 ps/nm) and (3.5 dBm, −30 ps/nm) (right half of Fig.3).

Disengaging from strict linear dispersion maps with constant per span residual dispersion we now turn to double periodic schemes: besides the common (local) per span rest dispersion an additional mean per span rest dispersion is defined by inserting/removing extra DCF after each fifth span. For these studies we set the per channel launch power to 3 dBm. The contour plot in Fig.4 left shows the dependence of the maximum achievable Q-factor on the mean and local rest dispersion values for NRZ modulation. Compared with the standard map results in the top left graph of Fig.3, Q can be enhanced by about 1.5 (3 by restriction to positive mean rest dispersion). Considering the approximately equal maximum Q values of the corresponding single channel graphs (bottom left graph of Fig.3 and Fig.4 right), one is led to the conclusion that the Q increment in multi-channel transmission with the improved new map (-120 ps/nm local and 40 ps/nm mean per span rest dispersion) stems primarily from effective XPM suppression.

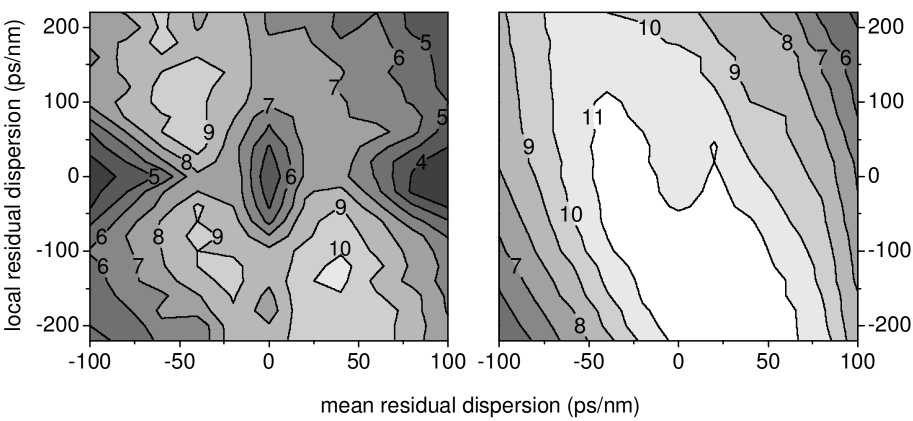


Fig. 4. Maximum Q-factor contour plots versus mean and local residual dispersion for NRZ modulated channels. Per channel launch power is fixed at 3 dBm. Single (right) and multi channel calculations (left)

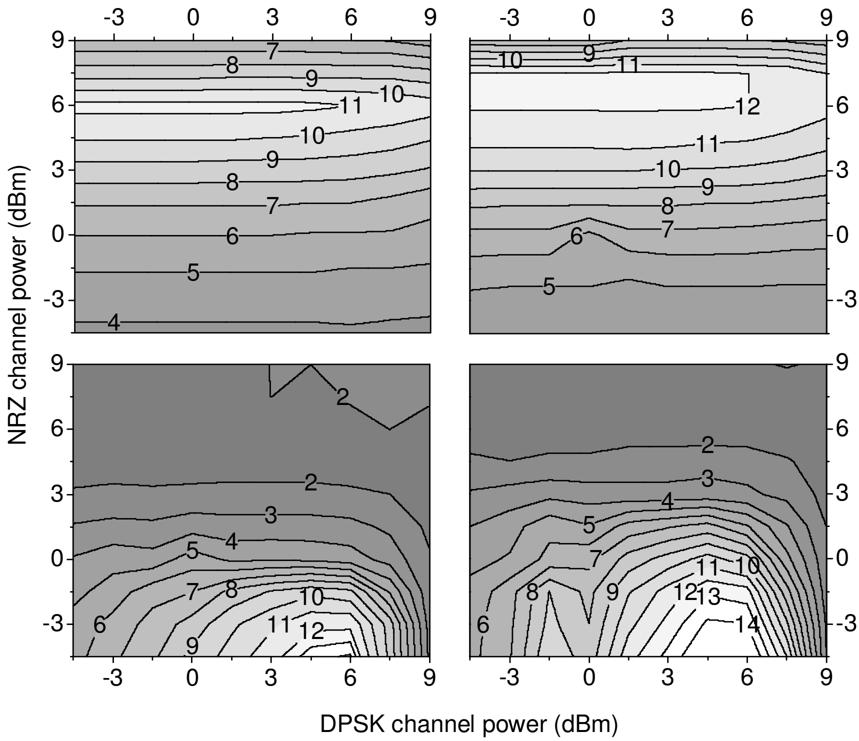


Fig. 5. Maximum Q-factor contour plots versus per channel launch powers for mixed modulation system with alternating NRZ and NRZ-DPSK channels. 193.4 THz NRZ (top) and 193.45 THz NRZ-DPSK channel (bottom) each for standard dispersion map with 30 ps/nm residual dispersion per span (left) and for double period dispersion map (right)

Finally the compatibility of simultaneous transmission of alternating NRZ and NRZ-DPSK channels is studied. For the left two graphs in Fig.5 a 30 ps/nm per span dispersion undercompensation scheme is employed. The right part is calculated for the novel map (-120 ps/nm, 40 ps/nm). The top (bottom) contour plots depict the functional dependence of the maximum Q of the NRZ (NRZ-DPSK) channel at 193.4 (193.45) THz versus DPSK and NRZ channel power. Quite general, phase modulated channels suffer dramatically from strong phase distortions injected from neighbored on-off keying (OOK) channels [8] – disproportionate to the higher peak power of OOK. The optimum linear dispersion map is not compliant with an established Q-factor limit of 6. But for the novel dispersion compensation scheme, both NRZ and NRZ-DPSK channels can be configured to meet this requirement.

5 Conclusion

Our new analytical method for rapid calculation of XPM impairments is based on a first order perturbational approach. A comparative study with numerical SSF simulations shows very good agreement for numerous test configurations. Applied to network optimization, improved non-standard configurations for multi-channel and mixed modulation format transmission are found.

References

1. Cartaxo, A.: Cross-Phase modulation in intensity modulation-direct detection WDM systems with multiple optical amplifiers and dispersion compensators. J. Lightwave Technol. **17** (1999) 178–190
2. Killey, R.I., Thiele, H.J., Mikhailov, V., Bayvel, P.: Prediction of transmission penalties due to cross-phase modulation in WDM systems using a simplified technique. IEEE Photon. Technol. Lett. **12** (2000) 804–806
3. Norimatsu, S., Iwashita, K.: The influence of cross-phase modulation on optical FDM PSK homodyne transmission systems. J. Lightwave Technol. **11** (1993) 795–804
4. Agrawal, G.P.: *Nonlinear Fiber Optics*. 2^{nd} ed., New York, Academic (1995)
5. Vanucci, A., Serena, P., Bononi, A.: The RP method: a new tool for the iterative solution of the nonlinear Schrödinger equation. J. Lightwave Technol. **20** (2002) 1102–1112
6. Lee, J.H.: Analysis and characterization of fiber nonlinearites with deterministic and stochastic signal sources. Dissertation, Virginia Polytechnic Institute and State University (2000)
7. Marcuse, D.: Calculation of bit-error probability for a lightwave system with optical amplifiers and post-detection Gaussian noise. J. Lightwave Technol. **9** (1991) 505–513
8. Spinnler, B., Hecker-Denschlag, Calabrò, S., Herz, M., Weiske, C.-J., Schmidt, E.-D., van den Borne, D., Khoe, G.-D., de Waardt, H., Griffin, R., Wadsworth, S.: Nonlinear tolerance of differential phase shift keying modulated signals reduced by XPM. OFC'2004, Los Angeles, CA, 2004, paper TuF3

Low-Cost Design Approach to WDM Mesh Networks

Cristiana Gomes and Geraldo Robson Mateus

Federal University of Minas Gerais, Computer Science Department,
6627 Avenida Antônio Carlos, Belo Horizonte, MG, Brazil
{cmng, mateus}@dcc.ufmg.br

Abstract. This article presents a mathematical model and an efficient heuristic that results in a low-cost network design to satisfy a set of static point-to-point demands. It considers the problem of routing working traffic and assigning wavelengths in an all-optical network. This problem is known as the Routing and Wavelength Assignment (RWA) problem.

The model and heuristic give a physical network configuration selecting a lowest cost set of components of the network (subnetworks and switches) with sufficient capacities to attend all demands.

The solutions obtained are compared to existing results found in the literature using the same instances.

We treated the project of a network without wavelength conversion because it introduces a delay (Optical-Electrical-Optical mappings) and this should be avoided in our environment, a core of a backbone.

1 Introduction

To support Internet demands there is a natural tendency to transform backbones in all-optical networks. Wavelength Division Multiplexing (WDM) technology is considering as a good option for building the next generation Internet structure.

This technology is an excellent way to increase the capacity of optical networks. It is necessary due the increase of communication applications. A given node may transmit optical signals on different wavelengths that are couples into a single fiber using wavelength multiplexers [1].

The Routing and Wavelength Assignment (RWA) problem is to find the suitable routing paths and wavelengths for each demand request so that no two paths sharing a link are assigned to the same wavelength. According to [2] the RWA problem is critically important to increase the efficiency of wavelength-routed all-optical networks. The solution of the RWA problem provides an optimal configuration to a WDM environment.

In an all-optical WDM network, a logical connection between a pair of nodes, say (o, d), is a path or route composed of a sequence of links from o to d called a lightpath [1]. Such a network consists of a number of optical cross-connects (OXCs) arranged in some arbitrary topology and each OXC can switch the optical signal coming in on a wavelength of an input fiber link to the same wavelength

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 60–67, 2005.

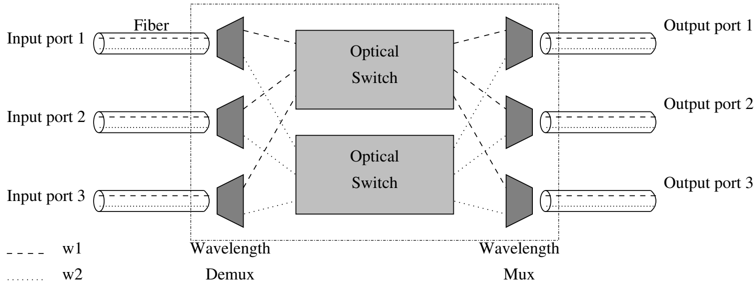


Fig. 1. 3×3 OXC with two wavelength per fiber [3]

in an output fiber link. An OXC with n input and n output ports capable of handling w wavelengths per port can be thought as w independent $n \times n$ optical switches. These switches have to be preceded by a wavelength demultiplexer and followed by a wavelength multiplexer to implement an OXC [3], as shown in Fig. 1.

In the core of the backbone it is wished that a lightpath does not undergo any conversion to and from electrical form, in this way there is nothing in the signal path to limit the throughput of the fibers.

International Telecommunications Union (ITU) has developed standards that specify the architecture of WDM optical transport networks (OTN) [4].

We define a set of possible paths for each demand pair. These paths should be selected and get an adequate wavelength. Such solution proposed by our heuristic can suggest a good initial configuration for a given demand. According to [5] alternate routing can improve the blocking performance and generally provides significant benefits.

In [6] were investigated different Integer Linear Programming (ILP) formulations for solving the RWA problem with two path protection schemes. We consider the instances and model in [1]. These instances have two link-disjoint routes between the source and the destination nodes in order to recover from any single link failure. We propose a modification of this mathematical formulation of the RWA problem over an all-optical network. This change disrespects the selection of wavelengths based in value of the allocated ones. It turns our model simpler. We will show that our model get best results considering time and cost.

The RWA problem is NP-complete. It was proved in [7], by showing the equivalence of the problem to the graph-coloring problem. Therefore several research works have focused on developing efficient heuristics. This article proposes a two-phase heuristic. In the first phase, the flows are distributed over the paths considering cost and links load. In the second phase, the wavelengths are assigned by solving a coloring problem. The second phase is often solved as a coloring problem defined on a conflict graph generated by the set of alternate paths. Such method was adopted in [8].

We used fixed-alternate routing and propose a heuristic that uses a method such LLR (Least-Loaded Routing) [9], using components cost in a first time, and link load in a second time. The heuristic selects several available routes in an adaptive way depending of the current state of the network. The load balancing is important to the second phase, when we use graph coloring. It prevents for blocking demands.

2 Model

We consider a network with N nodes and E links (i, j). The component capacity is measured in wavelength numbers such the set $W = \{4, 16, 20, 40, 80\}$. The wavelengths are enumerated from 1 to 80 that will be assigned as necessary. The demands appear in set D under belong to the format (o, d) where $o \in N$ and $d \in N$. To represent the physical network structure, it was defined a set of subnetworks that are mesh-type and must be chosen to compose the network in terms of nodes and links costs. Each mesh of this set is created from the spanning tree of the network graph, adding edges that are in the original graph and not in the spanning tree. Each added edge creates a cycle in the graph. Each cycle represents a subnetwork in set S. These cycles can have new edges from the original graph to obtain mesh-type subnetworks.

The links that compose a subnetwork $s \in S$ generate the set $E_{\mathrm{S},s}$. Switches are located between subnetworks to satisfy the existing demands; they join the subnetworks by means of some nodes in common permiting the communication. Let C be the set of available switches. Let P be the set of routes that link the pairs $(o, d) \in D$. These routes are disjoint (fault tolerance) and represent the shortest paths found between the pairs (o, d) using Dijkstra algorithm. These paths compose the set P, they are calculated on the complete graph taking into account all the potential subnetworks and switches. Several paths can satisfy the same demand pair $(o, d) \in D$ and compose the set $J_{o,d}$. The routes that involve two or more subnetworks must use intermediate switches $c \in C$. All routes using a switch c compose the set L_c.

A constant B exists to penalize the unsatisfied demand. The parameter $r_{o,d}$ is the number of used wavelengths by the pair $(o, d) \in D$. $f_{c,w}$ is the cost of using a switch $c \in C$ of size $w \in W$ and $a_{s,w}$ is the use of a subnetwork $s \in S$ of size $w \in W$. The variable x_p stands for the number of wavelengths that must be assigned to the path $p \in P$. The unsatisfied demands appear in $u_{o,d}$. The variables $y_{s,w}$ and $z_{c,w}$ represent, respectively, the use of a subnetwork s or a switch c with size w. $l_{\mathrm{f},i,j,s}$ and $c_{\mathrm{f},c}$ represent, respectively, the flow in number of wavelengths on the link (i, j) of a subnetwork s and on a switch c.

The modified model is shown below. The sets and variables were defined in [1] and we reuse them in our new model to be able to test in the same network instances. A comparison is shown in the next section.

$$\min \sum_{s \in S} \sum_{w \in W} a_{s,w} y_{s,w} + \sum_{c \in C} \sum_{w \in W} f_{c,w} z_{c,w} + B \sum_{(o,d) \in D} u_{o,d} \tag{1}$$

$$\sum_{p \in setjod} x_p + u_{o,d} = r_{o,d}, \ \forall (o, d) \in D \tag{2}$$

$$\sum_{p \in P\mathrm{es}_{,s,i,j}} x_p = l_{\mathrm{f},i,j,s}, \; \forall s \in S, i,j \in E_{\mathrm{S},s} \tag{3}$$

$$\sum_{(o,d) \in D} f_{\mathrm{r},o,d,w,s,i,j} \leqslant 1, \; \forall s \in S, (i,j) \in E_{\mathrm{S},s}, w \in F \tag{4}$$

$$\sum_{w \in F} f_{\mathrm{r},o,d,w,s,i,j} = \sum_{p \in \left(P\mathrm{es}_{,i,j,s} \cup J_{o,d}\right)} x_p, \; \forall s \in S, (i,j) \in E_{\mathrm{S},s}/i < j, \forall (o,d) \tag{5}$$

$$\sum_{p \in L_c} x_p = c_{\mathrm{f},c}, \; \forall c \in C \tag{6}$$

$$\sum_{w \in W} w.y_{s,w} \geqslant l_{\mathrm{f},i,j,s}, \; \forall s \in S, (i,j) \in E_{\mathrm{S},s} \tag{7}$$

$$\sum_{w \in W} w.z_{c,w} \geqslant c_{\mathrm{f},c}, \; \forall c \in C \tag{8}$$

$$\sum_{w \in W} y_{s,w} \geqslant 1, \; \forall s \in S \tag{9}$$

$$\sum_{w \in W} z_{c,w} \geqslant 1, \; \forall c \in C \tag{10}$$

The objective function minimizes the network cost. It selects switches and subnetworks with enough power to attend all demands and using the minimum number of wavelengths. The demand not attended is penalized.

- Eq.(2): For each pair (o,d), the demand must be attended assigning wavelengths to the paths $p \in J_{o,d}$ and considering the unsatisfied $u_{o,d}$ demand for the pair (o,d).
- Eq.(3): The total flow in the link (i,j) of a subnetwork s represents the sum of the wavelengths in each path using the same link and subnetwork.
- Eq.(4): The dedicated paths for a demand pair (o,d) are disjoints. There is only one path for each link (i,j) of a subnetwork s. Therefore in all paths in $J_{o,d}$, only one should use the wavelength $f \in F$ in the same link and subnetwork.
- Eq.(5): The wavelength number used by the pair (o,d) in a link (i,j) of the subnetwork s, must be equal to the flow in just one path in $J_{o,d}$ using the same link and subnetwork.
- Eq.(6): The wavelength number in a switch in the network is equal to the sum of the flows in all paths that use this switch.
- Eq.(7): The adequate capacity of a subnetwork s must be greater or equal than the wavelength number in any link in this subnetwork.
- Eq.(8): The adequate capacity of a switch c must be greater or equal than the number of wavelengths over it.
- Eq.(9-10): Each switch and subnetwork must have a single size $w \in W$.

3 Heuristic

We propose a two-phases heuristic for RWA. In the first phase, the flow is distributed over the paths in the network taking into account the path cost. The path cost is a sum of the components cost, subnetworks and switches, along of the path.

The flow is distributed until the path reaches the maximum capacity before to jump to the next level of capacity in W. The path capacity is represented by the bigger component capacity in this path. If it happens the algorithm chooses another path associated with the current demand. If all paths are full then the cheapest one is chosen. It increases the capacity to the next level in W. The demands are chosen considering their number of requests and their number of dedicated paths. They are sorted in a decrease order of the number of requests for each dedicated path. We consider a fairness method for the moment.

After completing flow distribution, a component is chosen if it presents little flow above its current capacity. For this reason it is easy to reduce its cost by moving the flow over. The algorithm manipulates the flow of the chosen components, trying to reduce the cost of this component. It saves the configuration if it gets a total network cost reduction.

The network cost is the sum of all components cost. The heuristic stop when a maximum number of iterations are reached. When it reaches a value lower than the values given by designer for the maximum network cost, the number of unattended demands and the number of iterations, the algorithm goes to the second phase. In this phase, the wavelengths are assigned for each demand request by solving a coloring problem. We use a variant of the color degree heuristic [10] in this phase.

We illustrate an instance of coloring problem in Figs 2-4. A five nodes network is represented in Fig 2. In Fig 3 we show a graph where the nodes represent the requests over the defined paths for the network topology. These paths were selected in the first phase and they represent the set of a minimum cost to attend the given demand. There are the demands $5 \to 1$, $5 \to 2$, $2 \to 3$ and $1 \to 4$. Each demand has one dedicated path in this example and we consider only one request to facilitate the comprehension. All requests in the same path will have necessarily an edge in graph on Fig. 3. In Fig. 4 we show the graph coloring. The colored nodes define the used paths. The colors define the selected wavelengths to each demand request.

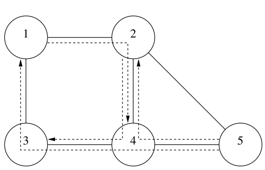


Fig. 2. Network graph with demand path

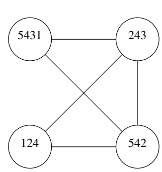


Fig. 3. Graph representing the path conflicts

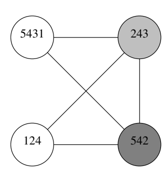


Fig. 4. Color graph and associate the wavelength to each path

We denote by I the maximum number of iterations, L the number of links, P the number of paths and R the number of requests. The worst-case complexity order to this algorithm is $O\left(I\left(LP^2 + R^2\right)\right)$.

4 Experiments

The instances are 20 randomly generated problems of various sizes. The problem characteristics are showed below. The problems named ATT01 through ATT05 and EUR01 through EUR05 represent a European network described in [11], and they have topologies given in Figs. 5 and 6.

Instance	Nodes	Links	Demands	Subnetworks	Switches	Paths
A01	6	9	9	5	5	15
A02	6	9	9	5	5	15
A03	6	9	9	5	5	15
J01	8	12	14	5	6	19
J02	10	15	24	5	6	30
J03-7	9	13	15	5	6	17
ATT01-5	11	23	16	6	7	22
EURO1-5	18	35	18	6	7	20

The instances also have differents paths and demand requests. Our model was coded using the AMPL modeling language and CPLEX Linear Optimizer 7.0.0 on the SunBlade UltraSPARC 500Mhz, 1GB RAM. Table 1[1] shows the results obtained by the proposed model and the results obtained by the model in [1], both using CPLEX.

Table 2 shows the results obtained by the proposed heuristic and the results obtained by the model in this article. The heuristic reached the cost showed in this table with a computation time machine lower than one minute. We repeat the computation time to solve the model using CPLEX in the last column.

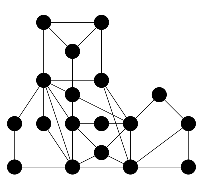


Fig. 5. EUR0x graphs

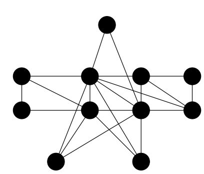


Fig. 6. ATTox graphs

In the majority of the problems our model obtains the best cost in a lower computation time than model in [1]. It happens because their model assigns the capacity of the subnetworks and switches based on the highest wavelength

[1] Meaning of symbols: * = 1h of Computational Time; ** = 8h of Computational Time; *** = Memory Limit; $= 'Branch=1'.

Table 1. Comparison between the models

	Cost				Time (s)	
Instance	OurModel	Attended dem.	Best result in [1]	Attended dem.	Our Model	CPLEX [1]
A01	1059	100%	1297(CPLEX)	100%	14.18	1300
A02	649	100%	874(CPLEX)	100%	15.66	17000
A03	1623	100%	1903(CPLEX)	100%	11.3	65
J01	1998	100%	3233(HEUR)	98,59%	34.29	29000
J02	1222**$	65,62%	2267(CPLEX)**	68,75%	29000	29000
J03	2682	100%	2885(CPLEX)**	100%	120.29	29000
J04	1465*	100%	1899(CPLEX)**	100%	3600	29000
J05	2087	100%	2816(HEUR)	100%	52.34	29000
J06	3738	100%	4312(CPLEX)**	100%	986.92	29000
J07	3167	100%	3585(CPLEX)**	100%	64.83	29000
ATT01	2301	94,12%	2707(CPLEX)**	94,12%	307.98	29000
ATT02	3187	100%	3810(CPLEX)***	100%	272.71	24000
ATT03	3901	94,44%	5042(CPLEX)**	94.44%	3624.47	29000
ATT04	4266	100%	5885(HEUR)	100%	73.53	29000
ATT05	4204	100%	5604(CPLEX)***	100%	49.93	9800
EUR01	no-sol**		14310(CPLEX)**	92,86%	29000	29000
EUR02	15603	98.16%	16830**	98,16%	3400	29000
EUR03	17149**	81,60%	19422**	81,60%	29000	29000
EUR04	21241**	85,96%	21240**	85,96%	29000	29000
EUR05	20558**	80,49%	23346**	80,49%	29000	29000

Table 2. Comparison between the model and heuristic

	Cost				Time (s)
Instance	Our Model	Attended dem.	Prop. heuristic	Attended dem.	Our Model
A01	1059	100%	1059	100%	14.18
A02	649	100%	649	100%	15.66
A03	1623	100%	1623	100%	11.3
J01	1998	100%	1998	100%	34.29
J02	1222**$	65,62%	1748	69%	29000
J03	2682	100%	2682	100%	120.29
J04	1465	100%	1465	100%	3600
J05	2087	100%	2087	100%	52.34
J06	3738	100%	3738	100%	986.92
J07	3167	100%	3167	100%	64.83
ATT01	2301	94,12%	2420	94%	307.98
ATT02	3187	100%	3187	100%	272.71
ATT03	3901	94,44%	4748	94%	3624.47
ATT04	4266	100%	4266	100%	73.53
ATT05	4204	100%	4204	100%	49.93
EUR01	no-sol**		14310	97%	29000
EUR02	15603	98.16%	15603	96%	29000
EUR03	17149**	81,60%	17977	81%	29000
EUR04	21241**	85,96%	19222	85%	29000
EUR05	20558**	80,49%	19290	77%	29000

allocated over them. They assigned blocks of wavelengths, and therefore, it's expected that the highest wavelength might be greater than the capacity to simply support the absolute flow in the components. Our model assigns capacity based only on the absolute flow. CPLEX directives are being studied, like 'branch=1' that considers only some nodes on the branch-bound tree, to get better results. We used it in J02 and EUR01 but it has not achieved a good outcome yet.

5 Conclusion

This paper proposes a simplified version of mathematical model in [1] and an efficient heuristic to the RWA problem.

We showed that not considering the value of wavelength to put the others wavelengths it simplifies the problem. The heuristic showed good results in an acceptable time for design networks. It is being modified to support dynamic requests and to treat traffic in number of Mbps considering grooming.

References

1. Kennington, J., Olinick, E.: Wavelength routing and assignment in a survivable WDM mesh network. Operations Research **51** (2003) 67–79
2. Ozdaglar, A., Bertsekas, D.: Routing and wavelength assignment in optical networks. IEEE/ACM Transactions on Networking **11** (2003)
3. Rouskas, G., Perros, H.: A tutorial on optical networks. ACM Advanced lectures on networking (2002)
4. ITU-T: Architecture of Optical Transport Network. (2001)
5. Ramamurthy, R., Mukherjee, B.: Fixed-alternate routing and wavelength conversion in wavelength-routed optical networks. IEEE/ACM Transactions on Networking **10** (2002)
6. Zang, H., al: Path-protection and wavelength assignment in WDM mesh networks under duct-layer constraints. IEEE/ACM Transactions on Networking **11** (2003)
7. Chlamtac, I., Ganz, A., Karmi, G.: Lightpath communications: An approach to high-bandwidth optical wans. IEEE Transactions Communications **40** (1992) 1171–1182
8. Li, G., Simha, R.: Routing and wavelength assignment by partition coloring. In: MIC2003: The Fifth Metaheuristics International Conference, Kyoto, Japan (2003)
9. Karasan, E., Ayanoglu, E.: Effects of wavelength routing and selection algorithms on wavelength conversion gain in WDM optical networks. IEEE/ACM Transactions on Networking **6** (1998)
10. Brelaz, D.: New methods to color the vertices of a graph. Communications of the ACM **22** (1979) 251–256
11. Caenegem, V., al: Dimensioning of survivable WDM networks. IEEE Journal on Selected Areas in Communications **16** (1998) 1146–1157

A New Path Protection Algorithm for Meshed Survivable Wavelength-Division-Multiplexing Networks

Lei Guo, Hongfang Yu, and Lemin Li

Key Lab of Broadband Optical Fiber Transmission and Communication Networks,
University of Electronic Science and Technology of China, Chengdu, 610054, P.R. China
{lguo, yuhf, lml}@uestc.edu.cn

Abstract. In this paper, we investigate the relationship between the resource sharing degree and the protection ability, and propose a new path protection algorithm (PPA) to protect the multi-link failures in survivable wavelength-division-multiplexing (WDM) mesh networks. Under dynamic traffic with different load, the simulation results show that PPA not only provide 100% protection for the single-link failure but also has higher resource utilization ratio than the dedicated-path protection (DPP) and higher protection ability than the shared-path protection (SPP) as the multi-link failures occur. With configuring different parameter, PPA can determine the appropriate tradeoffs between the resource utilization ratio and the protection ability.

1 Introduction

In wavelength-division-multiplexing (WDM) networks, a wavelength channel has the transmission rate of over gigabits per second [1]. If the fiber links fail, a lot of connection streams to be blocked. Protection design is very necessary for WDM optical networks, and many previous works have proposed their algorithms to protect the single-link failure [1-4].

1.1 Protection for Single-Link Failure

A conventional protection algorithm, which is called dedicated-path protection (DPP), computes a working path and a link-disjoint backup path for a connection request. The reserved backup wavelengths on a backup path cannot be shared with other backup paths. Then, the DPP has low resource utilization ratio.

Another protection algorithm, which is called shared-path protection (SPP), also computes a working path *wp* and a link-disjoint backup path *bp* for a connection request. Differing from the DPP, the reserved backup wavelengths on *bp* for the SPP can be shared with other backup paths if their corresponding working paths are link-disjoint with *wp*. Then, SPP has high resources utilization ratio.

1.2 Protection for Multi-link Failures

The amount of users increasing heavily leads to the size of networks keeping enlarging, and many heterogeneous networks interconnecting leads to more and more com-

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 68–75, 2005.

plicated structure of networks. Then, the probability of risks become much higher, and the protection design for the multi-link failures must been considered in WDM optical networks [5-7]. Previous algorithms, which are the DPP and the SPP, can completely protect the single-link failure. As the multi-link failures occur, the survivable situations of the DPP and SPP are illustrated as follows.

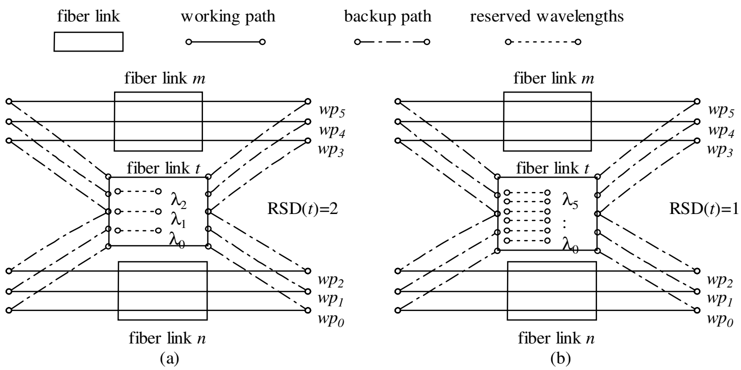


Fig. 1. Survivable situations as the multi-link failures occur for the (a) shared-path protection (SPP) and the (b) dedicated-path protection (DPP)

Unprotected Situation: We can see that, in Fig. 1, the backup paths all traverse the fiber link *t*. If fiber links *n* and *t* (or links *m* and *t*) fail simultaneously, then the working paths wp_0, wp_1, and wp_2 (or wp_3, wp_4, and wp_5) cannot be protected; that is, the connections will be blocked if their working and backup paths both fail.

Protected Situation: We assume the fiber link *t* will not fail. If fiber links *m* and *n* both fail, 1) in Fig. 1(a), the working paths wp_0, wp_1, and wp_2 can be protected, and but the wp_3, wp_4, and wp_5 cannot be protected because λ_0, λ_1, and λ_3 have been used by the wp_0, wp_1 and wp_2; 2) in Fig. 1(b), all working paths can be protected because there are enough reserved wavelengths on the link *t*.

Obviously, the DPP has higher protection ability and lower resource utilization ratio than the SPP; that is, for providing higher protection ability, we need reserve more backup wavelengths. Then, we can consider adjusting the reserved backup wavelengths to add a valuable elasticity between the protection ability and the resource utilization ratio in protecting the multi-link failures.

1.3 Proposed Algorithm

Because the DPP has low resource utilization ratio and high protection ability, and the SPP has high resource utilization ratio and low protection ability, so, we consider seeking for a tradeoff performance between the DPP and the SPP. We present a new path protection algorithm (PPA) that can adjust the reserved backup wavelengths according to the current resources sharing degree (RSD). When we assign the reserved wavelengths, if the RSD is bigger than the value of k that is a parameter, then we shall increase the backup wavelengths until the RSD is no more than the k. In Simulations, we can see that the PPA has higher protection ability than the SPP and higher resource utilization ratio than the DPP. With configuring different the k, the PPA can determine the appropriate tradeoffs between the resource utilization ratio and the protection ability. If we extend the link-disjoint to the shared-risk link group (SRLG) disjoint [8], the PPA for protecting the multi-link failures can easily be extend to protect the multi-SRLG failures [9].

The rest of this paper is organized as follows. Section 2 elaborates on network model, reserved backup wavelengths, and the link-cost assignment. The processes of PPA are detailed described in Section 3. Simulation results and analysis are presented in Section 4. Section 5 is for conclusions.

2 Problem Analysis

2.1 Network Model

Define a network topology G (N, L, W) for a given WDM mesh network, where N is the set of nodes, L is the set of bi-directional links, and W is the set of available wavelengths per fiber link. $|N|$, $|L|$ and $|W|$ denote the node number, the link number and the wavelength number, respectively. Connection request arrives dynamically, and there is only a connection request arrival at a time, defined by $r(s, d)$, where s, $d \in N$ denote the source node and destination node. A requested bandwidth is a wavelength. In this paper, we allow wavelength conversion. The least-cost path algorithm, which is Dijkstra's algorithm, applies to compute the routes. In the following, we introduce some notations.

l is a bi-directional fiber link in G. c_l is the basic cost of link l, and it is determined by many factors, such as physical length of the fiber link, installation cost of the fiber link, and so on. c'_l is the current cost of link l, and it is determined by the current network state. a_l is the number of wavelengths already consumed on link l. F_l is the number of free wavelengths on link l, and $a_l + F_l = |W|$ should be satisfied. W_l is the number of wavelengths already consumed by working paths on link l. R_l is the number of reserved wavelengths on link l, and $R_l + W_l = a_l$ should be satisfied. TR_l is the number of reserved wavelengths of temporary record on link l. wp_n is the working path for connection request n. bp_n is the backup path for wp_n. $|S|$ denotes the number of elements in the set S.

WV_l^n is a boolean variable that is defined as Eq. (1). If the connection n is protected by link l (namely, bp_n traverses link l), then WV_l^n =1. Otherwise, WV_l^n =0.

$$WV_l^n = \begin{cases} 1 & if \quad l \in bp_n \\ 0 & otherwise \end{cases} \tag{1}$$

WV_l denotes the number of all connections protected by link l, defined as

$$WV_l = \sum_{k=0}^{n} WV_l^k \qquad \forall l \in L \tag{2}$$

2.2 Reserved Backup Wavelengths

For arbitrary link l, we assume $v_l = \max(v_l^e)$ ($\forall e \in L$, $e \neq l$), where v_l^e denotes the number of working paths that traverse link e and are protected by link l (namely, their corresponding backup paths traverse link l). We let $TP_l = v_l$, and the resource sharing degree for link l is defined as

$$RSD(l) = \frac{WV_l}{TP_l} \tag{3}$$

We define a parameter k ($k \in [1, \infty]$), if $RSD(l) > k$, then we increase the TP_l until $RSD(l) \leq k$. It is obviously that, in Fig.1, bigger k means more reserved backup wavelengths and more connections will be protected, that is, higher protection ability.

2.3 Link-Cost Assignment

Assume that a connection request n arrives at a given time. First, we adjust the link-cost according to Eq. (4) and compute the least-cost working path.

$$c_l' = \begin{cases} \infty & if \quad F_l = 0 \\ c_l & otherwise \end{cases} \tag{4}$$

If the working path has been found, we adjust the link-cost according to Eq. (5) and compute the link-disjoint and least-cost backup path, where ε is a sufficient small constant (in simulations, we assume $\varepsilon = 1$) and $U = \{k;\, k \in wp_n\}$.

$$c_l' = \begin{cases} \infty & if \quad l \cap U \neq \varnothing \quad or \quad F_l + P_l < TP_l \\ \varepsilon & if \quad P_l \geq TP_l \\ c_l & otherwise \end{cases} \tag{5}$$

3 Proposed Algorithm

3.1 Processes and Complexity of PPA

Step 1: Wait for a connection request arrival. If a connection request arrives, then go to Step 2. Otherwise, update the network's state and go back to Step 1.

Step 2: Adjust the link-cost according to Eq. (4) and compute the working path. If succeed to find the working path, then go to Step 3. Otherwise, block the connection request, update the network's state and go back to Step 1.

Step 3: According to the k and Eq. (3), compute the TP_l for each link l. Adjust the link-cost according to Eq. (5) and compute the backup path. If succeed to find the backup path, then accept the connection request, update the network's state and go back to Step 1. Otherwise, block the connection request, update the network's state and go back to Step 1.

The complexity of PPA mostly depends on running the times of Dijkstra's algorithm. The complexity of Dijkstra's algorithm is $O(|N|^2)$. Analyzing the process, the complexity of PPA is approximately $O(2|N|^2)$ for a connection request.

3.2 Performance of Algorithm

The resource utilization ratio (RUR) is calculated in Eq. (6) below. It is obviously that a smaller value of RUR means that we need to assign fewer resources and also means a smaller backup bandwidths reserve on all the backup paths and a higher degree of spare capacity sharing, that is, a higher resource utilization ratio. Higher resource utilization leads to lower traffic blocking because more free resources can be used in the following traffic routing.

$$RUR = \sum_{l \in L} R_l / \sum_{l \in L} W_l \tag{6}$$

The requests blocking ratio (BR) is the ratio of $|R|$ to $|V|$, where R is the set of connection requests that are being abandoned by the network and V is the set of all connection requests that have arrived at the network. In the case of dynamic traffic, the BR can approximately reflect the effectiveness of resource utilization, and a smaller BR means a higher resource utilization ratio.

The protection ability (PA) is the ratio of $|D|$ to $|H|$, where D is the set of protected connections as the failures occur and H is the set of connections that are holding on the network. It is obviously that a bigger value of PA means that more connections will be protected in failures, that is, higher protection ability.

4 Simulations and Analysis

We simulate a dynamic network environment with the assumptions that connection requests arrival according to an independent Poisson process with arrival rate β, and the connections holding time is negative exponentially distributed 1/μ, so the network load is β/μ Erlang. We assume μ=1 and each requested bandwidth is a wavelength. If the connection fails to establish, the network abandons it immediately, i.e., there are no waiting queues. The test network is shown in Fig. 2, where nodes, which have wavelength conversion capacities, are interconnected by bi-directional fiber links that the basic link-cost is 10. The number of wavelengths of per fiber is assumed to be five. The value of k is selected form [1, ∞]. We compare the performance of the PPA with the DPP and the SPP [1-4]. All simulation results are averaged via simulating 10^6 connection requests.

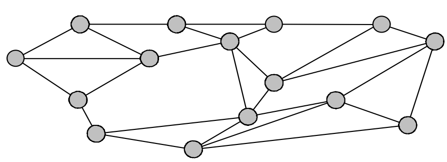


Fig. 2. National network topology of America

In Figs. 3 and 4, we assume A is a sufficient large constant, that is, $A \rightarrow \infty$. It is shown in Fig.3 (a) that the RUR decreases and gradually becomes invariable as k increases with different load (10-35 erlang), and this means the resource utilization ratio is

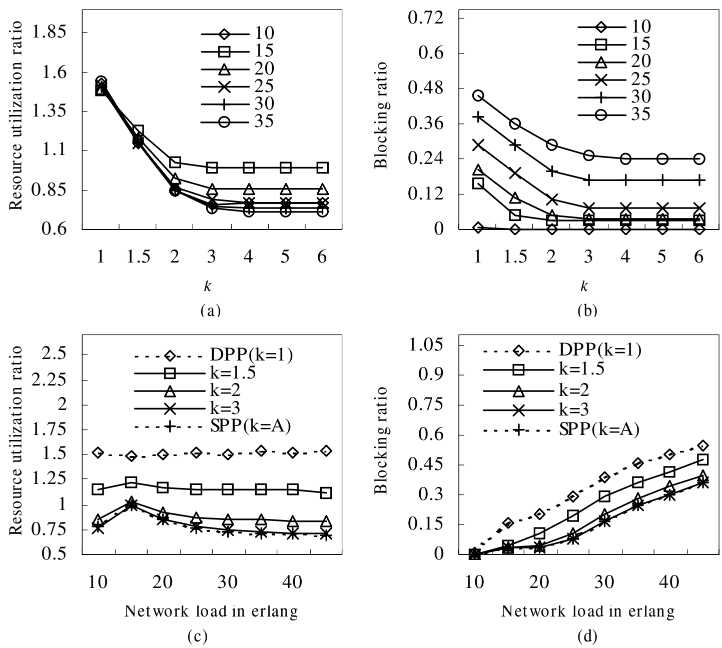


Fig. 3. With different load, (a) the RUR versus the *k*, and (b) the BR versus the *k*. With different *k*, (c) the RUR versus network load, and (d) BR versus network load

improved and gradually reaches its best performance. When $k\to\infty$, the resource utilization ratio is the highest. We can see in Fig.3 (b) that the BR decreases and gradually becomes invariable as *k* increases with different load, and this means the blocking ratio is gradually reduced and gradually reaches its best performance. When $k\to\infty$, the blocking ratio is the lowest. The reason for this is that, when *k* is bigger, the resource utilization ratio is higher, and more free wavelengths can be used by the following connection requests, and then less connection will be blocked.

According with Eq. (3) and Fig. 1, we can find, as k =1, the backup wavelengths cannot be shared, and the PPA is equivalent to the DPP; as $k\to\infty$, the backup wavelengths can be shared, and the PPA is equivalent to the SPP; as *k* is equal to a finite constant from $(1, \infty)$, the backup wavelengths can be partially shared, and the performance of the PPA can be tradeoffs between the DPP and the SPP.

In Fig.3 (c) and (d), it is shown that, as k=1, the performances of the RUR and the BR of PPA are the worst; as $k\to\infty$, the performances of the RUR and the BR are the best; as k=1.5, 2, and 3, the performances of the RUR and the BR are tradeoffs between the best and the worst.

In Fig. 4, we can see that, with different value of *k*, PPA can 100% protect the single-link failure. We also see that, as the multi-link failures occur, the performance of the PA is the best as k=1, it is the worst as $k\to\infty$, and it is tradeoffs between the worst and the best optimal as k = 1.5, 2, and 3. The reason for this is that, there are more

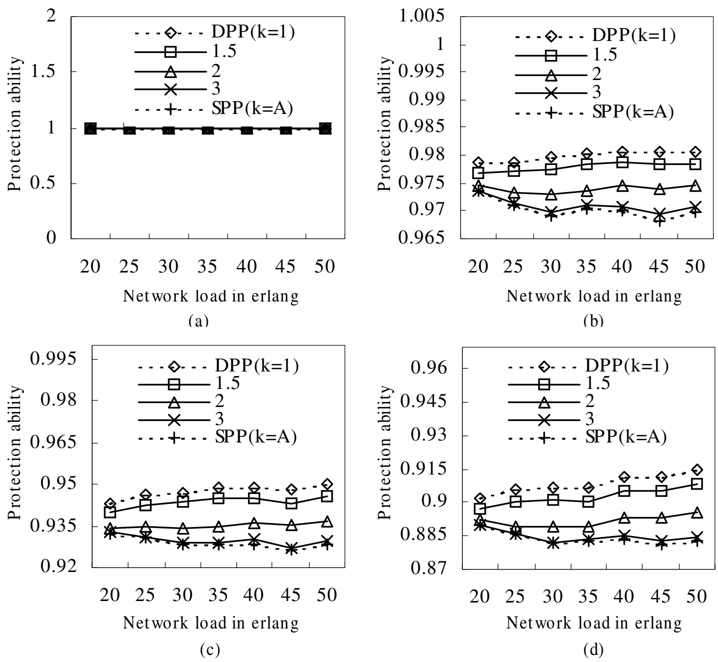


Fig. 4. The PA versus network load, as (a) a random link fail, (b) two random links fail, (c) three random links fail, and (d) four random links fail

reserved wavelengths as k=1, and more connections can switch their traffics on their backup paths as the failures occur (see **protected situation** in section 1.2), and then the protection ability is higher. As $k\to\infty$, there are fewer reserved wavelengths that can be used by the failed connections, and then the protection ability is lower. As k is equal to a finite constant from $(1, \infty)$, the reserved wavelengths are tradeoffs between the k=1 and the $k\to\infty$, and then the protection ability is also tradeoffs between the best and the worst.

5 Conclusion

In this paper, we investigate protection for the multi-link failures in survivable WDM mesh networks, and propose a new path protection algorithm (PPA). The simulation results show that PPA can provide 100% protection for the single-link failure and has higher resource utilization ratio than the DPP and higher protection ability than the SPP as the multi-link failures occur. With configuring different k, PPA can determine the appropriate tradeoffs between the resource utilization ratio and the protection ability.

Acknowledgment

This research was supported by National Natural Science Foundation of China (NSFC) under grants 60302010.

References

1. Ramamurthy S., Sahasrabuddhe L., Mukherjee B. Survivable WDM mesh networks. J. Lightwave Technol. 21 (2003) 870 –883.
2. R. He, H. Wen, L. Li, G. Wang. Shared sub-path protection algorithm in traffic-grooming WDM mesh networks. Photon. Netw. Commu. 8 (2004) 239-249.
3. H. Wen, L. Li, R. He, et al. Dynamic grooming algorithms for survivable WDM mesh networks. Photon. Netw. Commun. 6 (2003) 253-263.
4. C. Ou, J. Zhang, H. Zang, et al. New and improved approaches for shared-path protection in WDM mesh networks. J. Lightwave Technol. 22 (2004) 1223-1232.
5. B. G. Jozsa, D. Orincsay, A. Kern. Surviving multiple network failures using shared backup path protection. in Proc. IEEE ISCC, (2004) 1333 -1340.
6. W. S. He, and A.K. Somani. Path-based protection for surviving double-link failures in mesh-restorable optical networks. in Proc. IEEE GLOBECOM, (2003) 2558 – 2563.
7. L. Guo, H. Yu, and L. Li. Double-link failure protection algorithm for shared sub-path in survivable WDM mesh networks. Chin. Opt. Lett. 2 (2004) 379-382.
8. L. Guo, H. Yu, and L. Li. Joint routing-selection algorithm for a shared path with differentiated reliability in survivable wavelength-division-multiplexing mesh networks. Opt. Express, 12 (2004) 2327-2337.
9. L. Guo, H. Yu, T. Zhou, and L. Li. Dynamic shared-path protection algorithm for dual-risk failures in WDM mesh networks. in Proc. IEEE ICPP Workshops, (2004) 394-398.

Application Area Expansion in Quasi-Millimeter Wave Band Fixed Wireless Access System

Shuta Uwano and Ryutaro Ohmoto

NTT Access Network Service Systems Laboratories, NTT Corporation,
1-1 Hikari-no-oka Yokosuka Kanagawa 239-0847, Japan
{s-ueno, ohmoto}@ansl.ntt.co.jp

Abstract. NTT developed the Wireless IP Access System (WIPAS), a point-to-multipoint fixed wireless access (FWA) system utilizing the 26-GHz frequency band for home and SOHO users to provide broadband Internet access service. This service is a best-effort type IP service with the transmission rate of 80 Mbit/s and with the maximum Ethernet frame transmission rate of 46 Mbit/s. Recently, with the aim of further expanding the applications of WIPAS, advanced functions have been developed. The WIPAS service area has been expanded by applying WIPAS technologies in a variety of forms. The radio entrance line is a point-to-point connection using WIPAS equipment that can be applied to extend the broadband access service area where it is impossible or difficult to provide service using optical fibers. This paper describes the service concepts, configurations, and technologies of WIPAS, and illustrates technologies pertaining to added functions and extending the service area of this system.

1 Introduction

NTT developed the Wireless IP Access System (WIPAS), a low cost FWA system for home and SOHO users. WIPAS is a point-to-multipoint (P-MP) system that uses the 26-GHz frequency band for use by FWA services in Japan. The transmission capacity of this system is 80 Mbit/s and the maximum transmission rate of an Ethernet frame is 46 Mbit/s. IP services using WIPAS started in 2003 in Japan in urban and suburban residential areas. We reconsidered and revised the requirements for the FWA system to reduce the cost of the equipment and its installation, and to downsize the equipment [1].

In order to enhance the performance and reliability of this system, and to expand the application area more extensively, WIPAS requires additional functions that can respond to a variety of user requests. For this purpose, the system was upgraded by adding an adaptive modulation scheme and a minimum bandwidth guarantee function. Furthermore, the radio entrance line, which is advantageous because it can be constructed easily at low cost compared to using a wired line, was implemented in order to extend the range of the access service area. The radio entrance line is achieved in the form of a point-to-point (P-P) cascade connection using sets of WIPAS equipment.

This paper is organized as follows. Section II illustrates the service concept of WIPAS. Section III describes the system and equipment configurations. Section IV introduces further advanced functions and Section V describes the service area extension technologies of WIPAS. Finally, Section VI presents our conclusions.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 76–83, 2005.

2 Service

Figure 1 shows the service concept of WIPAS. Services that use this system are provided to home users and SOHO users. An access point (AP) is installed on a telephone pole or a rooftop of a building, and is connected to the core network using an optical fiber cable. A wireless terminal (WT) is installed on the user premises, apartment, or office building, and is connected to the user PC using an Ethernet cable. The transmission speed between the AP and WTs is 80 Mbit/s at maximum. Although WIPAS initially provided a best-effort-type IP service, it currently provides a new Quality of Service (QoS) minimum bandwidth guarantee service. The WT shares the wireless bandwidth fairly with other WTs that are connected to the same AP. WIPAS is applied to an area where it is impossible or difficult to lay optical fiber cables for an access network, but it is a system that combines wireless technology with an optical network, and is a complement service to fiber to the home (FTTH) service.

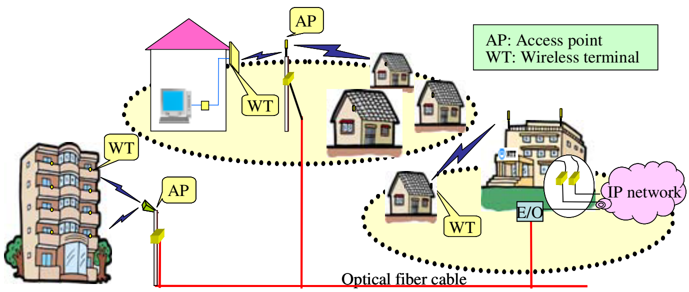


Fig. 1. Service Image of WIPAS

3 System and Equipment Configurations

3.1 System Concepts

The main specifications of WIPAS are given in Table 1. The transmission/access scheme is TDMA/TDD, and the modulation scheme uses QPSK or 16QAM in this system. The transmission capacity is 40 Mbit/s for QPSK,and 80 Mbit/s for 16QAM. The transmission power is 14 dBm for QPSK, and 11.5 dBm for 16QAM to reduce the equipment cost. A WT performs automatic transmission power control (ATPC) at the AP to adjust the reception level to the standard reception level. The control range of ATPC is 20 dB. One AP can manage up to 239 WTs.

3.2 Equipment Configuration

The equipment configuration of WIPAS is shown in Fig. 2. The AP consists of an AP- Radio frequency unit (RFU), which includes an antenna, RF and IF modules, and an AP- Interface unit (IFU) that includes a modem, TDMA control, and MAC processing modules. The AP-RFU uses omni, sector, horn, or Cassegrain antennas depending on the service type. The AP-IFU has an optical interface (100BASE-FX) and

Table 1. WIPAS Specifications

Frequency band		26-GHz band
Transmission		TDMA/TDD
Modulation		QPSK, 16QAM Adaptive modulation scheme
Wireless transmission rate (Maximum transmission rate of Ethernet frame)		QPSK: 40 Mbit/s (23 Mbit/s) 16QAM: 80 Mbit/s (46 Mbit/s)
TX power		QPSK: 14 dBm / 16QAM: 11.5 dBm
Number of WTs		Max. 239 WTs per AP
Network interface		100BASE-FX
User interface		100BASE-TX / 10BASE-T
MAC layer processing		VLAN: IEEE802.1Q
Antenna type	AP	Omni, Sector, Horn, Cassegrain antenna
	WT	Planar antenna
Cell radius		QPSK:700 m / 16QAM:400 m
QoS		Fairness assignment among WTs Minimum bandwidth guarantee

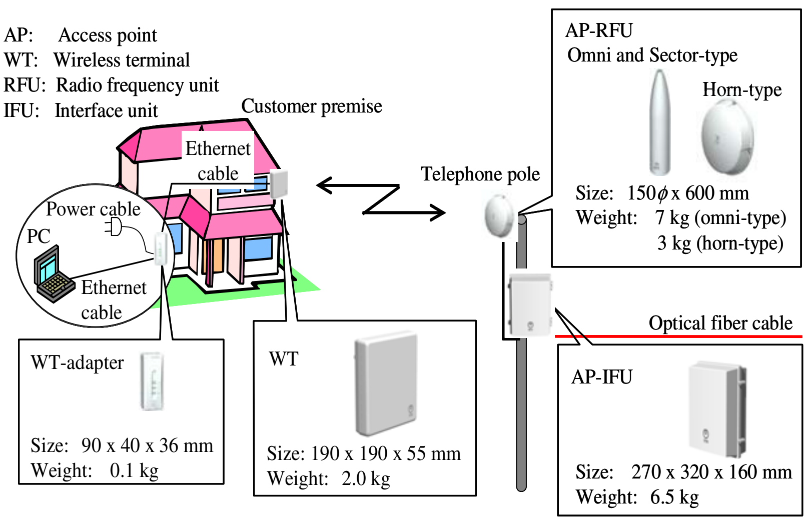


Fig. 2. Equipment configuration of WIPAS

an electrical interface (100BASE-TX) to connect to the core network. The WT comprises an antenna, RF, IF, modem, TDMA control, MAC processing modules, a WT adapter that has the functions of an Ethernet interface (10/100BASE-TX), power supply, and alarm indicators. The WT equipment connects to the WT adapter using an Ethernet cable (multiplexing power). The WT has a planer type antenna.

3.3 Radio Unit and Antenna

The antenna of the WT is a slotted-waveguide-array flat antenna [2, 3]. We downsized the WT equipment by using these fundamental technologies. In the AP, we separated the AP-RFU containing the antenna and RF and IF modules from the AP-IFU containing baseband modules, because the antenna of the AP-RFU is changed according to the service type. We downsized the AP-IFU by using two ASICs for baseband processing. One, which is commonly used by WTs, has functions of a modem, and the other one has a function for TDMA control and MAC processing. The cost of the WIPAS equipment is approximately ten times less than that of conventional systems, and its weight is approximately five times less than that of conventional systems.

4 Further Advanced Functions

In this section, we describe two additional functions that were developed recently to enhance the adaptive flexibility of WIPAS.

4.1 Adaptive Modulation Scheme

We newly developed the adaptive modulation scheme shown in Fig. 3, which automatically switches between 16QAM and QPSK according to the wireless transmission conditions. The transmission distance for 16QAM is 700 m in clear atmosphere, which is equivalent to that of QPSK under rainfall conditions. The transmission distance for 16QAM should be shortened to within 400 m since it is required by the system margin for rainfall. However, the novel adaptive modulation scheme enables the transmission distance to be extended to 700 m, while maintaining 80 Mbit/s of 16QAM in clear atmosphere and 40 Mbit/s of QPSK during rainfall.

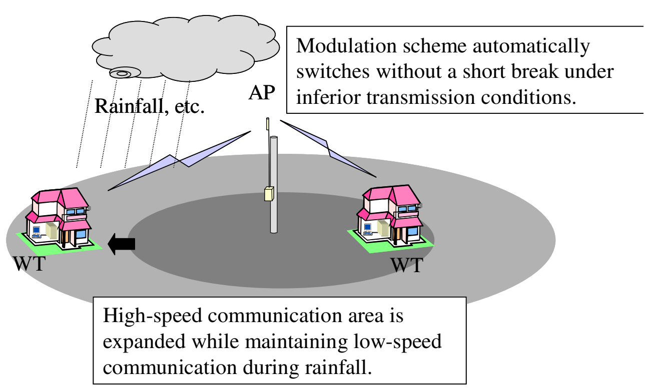


Fig. 3. Adaptive modulation scheme

The WT periodically measures the link quality of the user data in the downlink payload (16QAM), and reports the measurement results to the AP. The AP determines the modulation scheme based on the link quality report received from the WT. If the link quality is degraded by rainfall, the AP immediately switches the modulation scheme to QPSK. If the link quality recovers, the AP switches back to 16QAM after verifying the stability a few times.

4.2 Minimum Bandwidth Guarantee Service

By adding a preferred bandwidth assignment function, WIPAS provides a minimum bandwidth guarantee service, while providing the best-effort-type IP services to other users at the same time. When the requested bandwidth is less than the guaranteed bandwidth, the requested bandwidth is assigned, and the surplus bandwidth is reassigned to other WTs. Furthermore, the rest of the guaranteed bandwidth assignment is distributed fairly to all the WTs. Therefore, the bandwidth assignment is carried out in a flexible manner without wasting the bandwidth (see Fig. 4). The bandwidth-guaranteed-type service can be provided to special users such as business users or heavy traffic users while maintaining a mixture with best-effort-type service for general users.

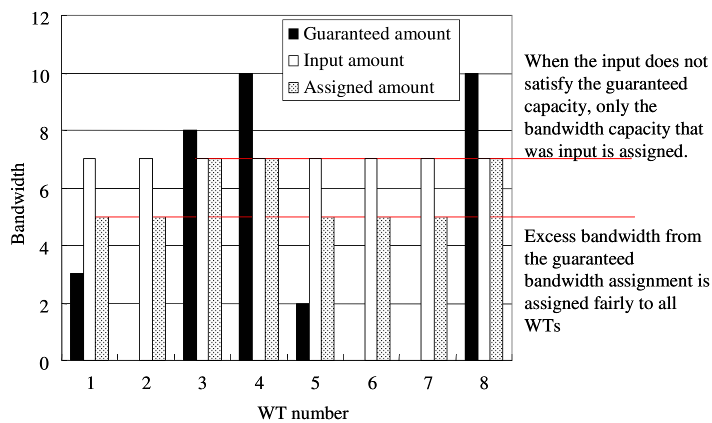


Fig. 4. Bandwidth assignment (example)

5 Service Area Extension

In this section, we introduce the technologies for extending the service area of WIPAS in a variety of forms.

The WIPAS service area is expanded further by connecting several APs and WTs as shown Fig. 5. Figure 5(a) shows the radio entrance line type. AP1 communicates with WT1 using a high-gain Cassegrain antenna. AP2, which is used in combination with WT1, connects to AP1 through WT1 and constructs a cell at a point distant from the optical fiber. Figure 5(b) shows the point-to-point (P-P) cascading type. A couple of WTs (WT3 and WT4) operate as repeaters and enable WT5 to connect to AP3 over two hops. Since both antennas of AP1 and WT1 have extremely high gain of more than 31 dBi, the transmission distance between the two is longer than 2 km for 16QAM. This is similar to the transmission distance between WT4 and WT5. Thus, the radio entrance line can be applied to increase the range of the access service area easily at low cost compared to extending optical fiber cable.

Figure 6 shows a practical example of dense deployment of WIPAS in a suburban residential area, Haramachi city in Japan. Because the AP employs an omni-directional antenna, the roughly circular cells overlap in the deployment area. The radio frequency channels are allocated optimally with no interference between the cells, and the cells are designed on the basis of the line-of-sight (LOS) calculation results derived from the building and vegetation information with a high degree of accuracy. The APs are generally connected to the IP network through the optical fibers of the municipal intranet. However, in areas that are distant from the optical fibers, supplementary APs can be installed by applying the radio entrance line, and the gaps in the service area are covered by auxiliary cells such as the gray cells in Fig. 6.

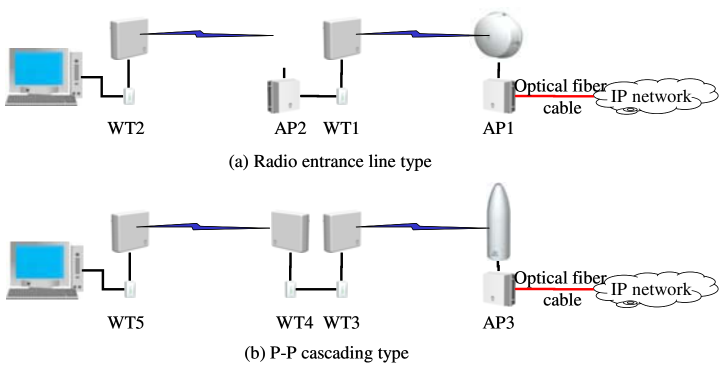


Fig. 5. Service area expansion scheme

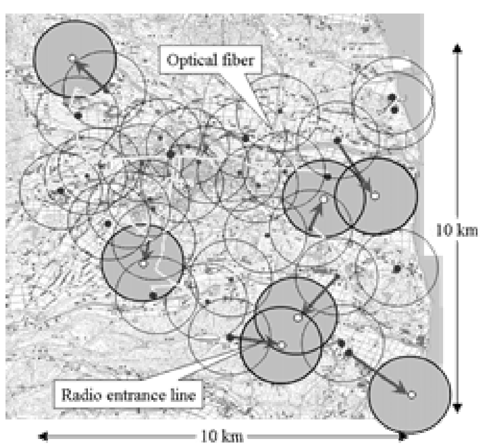


Fig. 6. Example of cell deployment design for Haramachi city in Japan

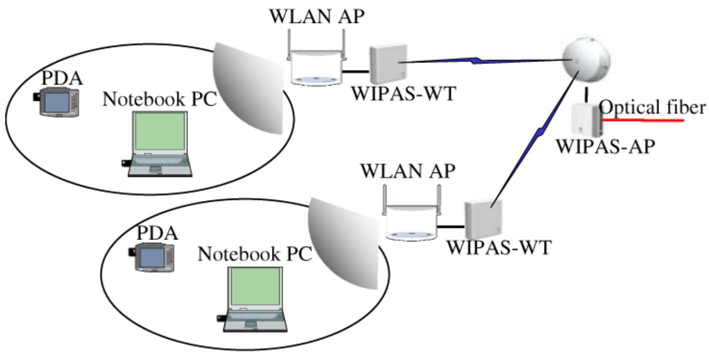


Fig. 7. Radio entrance line of hotspot service

Figure 7 shows the case in which WIPAS is applied to the radio entrance lines as an access line to the access point in a hotspot service using wireless LANs. Therefore, the access point of a wireless LAN can be installed at the point where it is hard to lay a cable and hotspot service area can be extended more easily. We present a model in which WIPAS is applied to the radio entrance line of a hotspot service. The service area is the Twin Ring Motegi circuit in Japan where motor races are held. Streaming live videos from cameras around the circuit are distributed from the wireless LAN AP (IEEE802.11 b/g) so that anyone with a PDA and a wireless card can view the live race. WIPAS provides the radio entrance line between the multicast server and the multiple wireless LAN APs.

6 Conclusions

NTT developed the Wireless IP Access System, a P-MP type FWA system which is a complement service to FTTH service. WIPAS uses the 26-GHz frequency band, and has the transmission capacity of 80 Mbit/s. We newly added an adaptive modulation scheme and minimum bandwidth guarantee function, and achieved enhanced system performance. We introduced examples of broadband IP services employing WIPAS, and presented application technology to extend the service area. The radio entrance line can be applied easily to increase the range that WIPAS covers, or it can be used as a backhaul line to an access point in a hotspot service using wireless LANs, so the hotspot service area can be extended more easily. In the future, we will improve construction technology to expand the application area of WIPAS such as hotspot backhaul in rural areas far from optical networks by connecting to satellite communications.

References

1. K. Nidaira, T. Shirouzu, M. Baba, K. Inoue: Wirelss IP access system for broadband access services. IEEE International Conference on Comminications, WC15-1.pdf, (2004)

2. Y. Kimura, Y. Miura, J. Hirokawa, M. Ando: A low-cost and compact wireless terminal with an alternating phase fed single-layer waveguide array for 26GHz fixed wireless access systems. JINA2002 conf. dig., vol.2 (2002) 455-458
3. Y. Miura, T. Shirosaki, T. Taniguchi, Y. Kazama, Y. Kimura, J. Hirokawa, M. Ando: A low-cost and very small wireless terminal integrated on the back of a flat panel array for 26GHz band fixed wireless access systems. IEEE Topical Conf. on Wireless Commun. Tech. Dig. (2003) s21p08.pdf

A Robust Service for Delay Sensitive Applications on a WLAN

Fanilo Harivelo and Pascal Anelli

IREMIA, Université de La Réunion BP 7151,
15 Avenue R. Cassin, 97715 Saint Denis Messag 9, France

Abstract. Technological advances in mobile terminals and the large spreading of Internet have led to the growing need of a certain level of a quality of service for the applications. Wireless networks characteristics make this task difficult. Thus, the classical protocols and models of QoS became inaccurate in this type of networks. This article presents a mechanism that guarantees a service corresponding to the Expedited Forwarding PHB (Per Hop Behavior) in a wireless network. Simulations under NS-2 are carried out to evaluate the performances of the solution.

1 Introduction

With the proliferation of mobile terminals and the popularity of Internet access, the IEEE 802.11 Working Group has proposed a standard [1] for wireless local area networks. It proposes two access methods: DCF (Distributed Function Coordination) and PCF (Polling Function Coordination). DCF is available in infrastructure mode as well as in ad hoc mode and is based on CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) method. Before initiating a transmission, a station senses the medium and executes an exponential backoff algorithm to avoid collisions. With DCF mode, no priority exists among the stations. Besides, a station with a low transmission rate, while capturing the channel, can penalize the other stations on the long run [2]. PCF method tackles with delay sensitive data transmissions and is limited to the infrastructure mode. In PCF, time is divided into superframes. A superframe consists of a period called CFP (Contention Free Period) during which the coordinator, generally the access point, polled each station if it has packets to send, and a CP (Contention Period) period during which DCF mode is used as access method. PCF is complex and some ambiguities remain in its specification. This article proposes a service for delay sensitive application aiming to support flows marked as EF (Expedited Forwarding) according to DiffServ Architecture [3]. This service is provided in a wireless network without access point. In the following, the considered network consists of stations having the same diffusion domain and the hidden station problem is supposed solved by a mechanism such as RTS/CTS. Section 2 gives a state of the art of QoS in wireless networks. Section 3 details the proposed architecture, which will be validated in section 4. The results are summarized in section 5.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 84–91, 2005.

2 Related Works

Many studies have been drawn to introduce QoS in the wireless networks. The IEEE 802.11e working group has defined improvements [4] to IEEE 802.11 standard which introduce two new access methods, namely, EDCF (Enhanced DCF) and HCF (Hybrid Coordination Function). In EDCF which derives from DCF, QoS is obtained by the use of eight levels of TCs (Traffic Categories). At the MAC level, the packets are transmitted via separate instances of the backoff algorithm, each instance having parameters set according to the priority level. Although EDCF ensures a better service for higher priority traffic, it does not offer any quantitative guarantee. Moreover, under high load, many collisions may occur even for the priority traffic. HCF function adopts the same principle as PCF and allows a hybrid coordinator, localized generally at the access point, to poll the stations having priority traffic for CFP period. Some of the drawbacks of PCF remain with HCF. To mitigate these shortcomings, [5] proposes a mechanism derived from EDCF, called AEDCF (Adaptive EDCF), which takes into account the contention level of the channel. AEDCF adjusts the size of the contention window and the persistence factor according to the number of collisions. [6] introduces a solution to support the real-time traffics. CFP Period will be used for the transmission of real-time traffic while the CP period is exclusively reserved for the Best-Effort traffic. To ensure a better bandwidth usage and to avoid starving the Best Effort traffic, [7] introduces the concept of free space which defines the unused bandwidth by the privileged traffic, that can be recovered by the Best Effort traffic. To a privileged packet can be piggybacked lower priority packets sharing the same next hop. [8] presents an architecture supporting EF and AF PHB (Per Hop Behaviors). EF PHB is ensured by allocating a low IFS to the corresponding stations. To alleviate the contention among EF flows, two jamming sequences are transmitted by each EF station. That which has the longest sequences will access the medium. [9] proposes to support EF PHB in a wireless network This approach consists in setting a map of the bandwidth usage in the network using an exchange of messages and in deducing the local BE traffic rate. For a better use of the resources, the unused bandwidth by the EF traffic is recovered by the BE traffic. The delay constraints are ensured by anticipating possible load increases by the introduction of a thresholds system that allocates a bandwidth margin to EF traffic.

3 Wireless Bandwidth Access Control

Our proposition stands for a limitation of the BE traffic of the network, that aims to guarantee initially fixed bandwidth and delay for the EF class of service. This restriction is done on the basis of the network state and require neither any exchange of messages nor the knowledge of the traffic of the other nodes. Indeed, this is used to prevent the network from congestion and to avoid overload due to signaling mechanism. The traffic control is conveyed to the policy layer in such a way that no queueing delay will be induced to the currently transmitted frame

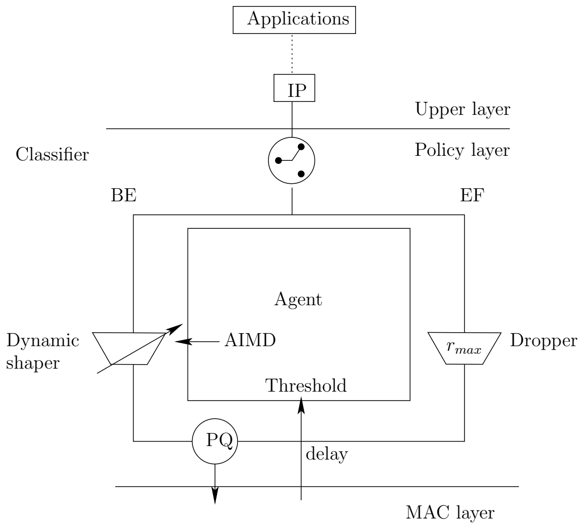


Fig. 1. Global architecture

at the MAC layer. A control function computes the sending rate of the local BE traffic. The EF traffic profiles are distributed to the appropriate stations. The method to distribute these profiles is out of the scope of this paper. However the adopted policy must take care not to exceed a certain ratio of the bandwidth [10]. The conformance to the profile is done using a token bucket and the excess traffic will be dropped. The suggested solution is localized between the MAC layer and the IP layer as shown in the Fig. 1. Each station implements this architecture.

The information obtained from the MAC layer will be used to determine the network state. This disposal is taken to make it possible for the architecture to deal with wireless network characteristics, while allowing the possibility of combination with a MAC level mechanism to accentuate the service differentiation. Indeed, the QoS support provides at the MAC level tackles with the choice of the node which will acquire the medium while an IP level solution defines the packet which will be transmitted within a node [11]. EF and BE packets are handed over to the MAC layer according to a PQ (Priority Queue) scheduling. The BE traffic limitation is done using a dynamic shaper whose parameters result from a congestion avoidance mechanism. This mechanism is highly interrelated with the control function used by the station to increase or to decrease its BE traffic and it is comprised in the agent localized in each station in such a way that each one reacts in the same manner depending on the network state. The agent estimates the network state and allocates the maximum BE sending rate of the station to ensure a high bandwidth usage while guaranteeing low delays. The packets are classified thanks to the DS field of the IP header.

The congestion avoidance control consists of a thresholds system similar as that of [12] and a binary feedback. The network state is provided, periodically every Δt, by the thresholds system. This information is deduced from the response

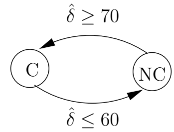


Fig. 2. Thresholds System with 2 states

time of the MAC level, i.e. the delay d taken by a packet to be transmitted, and the initially guaranteed MAC level delay d_{max}. The network load is estimates using the percentage $\hat{\delta} = 100\frac{d}{d_{max}}$. A binary feedback (0 or NC for not congested, and 1 or C for going to be congested) is determined by the stations, so that they can adjust (increase or decrease) their rate r_{BE}, via a control function. If this feedback estimates that the network is not congested, then, the BE traffic rate can increase, otherwise, the BE traffic rate is decreased. The choice of binary feedback is motivated by its simplicity and its efficiency for the resource controller. The thresholds system is used to bring up the measured load so that the network can act before the maximum delay is reached. So the network never enters in congestion. The delay is calculated on the packets successfully received and corresponds to the duration from the handling of the packet by the MAC level and the receipt of the acknowledgment. For example, by considering the model of Fig. 2, let $\hat{\delta}$ equals 75 and the previous state, NC, then the current state will be C, corresponding to a congested network.

However, a question arises on the way by which each station lower its rate in the case of congestion. Indeed, the flows having a high sending rate must decrease more their rate compared to the small flows, in other words, the reduction of the rate must be proportional to the rate. This is done by choosing a multiplicative function for the decrease. A similar consideration has to be made regarding the increase. The fairness constitutes the only condition required for BE traffic. The sharing of the bandwidth must be fair among the stations generating BE traffic and independent of the rate currently generated by each source. An additive function is appropriate for the increase in the rate. The choice of AIMD (Additive Increase Multiplicative Decrease) is judicious insofar as [13] shows that this algorithm ensures fairness and convergence. The AIMD control function is summarized in the following expression:

$$r_{be}(t) = \begin{cases} r_{be}(t - \Delta t) + r_{AI} & \text{if} \quad state = NC \\ r_{be}(t - \Delta t)/k_{MD} & \text{if} \quad state = C \text{ with } k_{MD} \in \mathbb{R} \text{ and } k_{MD} > 1 \end{cases} \tag{1}$$

in which r_{AI} et k_{MD} correspond respectively to the increment value and the decrease factor of BE traffic rate.

4 Performance Evaluation

The evaluation of the proposed mechanism was carried out with the NS-2 simulator in a IEEE 802.11 network comprising 6 stations, one of which is used as the destination for overall traffic. The capacity of the medium is set to 1 Mbits/s. 4 nodes generate BE UDP traffic with a packet size equals 512 bytes and a rate of 400 kbits/s. One of the nodes moves during simulation and becomes out of reach of the other nodes. The EF traffic consists of an MPEG encoded movie.

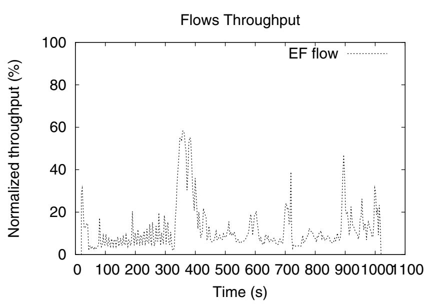
Three cases are considered:

- The first case evaluates the performances of the EF flow when it is the only one being in activity.
- The second case defines a common scenario where BE sources transmit until making network congested.
- The last case corresponds to the use of the proposed mechanism in the previous case to ensure a service to EF flow. The maximum delay imposed for packets at the MAC level is set to of 0.056 s with Δt = 40 ms. This choice is based on previous works [9]. The increase is done by an increment of r_{AI} = 400 bits/s and the decrease by a ratio of k_{MD} = 1.5, derived from empirical considerations.

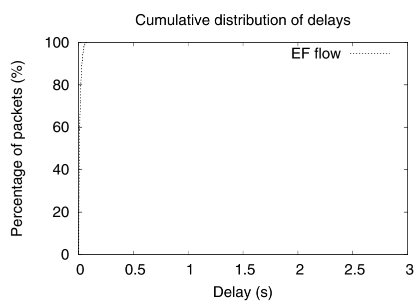
The results are summarized in the table 1 and the curves Fig. 3, 4, 5. In the first case, the bandwidth required by MPEG flow is granted (Fig. 3a) and

Table 1. Statistics of the networks in the three cases

Case	EF alone	WLAN	WLAN + QoS
Bandwidth usage (%)	12.23	69.8	46.63
Max bandwidth usage (%)	58.30	76.23	77.87
# Collisions	0	33505	1716
# Transmitted packets	21267	191650	124493
Exchanged bytes (MB)	14.52	100.23	66.90
# Dropped packets	0	229490	45600
Max EF delay (ms)	80	1700	130
Mean EF delay (ms)	15	90	20
EF standard deviation delay (ms)	13	110	20



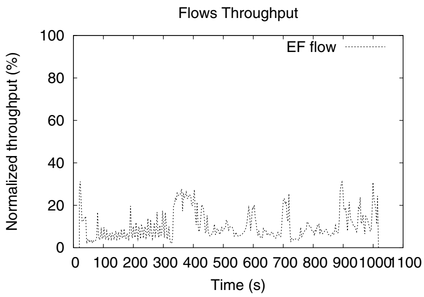
(a) Case 1: EF flow throughput/Time



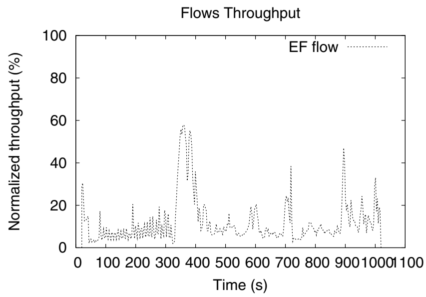
(b) Case 1: Distribution of delays/Delay

Fig. 3. Throughput and delays curves in the first case

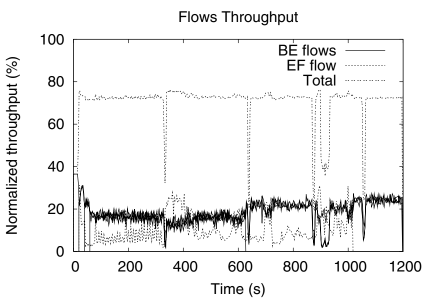
the delay is low (Fig. 3b) with a maximum value of 80 ms. No packet dropped because of the absence of contention on the medium. In the presence of BE flows, the constraints in term of bandwidth (Fig. 4a) and delay (Fig. 5a) are



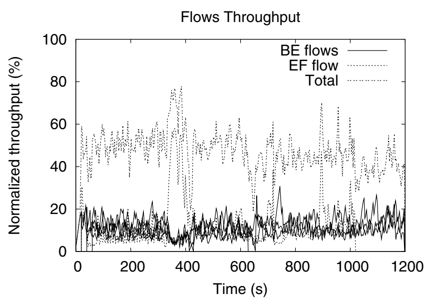
(a) Case 2: EF flow throughput/Time



(b) Case 3: EF flow throughput/Time

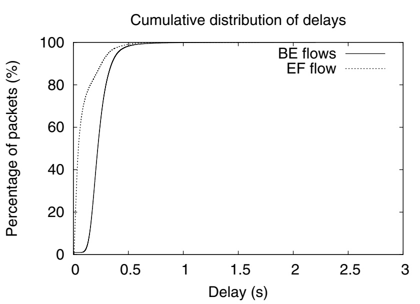


(c) Case 2: Flows throughput/Time

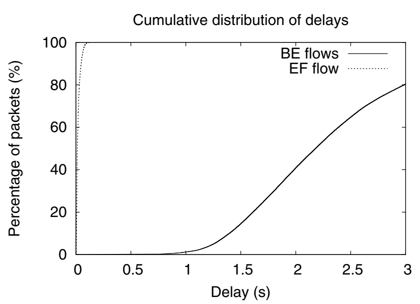


(d) Case 3: Flows throughput/Time

Fig. 4. Throughput curves in the second and the third cases



(a) Case 2: Distribution of delays/Delay



(b) Case 3: Distribution of delays/Delay

Fig. 5. Delays curves in the second and the third cases

not satisfied anymore. The EF packets delay are high with a maximum value of 1700 ms, while the flow experiences significant delay variation. However, the bandwidth usage is high, but this causes a large number of collisions. The proposed mechanism respects the constraints in term of bandwidth (Fig. 4b) and delay (Fig. 5b). Indeed, the maximum delay perceived by EF flow equals 130 ms while delay variation (20 ms against 110 ms in the first case) remains low. The rate control handles correctly the abrupt increases in the load of the EF flow, as that occurring at $t = 320$ s. The EF traffic is completely isolated from BE flows. Indeed, by comparing Fig. 3b and Fig. 5b, one notes that they are nearly the same. The bandwidth usage has been reduced with an average value of 46.63%. A higher EF load would lead to a better utilization ratio because the maximum value reaches 77.87% vs 76.23% in the case without QoS. Besides, the number of collisions decreases significantly (1716 vs 33505).

Thus, the proposed mechanism provides a QoS for an EF real VBR flow while offering fairness for the BE flows.

5 Conclusion

This article presents a robust mechanism for the support of EF PHB in a wireless network and the bandwidth sharing among the BE traffic. The principle consists in avoiding the network to be in a congested state. That is done by the restriction of the BE traffic on the basis of an estimation of the network state thanks to local information, namely, the MAC level delay. The BE traffic rate is decreased or increased according to whether the network is in a congested state or not. A maximum delay, below which a certain level of service can be assured, is initially fixed. To prevent abrupt increase in the load of EF traffic, a thresholds system is set up in order to put a margin on the MAC delay increase. Simulations show that the EF traffic is completely isolated from the BE traffic. The principal advantage of the proposal lies in its ease of implementation and the absence of overload: no signaling is needed. The mechanism works in a totally distributed mode, thus, the motion of a node does not affect the way the other nodes perform their computation. However, the performance of the mechanism can be improved by combining it with a MAC level solution such as IEEE 802.11e. A better bandwidth usage can also be obtained by making the increase and the decrease factor of the BE load variable with the MAC delay. This can be done by using much richer feedback for the congestion avoidance mechanism. A study on the contribution of the solution in the bandwidth allocation in the hidden station case will also be undertaken. Finally, the support of AF PHB would constitute an additional extension of this architecture.

References

1. IEEE 802.11 WG: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications. IEEE standard 802.11, 1999 Edition (1999)

2. Heusse, M., Rousseau, F., Berger-Sabbatel, G., Duda, A.: Performance anomaly of 802.11b. INFOCOM'03, San Francisco, USA (2003)
3. Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., Weiss, W.: An Architecture for Differentiated Services. RFC 2475, IETF (1998)
4. IEEE 802.11 WG: Draft Supplement to Standard for Telecommunications and Information Exchange between Systems-LAN/MAN Specific RequirementsPart 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Medium Access Control (MAC), Enhancements for Quality of Service (QoS). 802.11e Draft 4.1 (2004)
5. Romdhani, L., Ni, Q., Turletti, T.: Adaptive EDCF: Enhanced Service Differentiation for IEEE 802.11 Wireless Ad-Hoc Networks. IEEE WCNC'03 (Wireless Communications and Networking Conference), New Orleans, Louisiana, USA (2003)
6. Velayutham, A., Chang, J.M.: An Enhanced Alternative to the IEEE 802.11e MAC Scheme. www.cs.iastate.edu/ vel/research/E-802.11.pdf (2004)
7. Liu, W., Fang, Y.: Courtesy Piggybacking: Supporting Differentiated Services in Multihop Mobile Ad Hoc Networks. INFOCOM'04 (2004)
8. Banchs, A., Radimirsch, M., Pereza, X.: Assured and Expedited Forwarding Extensions for IEEE 802.11 Wireless LAN . In IWQOS'02 International Workshop on Quality of Service, Monterey, California, USA (2002)
9. Harivelo, F., Grand, G.L., Anelli, P., Wolf, J., Wolfinger, B.: Expedited Forwarding for WiFi. International Symposium on Wireless Communication Systems (2004)
10. Heusse, M., Starzetz, P., Rousseau, F., Berger-Sabbatel, G., Duda, A.: Bandwidth allocation for DiffServ based quality of service over 802.11. Globecom 2003, San Francisco (2003)
11. Le Grand, G., Meraihi, R., Tohmé, S., Riguidel, M.: Intelligent Ad Hoc Networking to Support Real Time Services. VTC, Vehicular Technology Conference (2003)
12. Wolfinger, B., Wolf, J., Le Grand, G.: Improving Node Behavior in a QoS Control Environment for Local Broadcast Networks. Proceedings of SPECTS 2003, Montreal **35** (2003) 361–371
13. Chiu, D.M., Jain, R.: Analysis of the Increase and Decrease Algorithm for Congestion Avoidance in Computer Networks. Computer Networks and ISDN Systems **17** (1989) 1–14

17 GHz Wireless LAN: Performance Analysis of ARQ Based Error Control Schemes

Giuseppe Razzano[1,2], Luca Cecconi[1], and Roberto Cusani[1]

[1] INFOCOM Dpt., University of Rome "La Sapienza", Italy
[2] Telecommunication Research Center Vienna (ftw.), Austria
{razzano, cusani}@infocom.uniroma1.it

Abstract. The paper presents the development of ARQ schemes applied to a 17 GHz single-hop ad hoc network, providing very high bit rate. In particular, the paper aims to analyze and compare the performances of different ARQ protocols, in order to outline the most suitable one for a very high bit rate wireless system. Simulation trials show that a reliable error control protocol, realized by means of a proper retransmission strategy, can sensibly reduce the number of transmission errors, thus improving the wireless network overall performances.

1 Introduction

It is well known that wireless communications are exposed to high probability transmission errors. This is even more challenging when the aim is to develop a wireless LAN working at very high frequency (17 GHz) and with tight QoS requirements, in terms of error probability and transmission delay.

In this work we consider a novel wireless LAN developed in the framework of the WIND-FLEX (WF) project [1], funded by the European Information Society Technology (IST) program. We investigate in particular the performance of error control (EC) handling protocols at data link layer, to deal with the noisy radio channel. The EC protocol uses an Automatic Repeat reQuest (ARQ) that works on a per connection basis.

ARQ scheme is just an aspect of a wide strategy to improve the performance of the error-prone air interface. At physical layer, the system employs Forward Error Correction (FEC) scheme that improves the receiver capacity to detect and correct garbled bits [2]. In this paper we deal with Protocol Data Units (PDUs) at the data link layer, after they have been processed and, when possible, corrected by means of the the FEC scheme.

For what concerns the upper layers, the interaction between TCP and data link layer ARQ over wireless links has been extensively studied in many papers in the past years, especially considering ARQ persistence consequences on TCP behavior. Different conclusions are drawn about this subject. According to some works (e.g. [3]) not-fully persistent ARQ strategies should be employed at the data link layer, while other authors claim that a completely reliable ARQ scheme improves TCP performance (e.g. [4], [5]). In this work, we do not analyze the

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 92–99, 2005.

interaction between TCP and the data link layer ARQ. We adopt a non-fully persistent ARQ scheme, adapting the maximum number of retransmissions to the QoS requirements (especially in terms of maximum tolerable delay [6]) of the applications. As showed in the simulation results, this approach leaves to the TCP layer a small residual packet loss percentage, when, after some retransmission, the packet cannot still be correctly delivered within a certain delay threshold.

The rest of the paper is organized as follows: Section 2 presents the ARQ protocols, while Section 3 describes the wireless channel model. Simulation results are reported in Section 4. Finally some conclusions are drawn in Section 5.

2 ARQ Protocols

As already mentioned, the analysis of the paper focuses on three ARQ protocols, namely Stop-and-Wait (SW), Go-Back-N (GBN), and Selective-Repeat (SR). The following section provides a short description of the algorithms, pointing out the way they have been adapted to the WF system.

2.1 Stop and Wait (SW)

SW is the simplest ARQ scheme: the sender waits for a positive acknowledgment (ACK), or a negative one (NACK) from the receiver, after every packet transmission. Depending on the reply, the sender either transmits the next packet (ACK reception), or retransmits the last one (NACK reception). If the packet is lost or the reply is lost or corrupted, when a timer expires, the sender retransmits the not acknowledged packet. Using SW scheme no buffering for transmitted packets is required (both transmission and reception sliding windows are unitary).

The simplicity makes the SW algorithm attractive in many situations, but given its scarce utilization of the available bandwidth, the algorithm is not suitable for a network characterized by very high bit rate and thus expected to support high traffic loads. As simulation results confirm (see Section 4), the network performances are definitely scarce, when either traffic load or Packet Error Rate (PER) increase.

2.2 Go Back N (GBN)

When GBN protocol is used, the transmitter sends several packets consecutively. Transmitted packets are stored in a retransmission buffer, until they are positively acknowledged (individually or cumulatively). The number of packets, sent consecutively without acknowledgment, cannot exceed a transmission window, whose length is one of the algorithm parameters and mainly depends on the available HW resources. The receiver instead does not use any buffer, packets are acknowledged as soon as they are received.

If a gap in the reception sequence is detected, meaning that a packet is lost, the receiver suspends accepting packets and sends a NACK to request a retransmission of the missing one. The sender receiving a NACK restarts transmission from the missing packet and proceeds in sequence. The N packets that were transmitted after the corrupted (or missing) one are discarded even if they reached the receiver without errors.

Contrary to SW algorithm, GBN does not force the transmitter to remain inactive waiting for ACK/NACK messages after every packet transmission, therefore it is able to achieve a greater efficiency. Nevertheless, whenever a packet is lost or gets damaged, up to N redundant retransmissions are made, which results in a considerable resource waste.

2.3 Selective Repeat

When SR schemes are employed only corrupted or lost packets are retransmitted. In particular the algorithm considered for the WF system is *Selective-Repeat-with-Partial-Bitmap* (*SRPB*), which makes use of an optimized bit mask field in the acknowledgment messages, in order to reduce the amount of overhead [7]. The protocol works as follows. The transmitter sends a series of PDUs, sequentially numbered within the predefined window, making full use of the available bandwidth. Transmitted PDUs are stored until they are not acknowledged; for each PDU a timer is set. Also at the receiver side, a window is used to buffer correctly received packets. For each packet of the window, the status of the packet (positively received or not) is stored.

The acknowledgment is done sending a packet, whose format is reported in Fig. 1. The TYPE field is used to distinguish among the three kinds of packets used: data packets, acknowledgment packets, and data plus acknowledgment packets (which allow the transmission of ACK message in a piggybacking fashion). The CONNECTION_ID field carries the connection number to which the ACK message is related, i.e. the connection used to send data packets that are being acknowledged. The Flow Control (FC) field is set to 1 if the receiver window is full, in order to suspend the transmission of other data packets, until a new communication (FC bit set to 0) is received by the transmitter. We employed transmission and reception sliding windows, whose length is W$_s$=512, thus the sequence numbers supported are at least twice the buffer size, representing numbers from 0 to 1023. Data packet acknowledgment is done considering

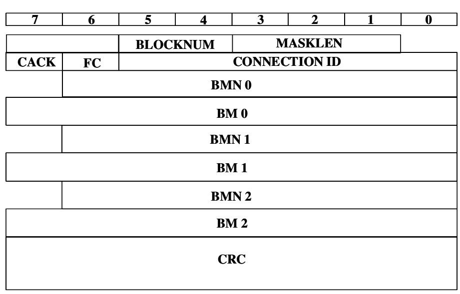


Fig. 1. SRPB acknowledgement packet format

blocks of packets and then specifying packets (PDUs), belonging to the block, which have or have not been received correctly. In particular, as reported in Fig. 1, we decided to use ACK messages containing up to 3 bitmap blocks, where each block (BM), identified by a 7-bit bitmap block number (BMN), is an 8-bit bitmask. Every packet belonging to a block is acknowledged with a bit set to 1 (correct packet) or 0 (corrupted or missing packet). For every ACK, the first 0 in the bitmask (corresponding to the first corrupted packet of the acknowledged blocks) suspends the progress of both the receiver and transmitter sliding windows, forcing the sender to retransmit the requested packet.

To construct blocks for the ACK message, groups of 8 packets to be acknowledged are considered. In the BLOCKNUM field it is reported the current number of bitmap blocks (BMs). When the last BM is used to acknowledge less than 8 packets, the MASKLEN field indicates the effective length of the block, thus allowing to dynamically allocate the necessary bitmap blocks, in order to reduce protocol overhead. The Cumulative ACKnowledgment (CACK) field is used to grant multiple packets, whose sequence number is lower than the one of the first packet in the first bitmap block. Finally the CRC field is used to detect transmission errors. Obviously, in presence of feedback errors, such as the lost of an ACK message, all packets related to the ACK blocks are retransmitted, after the expiration of the timers associated to the packets at transmitter side.

3 Channel Model

For modelling the error characteristics of a wireless channel between two stations, is widely used model the "Gilbert-Elliot" model: a two state Markov chain, where the states (*Good* and *Bad*) represent the possible behavior of the radio link. According to [8], assuming a flat fading channel and high data rates, such that the duration of a data packet (τ) is smaller than the coherence time of the channel (f_D), it is possible to consider as analytical channel model, a Gaussian random process with a given mean and the following covariance function:

$$K(\tau) = J_0(2 \cdot f_D \cdot \tau) \tag{1}$$

The covariance properties depend only on $f_D \cdot |\tau|$. When this quantity is small (i.e. < 0.1) the process is very correlated ("slow" fading). On the contrary, for larger values (i.e. > 0.2), two samples of the channel are almost independent ("fast" fading). Note that, for high data rate (small τ), the fading process can always be considered to be slowly varying. Therefore, the dependence between the transmission of consecutive data packets cannot be neglected, and the model for the success/failure process has to take into account this dependence. A general success/failure process model considers samples of the fading process:

$$\underline{\alpha}_n = (\alpha_1, \alpha_2, ..., \alpha_n), \qquad \alpha_i = \alpha(iT) \tag{2}$$

From a communication point of view, dealing with data link protocols, the aim is to evaluate the binary random process that describes the successes and the failures of the packet transmissions: $\beta(t) = \phi(\alpha(t))$ (assumed, for simplicity, memoryless). The success or failure of a packet is determined by comparing the signal power to a certain threshold (i.e. the case of power under this threshold stands for a packet failure). For highly correlated fading, it is possible to extend the (approximate) Markovian character of the fading process to the success/failure process. What is now left to verify, adopting a first-order Markov model, is that the success/failure of the transmission in the previous slot summarizes almost all the information contained in the past. The verification is based on [9] and [8], where is considered the average mutual information between the success/failure process β_i and the past two transmissions β_{i-1}, β_{i-2}. A measure of the goodness of the first-order Markov approximation can be given in terms of the negligibility of the additional information on β_i carried by β_{i-2} when β_{i-1} is known. For slow fading, this can be demonstrated, validating the first-order Markov approximation to be adequate for packet success/failure process on a fading mobile radio channel.

Given the coherence time $\Delta\mathrm{T}_c$ (1 ms) and the high data rates (up to 160 Mb/s) of the WF wireless network, it is possible to confirm the adequacy of the Gilbert-Elliot channel model in representing the considered radio channel. In fact, the bit rate and packet lengths, used in simulation trials (see Section 4), lead to a packet duration in the range [3.33μs, 50μs], which is surely smaller than the coherence time of the channel (1 ms). Moreover, given the coherence time, according to [10], Doppler frequency can be expressed as:

$$f_D = \frac{9}{16\pi \Delta t_C} \quad \Rightarrow \quad f_D \approx 179Hz \tag{3}$$

Therefore the product of the Doppler frequency and the packet duration approximately belongs to the following range:

$$f_D \cdot \tau \in \left[5.96 \cdot 10^{-4}, 8.95 \cdot 10^{-3}\right] \tag{4}$$

Being $\tau < \Delta\mathrm{T}_c$ and $f_D|\tau| << 0.1$, it is possible to conclude that the Gilbert-Elliot channel is adequate to characterize the behavior of the considered wireless channel.

4 Performance Analysis

The WF network has been simulated via software, using OPNET *Modeler 9.0*, and several trials have been carried out with different network scenarios and traffic conditions. The presented results refer to a scenario with five (fixed) devices in a 20x20 single-hop cluster. Three classes of service have been considered as representative of WLAN applications: *Streaming class* (used for audio and video streaming applications), *Interactive class* (used for web browsing applications), *Background class* (used for e-mail or FTP applications). The ARQ protocols are applied only on *Interactive* and *Background classes*. ARQ algorithms are not applied to *Streaming* class, whose low delay requirements make ineffective

Table 1. Traffic sources Parameters

Class of Service	QoS Requirements	Simulation Parameters		
	Maximum Delay	ON State	OFF State	Interarrival Time
Background	0.33 s	Exp(2s)	Exp(0.2s)	Exp(0.0015s, 0.0075s)
Interactive	0.125 s	Exp(3s)	Exp(0.1s)	Exp(0.0015s, 0.0075s)
Streaming	0.00275 s	Exp(2s)	Exp(0.1s)	Exp(0.0015s, 0.0075s)

Table 2. Channel parameters

Good State BER	$e_G = 0$
Bad State BER	$e_B = 10^{-4} : 10^{-3}$
Transition State Probability Good-Bad	$P_{GB} = 0.005$
Transition State Probability Bad-Good	$P_{GB} = 0.04$

a retransmission strategy for such applications. Table 1 describes the main parameters of the simulated traffic sources, and the maximum acceptable delay for the three classes of service. As already explained, we assume that the forward channel is a random-error channel, represented as a Gilbert-Elliot model. The channel status is defined by the BERs of the two states (*god* and *bad*) and the transition probability matrix, set as reported in Table 2. The feedback channel is assumed to be an ideal error free link. The considered values for e_B, together with the adopted packet size (450 bits), lead to a PER range of 1-10%. Obviously, the PER depends on the relationship between the channel error process and the packet size (longer packets are more likely to be hit by an error). Assuming that the CRC code error detection probability is ideal, a packet is considered garbled when at least one bit is hit by error.

Dealing with a very high bit rate WLAN, our interest was mainly focused on finding the most suitable protocol, to such a system. Out of the parameter considered to evaluate the respect of the QoS requirements, one of the most significant is definitely the maximum delay experienced by the packets, belonging to the three classes of services. As shown in Fig. 2-left, with respect to Background and Interactive classes, SW has a very high percentage (from about 75% to about 90%) of packets discarded due to high delay, mainly caused by the way retransmissions are handled by the protocol. This percentage increases with the traffic load and with PER (although the dependence with PER is weaker). The statistic values are not reported for GBN and SRPB scheme, because the percentage of packets not satisfying the QoS requirements is very low for both the protocols (less than 0.9% for GBN and less than 0.25% for SRPB), for every traffic situation and packet length. The opportunity to retransmit a garbled packet depends on its time to live (TTL), that is the time left which still enables the QoS requirements satisfaction (see Table 1). If a packet has not been received correctly, and its TTL is elapsed, the packet is discarded by the transmitter. The statistic of discarded packets, that cannot be “corrected” by means of the ARQ error control scheme, is shown in Fig. 2-right. As expected, SW is the worst

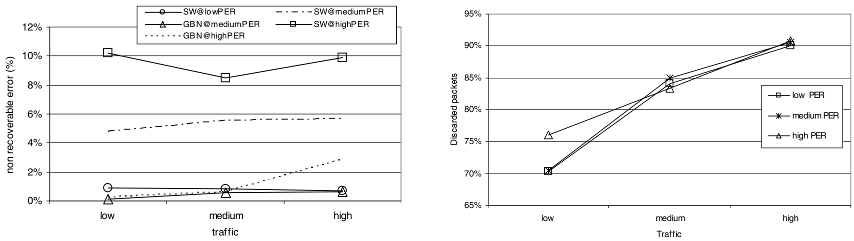


Fig. 2. *left*: Unrecoverable errors - *right*: Discarded packets (SW algorithm)

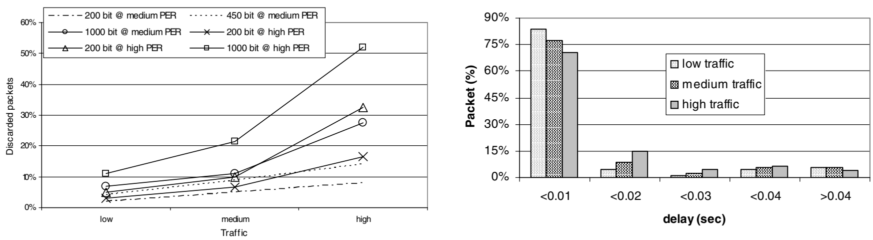


Fig. 3. *left*: Discarded packets (GBN algorithm) -*right*: Resequencing delay (SRPB algorithm)

protocol in recovering the garbled packets. GBN has a very low percentage of unrecoverable packets (from 0% to 3%), while the best results are obtained using SRPB (not reported in the picture), with a not-recovery percentage always lower then 0.002%.

Concerning GBN, it is interesting to show how many packets correctly received, are discarded due to protocol mechanism, which is based on an unitary receiver window. Fig. 3-left reports such percentage. This number grows, with the increasing of traffic load or of PER, reaching very high values and leading to a considerable resource wasting.

When analyzing the performances of SRPB protocol, it is important to evaluate the resequencing delay, which is the time that correctly-received packets spend in the receiver buffer, waiting for corrupted or missing packets to be received correctly. The resequencing delay has to be considered for SRPB protocol, which is the only protocol allowing the reception of out of sequence packets. As can be seen from Fig. 3-right, the receiver buffering time is, in great percentage, concentrated below 10 ms, with a reduction of this percentage when traffic increases, and a very little growth of the percentages related to higher waiting intervals. The resequencing delay is the acceptable price to pay, for SRPB, which avoids retransmission of correctly received out-of-order packets.

5 Conclusions

In this paper we investigated the performances of an EC protocol at DLL, for a very high speed wireless LAN. The EC protocol has been implemented choosing three ARQ schemes: Stop and Wait, Go Back-N and Selective Repeat with Partial Bitmap. We analyzed the performances under different traffic loads, different packet lengths and error rates. On one hand, SW and GBN have outlined great inefficiencies: with regard to the respect of QoS parameters, unrecoverable errors and bandwidth utilization (mainly SW) and overhead, energy waste and bandwidth waste due to useless retransmissions (mainly GBN). On the other hand, SRPB ensured: high efficiency, low overhead, high QoS parameter respect and very low percentage of unrecoverable errors. In particular, the overcoming of Selective Repeat ARQ schemes on the other two protocols, in such a network, comes from considering its Partial Bitmap version. The innovative acknowledging mode, presented in the paper, enables to grant blocks of packets and to dynamically allocate the size of the ACK packet, thus enabling to obtain all the above listed advantages at a reasonable increase of the computational cost.

References

1. Polydoros A. et alii “WIND-FLEX: Developing a Novel Test-Bed for Exploring Flexible Radio Concepts in an Indoor Environment” Communications Magazine,IEEE , Volume: 41 Issue:7, July 2003 pp 116-122
2. Saarinen I. et alii “High Bit Rate Adaptive WIND-FLEX Modem Architecture for Wireless Ad-Hoc Networking in Indoor Environments”, proceedings of IST Mobile Communications Summit 2002 pp 649-654
3. Balakrishnan H. et alii “Improving reliable transport and handoff performance in cellular wireless networks” Wireless Networks, December 1995
4. Chaskar H.M. et alii “TCP over wireless with Link Level Error Control: Analysis and Design methodology”, IEEE/ACM Trans on Networking, Vol.7, Oct 1999
5. Vacirca F. et alii “Optimal Design of Hybrid FEC/ARQ Schemes for TCP over Wireless Links with Rayleigh Fading”, to appear on IEEE Trans. on Mobile Comp.
6. Razzano G. et alii “New Generation Wireless LAN: Hardware Description and Measurements of 17 GHz RF board, Performance Analysis of DLC and MAC layers”, proceedings of IST Mobile Communications Summit 2002 pp 202-206
7. Li H. et alii “Automatic repeat request (ARQ) mechanism in HIPERLAN/2” Vehicular Technology Conference Proceedings, 2000. VTC 2000-Spring Tokyo. 2000 IEEE 51st ,Vol.: 3 ,May 2000 pp 2093-2097 vol.3
8. Zorzi M. et alii “On the Accuracy of the first-order Markov model for data transmissions on fading channel” Proceedings of IEEE ICUPC, pp. 211-215, 1995
9. H.S. Wang et alii “Finite state Markov channel - A useful model for radio communication channels”, IEEE Trans. on Vehicular Tech., 44 pp. 163-171, Feb. 1995
10. R. Van Nee and Prasad R. “OFDM for wireless multimedia communications”, Artech House 2000

Performance Analysis of MAC-hs Protocol

Robert Bestak

Czech Technical University in Prague, Department of Telecommunications Engineering,
Technicka 2, 16627 Prague 6, Czech Republic
bestar1@fel.cvut.cz

Abstract. Two main features of the MAC-hs protocol of HSDPA are retransmissions of erroneous blocks and in-sequence data delivery to the upper layer. The first function is fulfilled through the HARQ mechanism. The second function is achieved by managing transmitting/receiving window and by using a specific numbering. In this paper, the MAC-hs performance for different window sizes, timer's values and number of retransmission attempts are studied. Simulations show that values of these parameters have to be carefully set up in order to prevent incorrect block discards at the receiver side.

1 Introduction

The HSDPA concept (High Speed Downlink Packet Access, e.g., [1], [2]) of UMTS has been introduced in Release 5 of 3GPP. The HSDPA includes several enhanced techniques such as fast link adaptation, HARQ (Hybrid ARQ) or higher order modulation (16-QAM) that increase downlink data rates up to 10 Mbit/s on the air interface.

The HSDPA introduces a new transport channel and three physical channels (see fig. 1). The transport channel, called High Speed Downlink Shared Channel (HS-DoShCH), is shared among several users. The HSDPA scheduler reallocates radio resources (i.e. channelization codes) with a period called HS-DoShCH TTI (Transmission Time Interval). For the FDD mode, the HS-DoShCH TTI is specified to be 2 ms ([2]). In the rest of paper, the period is simply called TTI since there is no confusion with the conventional duration of TTI in UMTS, which can be 10, 20, 40, or 80 ms. Within a TTI, radio resources can be allocated to one or several users.

At the physical level, data of HS-DoShCH (i.e. MAC-hs PDUs) are mapped into the frame structure of HS-Physical Downlink Shared Channel (HS-PDoShCH). Three consecutive slots in the HS-PDoShCH frame form a radio "unit" for traffic. We denote this three slot unit as T-slot. The T-slot duration corresponds with the TTI duration. One HS-PDoShCH corresponds to one channelization code (with a fixed spreading factor SF = 16, [3]). There can be employed up to 15 channelization codes ([4]), i.e. up to 15 HS-PDoShCHs can be assigned in a T-slot.

HSDPA signalling information (downlink/uplink) is convey via control channels. The downlink signalling informs a mobile how to decode transmitted data on the HS-PDoShCHs (type of modulation and coding, transport format, HARQ information). The signalling is carried by a downlink HS-Shared Control Channel (HS-ShCoCH). The transmission of HS-ShCoCH precedes HS-PDoShCH by 1,33 ms (or 2 slots, [1]).

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 100 – 108, 2005.

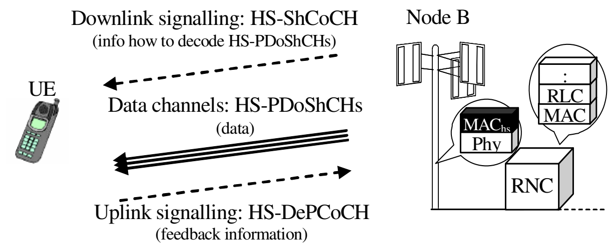


Fig. 1. HSDPA physical channels: HS-ShCHCH (High Speed-Physical Downlink Shared Ch.), HS-ShCoCH (HS-Shared Control Ch.), HS-DePCoCH (HS-Dedicated Physical Control Ch.)

The uplink signalling (HARQ Ack/NAck and Channel Quality Indication, or CQI) is carried by HS-Dedicated Physical Control Channel (HS-DePCoCH).

High flexibility of the HSDPA allocation mode is reached by reducing the basic allocation period from 10 ms to 2 ms. The assignment of radio resources (scheduling) and HARQ functions are implemented in a new MAC-hs entity/layer (Medium Access Control - high speed, [5]). The MAC-hs is located in the Node B. There is one MAC-hs entity per UE (User Equipment). The MAC-hs layer can be seen as a layer composing of two sub-layers: upper one and lower one.

The lower MAC-hs sub-layer handles (re)transmissions of blocks between the Node B and UE. The HSDPA uses HARQ mechanism that is based on the ARQ method Stop and Wait. Up to 8 independent HARQ processes (or instances) per UE can simultaneously be active ([5]), i.e. up to 8 MAC-hs PDUs of a UE can be handled at the same time. We shortly denote MAC-hs PDUs as d-Blocks in the rest. At most one HARQ process of UE can be activated in a T-slot. A comparison study of different HARQ schemes (HARQ I, HARQ II, etc.) can be found for example in [6], or [7].

The upper MAC-hs sub-layer manages flow control, reassembling/segmentation, numeration, and in-sequence data delivery to the upper layer. The MAC-hs in-sequence function is fulfilled via transmitting/receiving window and by using a specific numbering. This paper focuses on parameter settings that are tied with the in-sequence function. We investigate the MAC-hs performance for different values of transmitting/receiving window size and reordering release window timer. Further, we look how the performance change for different number of retransmission attempts.

The rest of the paper is organized as follows. The next section describes the MAC-hs protocol with the focus on the in-sequence data delivery function. The simulation model is presented in section III. Section IV describes simulation scenarios and results. The last section presents our conclusions.

2 Basic Features of MAC-hs Protocol

A UMTS user can activate several radio bearers with different priorities. To reflect this UMTS feature, up to 8 priority queues per MAC-hs entity can be activated. The pot of 8 HARQ processes is common for all live queues of the given UE. Fig. 2 shows an example where two priority queues (Q1 and Q2) are active per user (UE1).

Each time a new d-Block is scheduled to be sent, the MAC-hs scheduler determines: UE, the UE's priority queue and a suitable d-Block payload size. A d-Block payload consists of one or several RLC blocks. The selection of suitable d-Block payload size is important since the d-Block size cannot be modified during retransmission attempt(s). A proposal dealing with this issue can be found in [8].

A sending d-Block is assigned a Transmission Sequence Number (TSN, modulo 64), Queue Id (3 bits) and one of the free HARQ Ids (3 bits). The TSN and QId are carried in the d-Block header, whereas HARQId is carried by the physical downlink control channel. Each priority queue manages numbering of d-Blocks (i.e. TSN) independently to other priority queues.

Fig. 2 shows a case where the MAC scheduler selects UE = 1, a priority queue Q1 of UE1 and the d-Block is assigned to HARQ process with HARQId = 2. Since two priority queues are active, two MAC-hs windows are managed.

The assigned HARQ process controls (re)transmissions of d-Blocks. If a retransmission occurs, the originally selected d-Block size is kept constant and different MCSs (Modulation Coding Scheme) may be employed. Different MCSs lead to different coded block sizes and thus different number of channelisation codes is needed. We denote a d-Block on which is applied on it channel coding (ECC) as c-Block.

Via the TSN, the MAC-hs layer provides in-sequence data delivery to the upper. The sorting of received d-Blocks is realized in the MAC-hs reordering entity (fig. 3). The MAC-hs scheduler may only send d-Blocks with TSN that lie within the MAC-hs transmitter window. The maximum transmitter/receiver window size is 32 ([9]).

Notice that due to the multi-instance ARQ feature together with the specific numbering of d-Blocks, the MAC-hs retransmission mechanism behaves as if the ARQ scheme Selective Repeat would be used.

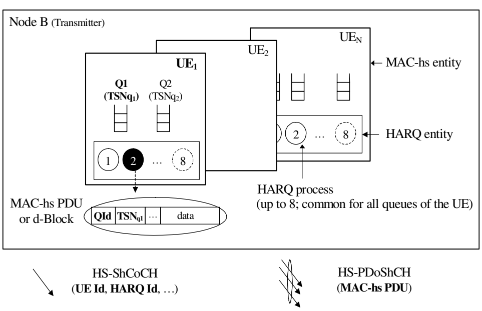


Fig. 2. The MAC-hs entities in the Node B and distribution of information field in the downlink physical channel; the fig. shows an example when the MAC scheduler selects: UE = 1, Id = 1, HARQId = 2

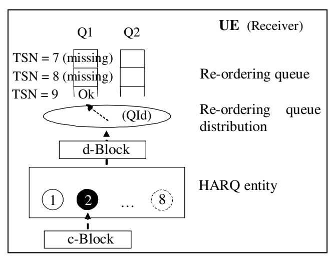


Fig. 3. The MAC-hs entity in a UE; as in fig. 2, there is shown an example when UE = 1, QId = 1, HARQId = 2

The transmitting MAC-hs entity is configured by the upper layer to discard data from its buffer that is out of date. There is no explicit signaling between the Node and UE when discarding data. The receiving MAC-hs entity (in the UE) is informed about a discard implicitly: either (i) by expiration of the re-ordering release timer or (ii) by reception of a fresh d-Block above the upper edge of receiving window. We denote the first type of discard as timer discard and the second one as window discard.

The re-ordering release timer is called, in 3GPP specifications, as timer T1. The timer T1 controls stall avoidance events in the UE reordering buffer. There is one timer per receiving priority queue. The T1 is initialized for a d-Block that cannot be delivered to the upper layer due to previous missing d-Block(s) in the reordering buffer (e.g.; d-Block with TSN = x, d-Block$_x$). In fig. 3, the T1 is started for the d-Block$_9$. The T1 is stopped as soon as the d-Block$_x$ can be delivered to the upper layer.

If the T1 expires, the MAC-hs receiver window is advanced in such a way that: a) all correctly received d-Blocks up to d-Block$_x$ (including) are delivered to the upper layer, and b) all following correctly in-sequence received d-Blocks above d-Block$_x$ are also delivered to the upper layer. If there is still a d-Block in the reordering entity that can not be delivered to the upper layer, the T1 is restarted for the first non-deliverable d-Block in the window.

The receiving window is also advanced and data discarded whenever a fresh d-Block above the upper window edge is received (i.e. the window discard occurs). The received d-Block forms the new upper edge of receving window. After the window update, the T1 is started for the first non-deliverable d-Block; as in the above describe timer discard procedure.

If necessary, discarded data at the MAC-hs level is retransmitted through ARQ mechanisms of the upper layer, e.g., RLC (Radio Link Control) or TCP (Transmission Control Protocol). Neither RLC nor TCP is considered in our simulation model.

3 Model of Simulation

The simulation model and layer architecture are illustrated in figure 4.

A fixed number of UEs (= 10) in the cell is considered. All UEs have the same capabilities parameters: (i) maximum number of channelization codes per UE = 6 and (ii) minimum inter-TTI = 2 ms. The minimum inter-TTI specifies the minimum period between the beginning of a TTI and the beginning of the next used TTI that can be supported by UE ([4]).

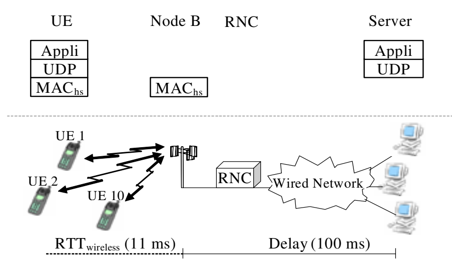


Fig. 4. Model of simulation and layer architecture

MAC-hs Scheduler. The MAC-hs scheduler implements Round Robin algorithm. The maximum HSDPA channelization codes that the scheduler can assign in a T-slot is 12. The scheduler uses in a T-slot all available channelization codes, if possible.

MCS and Simulation of Radio Conditions. There are considered 9 types of MCSs and 6 sizes of d-Blocks (see table 1).

Table 1. Size of d-Blocks and MCSs

	Size of MAC-hs PDU (data rates), MCSs					
Channel. codes	480 bits (240kb/s)	720 bits (360kb/s)	960 bits (480kb/s)	1440 bits (720kb/s)	1920 bits (960kb/s)	2880 bits (1,44Mb/s)
2	QPSK ¼ (MCS4)	QPSK 1/3 (MCS5)	QPSK ½ (MCS6)	QPSK 3/4 (MCS7)	16QAM1/2 (MCS8)	16QAM 3/4 (MCS9)
4	QPSK 1/8 (MCS2)	QPSK 0,18 (MCS3)	QPSK 1/4 (MCS4)	QPSK 1/3 (MCS5)	QPSK 1/2 (MCS6)	QPSK 3/4 (MCS7)
6	QPSK 0,08 (MCS1)	QPSK 1/8 (MCS2)	QPSK 1/8 (MCS2)	QPSK 1/4 (M\CS4)	QPSK 1/3 (MCS5)	QPSK 1/2 (MCS6)

Variation of the radio channel is simulated through a variable SIR (Signal to Interference Ratio). The SIR is variable following a normal distribution $N(\mu, \delta^2)$; the mean $\mu = 0$ and the standard deviation $\delta = 4$ dB. The memory of the random process indicates a parameter Tv, Tv $\in$ (15 ms; 20 ms; 100 ms, 400 ms; 1,5 s).

HARQ Processing. For a scheduled UE, a MCS is selected in such way that SIR(MCS) < SIR_{NodeB}, where SIR_{NodeB} is the last known value of SIR (for the given UE) in the Node B. The SIR(MCS) thresholds are given in the table 2 ([10]).

Table 2. SIR thresholds for different MCSs

	MCS1	MCS2	MCS3	MCS4	MCS5	MCS6	MCS7	MCS8	MCS9
SIR [dB]	-12	-7	-5	-4	-1	1	3	5	9

Due to feedback delay (propagation, data processing in the Node B and UE) and scheduling, there is a delay between the last indicated value of SIR by UE and the moment of selecting MCS. The min. delay is assumed to be 6 ms (3 T-slots) and the max. delay 20 ms (10 T-slots). After 20 ms,the value of SIR in the Node B is updated.

When transmitting a new d-Block, the selected MCS can correspond to several d-Block sizes (see table 1). To enlarge the set of possible MCSs that can be employed for retransmissions, the lowest d-Block size is chosen; channel conditions are expected to get worse rather than to ameliorate. The selected MCS and d-Block size determine a number of channelization codes that need to be used. Retransmissions are performed by selecting a MCS in the column of the corresponding d-Block size.

A HARQ instance in UE processes c-Blocks according to the following procedure:

if SIR(MCS) < SIR_{UE} than erroneous c-Block
else correctly decoded c-Block

where SIR_{UE} is the latest value of SIR calculated in the UE. The maximum retransmission attempts per c-Block are delimited by a parameter denoted MaxDat in our paper. The downlink and uplink signalling is assumed to be error free.

Wireless and Wired Delay. A wireless logical Round Trip Time ($RTT_{wireless}$), i. e. the time between the transmission of the first control bit on the HS-ShCoCH and the reception of the last bit of the corresponding Ack/NAck on the HS-DePCoCH, is set to 16,5 slots (or 11 ms). The wired delay between the server and Node B is 100 ms.

Traffic Model. Users run web-browsing above UDP (User Data Protocol). A web-browsing session is comprised of several packet calls. A packet call is followed by a reading time interval to view the download contents. At the end of reading time interval, the UE downloads another web page and so on. The packet size is modeled by Pareto random variable with cutoff (α = 1,6; min = 1,8 kB; max = 40 kB; mean = 4,4 kB [11]). The reading time interval is an exponential random variable (mean = 5s).

4 Simulation Results

Simulation experiments are carried out for two MAC-hs transmitting/receiving window sizes (4 and 16) and for two values of MaxDat (2 and 4).

In the Node B, a d-Block is discarded if a retransmission counter associated to every c-Block reaches a value of the parameter MaxDat (MaxDat discard). In a UE, d-Blocks are discarded either due to the timer discard or due to the window discard.

Figure 5 and 6 show a ratio of discarded d-Blocks in the Node B versus discarded d-Blocks in UEs; in fig. 5 (fig. 6) the MSC-hs window size is set to 4 (16):

$$\frac{\sum_{NodeB} \textit{discarded blocks due to the MaxDat discard}}{\sum_{UE} \textit{discarded blocks due to the timer discard} + \sum_{UE} \textit{discarded blocks due to the window discard}} \tag{1}$$

From fig. 5 we can observe that for small T1 values (20, 50, 100 ms) a UE discards more d-Blocks than the Node B does. A missing d-Block in a UE is discarded before the erroneous c-Block can be corrected through the MAC-hs retransmission mechanism or MaxDat discard occurs in the Node B. For higher T1 values (400, 500 ms), the number of discarded d-Blocks in the Node B and UEs is same. The retransmission mechanism has enough time to correct erroneous c-Blocks or activate MaxDat discard before the UE timer discard takes place. As the variation of channel conditions gets slower (Tv values increase), the ratio gets smaller for the T1 = 20 ms. For higher T1 values, the ratio increases T1 $\in$ (50, 100 ms).

Larger MAC-hs window size (see fig. 6) has a little impact on the ratio of discarded d-Blocks in the Node B versus UEs. Both graphs are about the same.

Let's now investigate which of the UE's discard mechanisms dominate: timer discard or window discard. The ratio of discarded d-Blocks due to the window discard versus all discarded d-Blocks in UEs is shown in fig. 7 and fig. 8:

$$\frac{\sum_{UE} \textit{discarded blocks due to the window discard}}{\sum_{UE} \textit{discarded blocks due to the timer discard} + \sum_{UE} \textit{discarded blocks due to the window discard}} \tag{2}$$

Fig. 7 shows that for T1 = 20 ms, the timer discard dominates no matter how fast change the channel conditions. Just a few d-Blocks are discarded via the window discard; the T1 values are so small that missing d-Blocks in UEs are discarded before the MAC-hs retransmissions of erroneous c-Blocks can go through or MaxDat discard occurs. For other T1 values, the window discard becomes more and more important as the variation of channel conditions gets slower (Tv values increase). The discarded d-Blocks in the Node B are detected in UEs by reception of d-Blocks above the upper edge of the MAC-hs receiving window.

When setting the MAC-hs window to larger size, more d-Blocks become process at the same time. The transmission time (including retransmissions) of d-Blocks increases and the timer discard becomes more dominant (fig. 8).

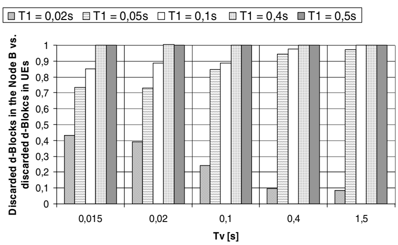


Fig. 5. Ratio of discarded d-Blocks in the Node B versus discarded d-Blokcs in UEs for various values of Tv and T1; **MaxDat = 2, window size = 4**

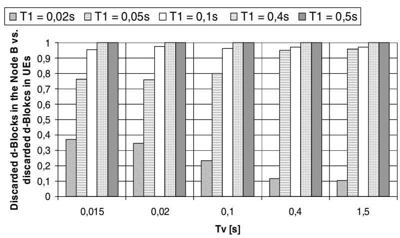


Fig. 6. Ratio of discarded d-Blocks in the Node B versus discarded d-Blocks in UEs for various values of Tv and T1; **MaxDat = 2, window size = 16**

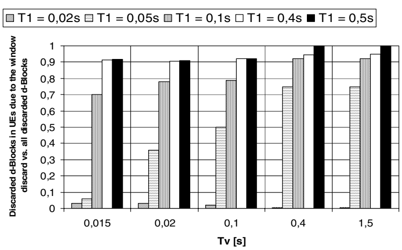


Fig. 7. Ratio of discarded d-Blocks in UEs due to the window discard versus all discarded d-Blocks for different values of Tv and T1; **MaxDat = 2, window size = 4**

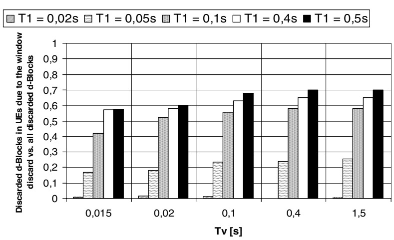


Fig. 8. Ratio of discarded d-Blocks in UEs due to the window discard versus all discarded d-Blocks for different values of Tv and T1; **MaxDat = 2, window size=16**

Figures 9-12 show simulation results of the second scenario where MaxDat = 4. By comparing fig. 5 (fig.6) and fig. 9 (fig. 10), we observe that there are more discarded d-Blocks in UEs for smaller T1 values (20, 50, 100 ms) than in the first scenario (MaxDat = 2). In this case, the MaxDat value and the T1 values are not proportional. The T1 values are too small compared to the value of MaxDat. The timer discard is dominant (fig. 11) and the UE discards d-Blocks before the Node B really discards d-blocks itself due to MaxDat.

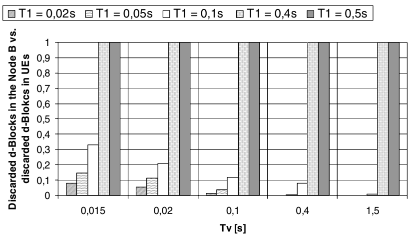


Fig. 9. Ratio of discarded d-Blocks in the Node B versus discarded d-Blokcs in UEs for various values of Tv and T1; **MaxDat = 4, window size = 4**

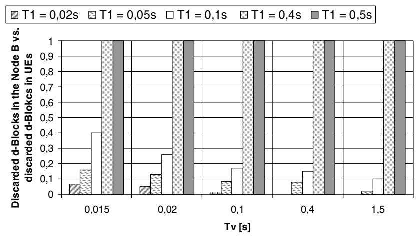


Fig. 10. Ratio of discarded d-Blocks in the Node B versus discarded d-Blocks in UEs for various values of Tv and T1; **MaxDat=4, window size = 16**

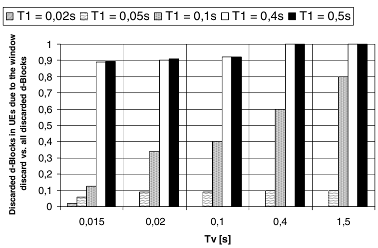


Fig. 11. Ratio of discarded d-Blocks in UEs due to the window discard versus all discarded d-Blocks for different values of Tv and T1; **MaxDat = 4, window size = 4**

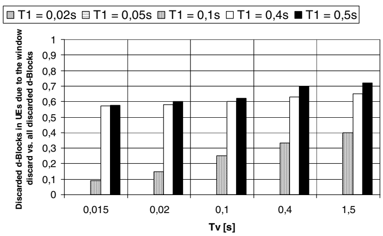


Fig. 12. Ratio of discarded d-Blocks in UEs due to the window discard versus all discarded d-Blocks for different values of Tv and T1; **MaxDat=4, window size = 16**

5 Conclusions

We have studied performance of the MAC-hs protocol for different window sizes, timer values and number of retransmissions. Simulations show that values of T1 and MaxDat have to be adequately set up. Setting up values of T1 too small, compared to values of MaxDat, results in more discarded d-Blocks in UEs than the Node B really discard. In such case, d-Blocks are discarded in the UE due to the timer discard. The MAC-hs window size has not impact on the ratio of discarded d-Blocks in the Node B versus discarded d-Blocks in the UE. However, a larger size of the MAC-hs window increases the number of discarded d-Blocks in UEs due to the timer discards.

References

1. Hedberg, T., S. Parkvall, S.: Evolving WCDMA. Ericsson review No.2. 2000.
2. TS 25.308: High Speed Downlink Packet Access (HSPDA), Overall Description, Stage 2 (Release 6), 3GPP. Mars, 2004.
3. TS 25.211: Physical channels and mapping of transport channels onto physical channels (FDD) (Release 5), 3GPP, September 2003.
4. TS 25.306: UE Radio Access capabilities (Release 5), 3GPP. September, 2003.
5. TS 25.321: MAC protocol specification (Release 6), 3GPP. Mars, 2004.

6. Das, A., Khan, F., Su, H.: Adaptive Asynchronous Incremental Redundancy (A2IR) with Fixed Transmission Time Interval (TTI) for HSDPA. PIMRC02. Lisbon, Portugal, 2002.
7. Frenger, P., Parkvall, S., Dahlman, E.: Performance Comparison of HARQ with Chase Combining and Incremental Redundancy for HSDPA. VTC Fall 01. Atlantic City, USA, October 2001.
8. Bestak, R., Godlewski, P., Martins, P.: HSDPA adaptation scheme of MAC-hs PDU size for retransmissions. CSN 2003. Benalmadena, Spain. September, 2003.
9. TS 25.331: RRC protocol specification (Release 6), 3GPP. Mars, 2004.
10. Parkvall, S., Peisa, J., Furuskär, A., Samuelsson, M., Persson, M.: Evolving WCDMA for Improved High Speed Mobile Internet. Proc. of the Future Telecommunications Conference 2001. Bejing, China. November, 2001.
11. Stacey, M., Nelson,J.,Griffin, I.: TCP for Transactions. Linux journal Is. 47. November, 99.

Distributed k-Clustering Algorithms for Random Wireless Multihop Networks

Vlady Ravelomanana

Université de Paris – Nord, LIPN – UMR 7030,
99, Av. Clément 93430 Villetaneuse, France

Abstract. Ad hoc networks consist of wireless hosts that communicate without the need of any fixed infrastructure. A k-clustering protocol is an algorithm in which the wireless network is divided into non-overlapping sub networks, referred to as clusters, and where every node of a sub network is at most k hops from a distinguished station called the clusterhead. Clustering is commonly used in ad hoc networks in order to limit the amount of routing information stored and maintained at individual nodes. In our setting, a large number n of distinguishable stations (e.g. sensors) are randomly deployed in a given area of size $|\mathcal{S}|$. We assume that the nodes use synchronous radio transmissions and any pair of nodes u and v are able to communicate if they are within a distance less than their transmitting range of each other. Moreover, if more than two neighbors of a node u transmit simultaneously, u is assumed to receive no message (collision). Under these assumptions, we propose and analyze efficient and fully distributed algorithms for the k-clustering problem.

1 Introduction

Advances in micro-electro-mechanical systems (MEMS) technology, wireless communications and digital electronics have enabled the development of multifunctional (tiny) nodes. Since a few, multihop ad hoc wireless networks gained in importance as subject of intense and attractive research [26]. For instance, sensor nodes are small miniaturized devices which consist of sensing, data processing and communicating components [1, 11, 26]. In this paper a network is a collection of transmitter-receiver devices, referred to as *stations* (*processors* or *nodes*). Multihop wireless networks consist in a group of stations that can communicate with each other over one wireless channel by messages (signals). Besides, messages may go through intermediate stations before reaching their final destination. The network is fully distributed: it comes without links and without centralized controller. At any given time t, the network may be modeled with its *reachability graph*: for any pair of stations u and v, there exists one directed edge (arc) $u \rightarrow v$ *iff* v can be reached from u. If the power of all transmitters/receivers is the same, the underlying reachability graph is *symmetric*, which is the case in our paper. Such networks are well suited to specific and often extremal situations, such as disaster-relief, law-enforcement, fire-detection, or simply for collaborative computation in some public short-term events (see for instance [2, 5, 26]).

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 109–116, 2005.

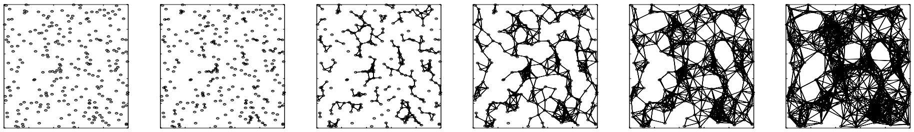


Fig. 1. Typical networks generated via uniform distribution. The transmission ranges increase from left to right and reaches the connectivity (on the right)

A global model for a mobile computing environment is a graph $G_t(V,\ E_t)$ where V is the set of stations and E_t is the set of links, which are present at time slot t. The problem under consideration consists in partitioning an ad hoc network into limited-diameter clusters (k-clustering). More specifically, given a graph $G_t(V,\ E_t)$ and a positive integer k, find the smallest value of m such that there is a partition of V into m disjoint subsets $V_1, \cdots, V_m$ and $diam(G\,[V_i]) \leq k$ for $i \in [1,\ m]$. Note that the algorithmic complexity of k-clustering has been shown to be NP-complete for simple undirected graphs [9, 12].

A commonly encountered model of network is defined by a pair n and $\mathcal{S}$ where n homogeneous nodes are randomly thrown in a given region $\mathcal{S}$ of surface $|\mathcal{S}|$, uniformly and independently. This typical modeling assumption is commonly used by many researchers [6, 7, 15, 16, 17, 20, 27, 28, 29, 30]. In particular, the initial placement of the nodes is assumed to be random when sensors nodes [1] are distributed over a region from a moving vehicle such as an airplane. This issue allows rapid deployment particularly in inaccessible terrains and, in this setting, the positions of the nodes need not be engineered or pre-determined.

As customary (e.g. [21, 23]), the time is assumed to be slotted and in each time slot (round) every node can act either as a *transmitter* or as *receiver*, but *not both*. In any given time slot, a station u acting as a receiver gets a message, if and only if, exactly one of its neighbors transmits within the same round. If more than two neighbors of u transmit simultaneously, u is assumed to receive no message (*collision*). That is, the considered networks has no ability to distinguish between the lack of message and the occurrence of some collisions or conflicts. This assumption is motivated by the fact that in many real-life situations, the stations are (small) devices and do not always have the ability for collision detection. Moreover, even when such detection mechanism is present, it may be of limited value, especially in the presence of noise. Therefore, it is highly desirable to design protocols working independently of the existence/absence of any collision detection mechanism.

In the context of mobile applications, the users of the network can move, and therefore the topology is unstable. For this reason, it is desirable for the network's protocols not to assume any knowledge on the network topology, or about structural information that stations may have regarding the topology. Therefore, we assume in this work that the stations have initially *no* topological information. Moreover, even the IDs (or IP addresses) of their respective neighbors are not known to the stations.

For sake of simplicity, we suppose that the network topology remains unchanged throughout the execution of the algorithms. This assumption is justified for sensor networks and is partly justified if the used protocols are sufficiently "fast" for "slowly" moving nodes.

Problem Statement and Requirements. Our aim is to design and analyze algorithms working on n randomly deployed nodes (such as those depicted in figure 1) in order to partition the underlying reachability graph G into subgraphs $H_1, \cdots, H_m$ where

- for each H_i, $1 \leq i \leq m$, there is a distinguished node called the clusterhead,
- any node u of the network belongs to a cluster H_i; u knows its clusterhead and is at most k hops from it,
- two distinct clusterheads are at distance at least $(k+1)$ hops from each other and
- a node "reachable" by a path of length less than k of at least two clusterheads is a *gateway* and should maintain a list of all clusters in its neighborhood.

Moreover, our k-clustering protocols should be `fully distributed` and have to take into account the `interferences` between the radio transmissions.

2 The Basic Case (k = 1)

We now introduce the 1-clustering algorithm. Our algorithm can be split into two steps:

- First, each station has to discover its proper neighborhood. This is done using the randomized algorithm ExchangeID. This protocol needs $O(\log(n)^2)$ steps.
- Next, once the station nodes know their neighborhood, we run BasicClustering which is a randomized (greedy) algorithm. This protocol builds non-overlapping clusters $H_1, \cdots, H_m$ of hop-diameter less than 2. In each cluster H_i, a specific node h_i is designed as *clusterhead* whereas the other nodes are 1 hop far from h_i.

In both cases, the protocols are fully distributed and they are executed independently and simultaneously by all the participating stations. Moreover, they take advantage of being simple. The first algorithm is necessary since as already stressed, our algorithm design is intended to wireless mobile networks. In such context, nodes are continuously moving and no station has to be aware of its neighbors identities permanently. Thus, ExchangeID can be invoked frequently and regularly for this purpose. Our results remain valid if the nodes are moving uniformly.[1]

[1] It is known that the main properties of the random Euclidean network are **invariant** if every node is translated independently and uniformly [18, 24].

2.1 Discovering the Neighborhood

Throughout this paper, we will often use a simple protocol which we shall call Send. Its aim is to allow a node u to send a given message to all of its 1-hop neighbors. The first parameter of Send is its duration, the second represents the message to be sent (a message '*msg*' is denoted $\prec$ msg $\succ$) :

Algorithm 0: Send(*duration*, *message*)

---

for $i := 1$ to *duration* **do**

With probability $\frac{1}{\log n}$, broadcast $\prec$ *message* $\succ$;

The protocol intended for neighborhood discovery uses Send. In all the following, the constant $C(\ell)$ is a parameter of the algorithm which will be clear from the context. Moreover (see [28]), $C(\ell)$ satisfies $C(\ell) \geq 2W_0(-\ell/e(1+\ell))$ where W_0 is the Lambert W function (see [8]). We have the following result related to the ExchangeID algorithm as well as the fundamental characteristics of randomly deployed networks:

Algorithm 1: ExchangeID

---

begin

Each node u sends a message containing its own identity :

Send$\big(C(\ell)\, \log{(n)}^2, \prec \mathrm{ID}(u) \succ\big)$;

end

Theorem 1. *For any fixed constant $\ell > 0$, there exists a constant $C(\ell)$ such that if the transmission radius of each station is set to $r = \sqrt{\frac{(1+\ell)\,|\mathcal{S}|\,\log n}{\pi n}}$ then with probability tending to 1 as n tends to ∞, after an execution of the protocol* ExchangeID*, every node has received all the identities of all its neighbors.*

Proof. See [28].

2.2 1-Clustering Protocol

For any participating node u, let us denote by Γ_u the set of its (known) neighbors. Recall that if the transmission range is set to $r = \sqrt{\frac{(1+\ell)\,|\mathcal{S}|\,\log n}{\pi n}}$, then with high probability $|\Gamma_u| = \Theta(\log n)$ (cf. [28]). The protocol BasicClustering given in the next page proceeds as follows. First, each node has to discover its neighborhood. Then, we can start the proper clustering algorithm. If a node u has the lowest-ID among its neighbors and itself, u has to try to "access the channel" in order to advert its neighbors. After that, u becomes the clusterhead known by its 1-hop neighbors. Observe that the running time of our algorithm is $2\,C(\ell)\,\log{(n)}^2 + O(1)$. A node which can receive messages from two or more clusterheads is a *gateway*.

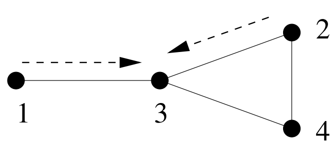
Algorithm 2: BASICCLUSTERING

begin
  Run EXCHANGEID;
  For each node u : (i) define $\Gamma_u :=$ {set of 1-hop neighbors},
  (ii) CLUSTERID(u) := UNKNOWN and (iii) GATEWAY(u) := $\emptyset$;
  Set $\Gamma_u := \Gamma_u \cup \{u\}$;
  if (ID(u) == min(Γ_u)) **then**
    for $i := 1$ *to* $C(\ell)\log(n)^2$ **do**
      With probability $\frac{1}{\log n}$:
        (i) Set CLUSTERID(u) := ID(u);
        (ii) u broadcasts a message of the form $\prec$ID(u), CLUSTERID(u)$\succ$;
  if v *receives a message of the form* $\prec$***id***, ***id***$\succ$ **then**
    if (CLUSTERID(v) == UNKNOWN) or (CLUSTERID(v) > **id**) **then**
      CLUSTERID(v) := **id**;
    if (CLUSTERID(v) $\neq$ UNKNOWN) and (CLUSTERID(v) $\neq$ **id**) **then**
      Set GATEWAY(v) := GATEWAY(v) $\cup$ {**id**};
end

We then have the following result associated to the BASICCLUSTERING protocol:

Theorem 2. *For any fixed constant $\ell > 0$, there exists a constant $C(\ell)$ such that if the transmission radius of each station is set to $r = \sqrt{\frac{(1+\ell)\,|S|\,\log n}{\pi n}}$ then with probability tending to 1 as n tends to ∞, after an execution of* BASIC-CLUSTERING, *every participating node knows the identity of its cluster and the clusters are non-overlapping.*

Proof. The proof of Theorem 2 is close to that of Theorem 1. Therefore, such proof is only sketched in this extended abstract. By Theorem 1, after one invocation of EXCHANGEID, with high probability, every node is aware of its neighborhood. A node u with the (local) lowest-ID is know ready to become the clusterhead of its neighbors. The next figure shows briefly why such a node has to compete to resolve (probable) ties. By the same arguments as given above, we insure that with high probability the duration of the for loop (viz. $C(\ell)\log(n)^2$) is sufficient for the clusterheads to send their messages to all their neighbors.



In the small graph depicted on the left, the nodes 1 and 2 are the lowest-IDs of their respective neighbors. Since they are both neighbors of the node 3, processors 1 and 2 have to compete in order to send the right informations to 3. This can be done by means of randomness (cf. the for loop in the protocol).

Algorithm 3: k-CLUSTERING

begin
  For each node u set CLUSTERID(u) := ID(u) and GATEWAY$(u) := \emptyset$;
  Each node u starts diffusing its own identity and
  an initial value called "*distance*" is set to **distance** := 1.
  SEND($C(\ell)\log(n)^2$, $\prec$ID(u), **distance**$\succ$) ;
  if *u receives* ***for the first time*** *a message from a node v* **then**
    Suppose that the message is of the form $\prec v, d\succ$:
    if $1 \leq d < k$ **then**
      Set **distance** := d and forward the message :
      SEND($\alpha_k \log(n)^2$, $\prec v$, **distance** $+ 1\succ$); (♡)
      if *v is the lowest known* ID **then**
        CLUSTERID$(u) := v$;
    else
      ($\star$ u is a **gateway** *iff* $d == k$ and CLUSTERID$(u) == v$ $\star$)
      Set GATEWAY(u) := GATEWAY$(u) \cup \{v\}$;
      It has to advert its neighbors with a special message :
      SEND($C(\ell)\log(n)^2$, $\prec$CLUSTERID(u), GATEWAY$\succ$) ;
  if *u receives a message of the form* $\prec v$, GATEWAY$\succ$ **then**
    u is also a *gateway* : Set GATEWAY(u) := GATEWAY$(u) \cup \{v\}$;
end

3 Generalized k-Clustering Algorithms

In this section, we shall describe algorithms for modifying the size of the clusters. Given an integer k[2], known by each node, the k-clustering algorithms build disjoint clusters containing nodes at distance at most k hops from the clusterhead. Note also that two clusterheads must be at distance at least $k+1$ hops each other. The k-clustering algorithm works as follows. Each node's first task is to "flood" its k-hop neighbors with its ID. Each time a node u "hears" a new ID, he has to compare this latter to the last lowest known ID. As for the basic protocol, the lowest-ID node among its k-hop neighbors is "chosen" to be the clusterhead. The ID of this clusterhead is stored and forwarded at most k times but at least once.

We observe here that in the algorithm (cf. line ♡), we intentionally choose α_k satisfying

$$\alpha_k = \left(2 + \frac{2(k+1)\log\log n}{\log n}\right) C_\ell \, . \tag{1}$$

To prove the correctness of the algorithm, let us compute first an upper-bound of the number of k-neighbors of a given node u:

[2] It is important to note here that k can depend on n, e.g. $k = \lfloor \log\log n \rfloor$.

Lemma 1. *Suppose that n nodes are deployed randomly uniformly on a surface of size $|\mathcal{S}| = O(n)$ and suppose that their transmission ranges are set to $r = \sqrt{\frac{(1+\ell)\,|\mathcal{S}|\,\log n}{\pi n}}$. For a node u, the number of nodes at most at k hops from u is with high probability less than $O(k(n)\,\log(n))$.*

Note that, lemma 1 tells us that $k \equiv k(n)$ can tends to ∞ with n.

Theorem 3. *The* k-Clustering *protocol terminates in at most* $O\Big(\max\Big(k(n)\,\log(n)^2, k(n)^2\,\log\log n\,\log(n)\Big)\Big)$ *steps.*

Proof (sketch). By lemma 1, any given node u has at most $O(k\,\log(n))$ k-hop neighbors. In our algorithm, since a node u will forward only messages from new k-hop neighbors, there are at most $O(k\,\log(n))$ attempts to forward the received messages from these neighbors. By construction (cf. the line marked with the symbol ♡ above), the time complexity of each attempt is $O(\alpha_k \log(n)^2)$ and the proof of theorem 3 is now complete.

Theorem 4. *Suppose that n stations are deployed randomly uniformly on a surface of size $|\mathcal{S}| = O(n)$ and suppose that their transmission ranges are set to $r = \sqrt{\frac{(1+\ell)\,|\mathcal{S}|\,\log n}{\pi n}}$. After one invocation of the protocol* k-Clustering *and with high probability,* **(i)** *every node knows its unique cluster and* **(ii)** *every gateway node knows the list of its adjacent clusters.*

References

1. Akyildiz, I. F., Su, W., Sankarasubramaniam, Y. and Cayirci, E. Wireless sensor networks: a survey. *Computer Networks* 38: 393–422, 2002.
2. Al Agha, K., Pujolle, G. and Vivier, G. *Réseaux de mobiles & réseaux sans fil.* Éditions Eyrolles, 2001.
3. Basagni, S. Distributed Clustering for Ad Hoc Networks. *in Proc. Inter. Symp. Par. Architectures, Algorithms and Networks (ISPAN)*, 1999.
4. Battiti, R., Bertossi, A. A. and Bonuccelli M. A. Assigning Codes in Wireless Networks: Bounds and Scaling Properties. *Wireless Networks*, 5: 195–209, 1999.
5. Black, U. *Mobile and Wireless Networks.* Prentice-Hall, 1996.
6. Chatzigiannakis, L., Nikoletseas, S. and Spirakis, P. Efficient and Robust Protocols for Local Detection and Propagation in Smart Dust Protocols. *To appear in ACM/Kluwer Mobile Networks and Applications*, 2004.
7. Cheng, Y.-C. and Robertazzi T. G. Critical connectivity phenomena in multihop radio models. *IEEE Trans. on Communications*, 36: 770–777, 1989.
8. Corless, R. M., Gonnet G. H., Hare D. E. G., Jeffrey D. J. and Knuth D. E. On the Lambert W Function. *Advances in Computational Mathematics*, 5: 329–359, 1996.
9. Deogun, J. S., Kratsch D. and Steiner G. An approximation algorithm for clustering graphs with dominating diametral path. *Inf. Proc. Letters*, 61(3): 121–127, 1997.
10. Ephremides, A., Flynn, J. A. and Baker, D. J. The design and simulation of a mobile radio network with distributed control. *IEEE Journal on Selected Areas of Communications* 2(1): 226–237, 1984.

11. Estrin, D., Govindan, R., Heidemann, J. and Kumar S. Next century challenges: Scalable coordination in sensor networks. *Proceedings of IEEE/ACM International Conference on Mobile Computing and Networking*, pp 263–270, 1999.
12. Fernandess, Y. and Malkhi, D. K-Clustering in Wireless Ad Hoc Networks. *in Proceedings of ACM POMC 2002*.
13. Garcia Nocetti, F., Solano Gonzalez, J. and Stojmenovic, I. Connectivity Based k-hop Clustering in Wireless Networks. *Telecommunication Systems*, 22: 205–220, 2003.
14. Gerla, M. and Tsai, J. T. C. Multicluster, mobile, multimedia radio network. *ACM/Kluwer Wireless Networks*, 1: 255–265, 1995.
15. Gilbert, E. N. Random Plane Networks. *Journal of the Society for Industrial and Applied Math*, 9: 533–543, 1961.
16. Gupta, P. and Kumar P. R. Critical power for asymptotic connectivity in wireless networks. *Stochastic Analysis, Control, Optimization and Applications: a volume in honor of W. H. Fleming, W. M. McEneaney, G. Yin and Q. Zhang*, Birkhauser, Boston, 1998.
17. Gupta, P. and Kumar P. R. Internet in the Sky: The Capacity of Three Dimensional Wireless Networks. *Communications in Information and Systems*, 1:33–49, 2001.
18. Hall, P. *Introduction to the Theory of Coverage Processes.* Birkhäuser, Boston, 1988.
19. Krishna, P., Vaidya, N. H., Chatterjee, M. and Pradhan, D. K. A cluster-based approach for routing in dynamic networks *ACM Computer Comm. Review*, vol. 27, 1997.
20. Krishnamachari, B., Wicker S. B., Bejar R. and Pearlman M. Critical Density Thresholds in Distributed Wireless Networks. Book chapter in *Communications, Information and Network Security, eds. H. Bhargava, H.V. Poor, V. Tarokh, and S. Yoon, Kluwer Publishers*, 2002.
21. Kushilevitz, E. and Mansour, Y. An $\Omega(Dlog(N/D))$ Lower Bound for Broadcast in Radio Networks. *SIAM Journal on Comput.* 27(3): 702–712, 1998.
22. Lin, C. R. and Gerla, M. Adaptive clustering for mobile wireless networks. *IEEE Journal on Selected Areas of Communications* 15(7): 1265–1275, 1997.
23. Malpani, N., Vaidya, N. and Welch J. L. Leader Election Algorithms for Mobile Ad Hoc Networks. *Proceedings of 4th Int. Workshop on Disc.Algorithms and Methods for Mobile Computing and Communications – DIALM 2000*, pp. 96–103.
24. Miles, R. E. On the Homogenous Planar Poisson Point Process. *Math. Biosciences*, 6: 85–127, 1970.
25. Parekh, A. K. Selecting routers in ad hoc wireless networks. *Proceedings of SBT/IEEE Int'l Telecomm. Symp.*, 1994.
26. Perkins, C. E. *Ad Hoc Networking.* Addison-Wesley, 2001.
27. Philips, T. K., Panwar, S. S. and Tantawi, A. N. Connectivity properties of a packet radio network model. *IEEE Trans. on Information Theory*, 35: 1044–1047, 1989.
28. Ravelomanana, V. Extremal Properties of Three-Dimensional Sensor Networks with Applications. *IEEE Trans. on Mobile Computing*, 3: 246–257, 2004.
29. Ravelomanana, V. Randomized Initialization of a Wireless Multihop Network. To appear in *Proceedings of the* 38^{th} *HICSS*, 2005.
30. Santi, P. and Blough D. M. The Critical Transmitting Range for Connectivity in Sparse Wireless Ad Hoc Networks. *IEEE Trans. on Mobile Computing*, 2: 1–15, 2003.

Call Admission Control with SLA Negotiation in QoS-Enabled Networks

Iftekhar Ahmad, Joarder Kamruzzaman, and Srinivas Aswathanarayaniah

Gippsland School of Computing and IT, Monash University, Australia
{Iftekhar.Ahmad, Joarder.Kamruzzaman, Srinivas.Aswathanarayaniah}@infotech.monash.edu.au

Abstract. This paper presents a Service Level Agreement (SLA) negotiation technique for Instantaneous Request (IR) call connections based on information in context of Book-Ahead (BA) reservation. Resource sharing between IR and BA reservation imposes a number of problems in a QoS-enabled network. One of the problems is preemption of on-going IR calls. Call admission control models proposed in literature reduce high preemption rate at the cost of higher call blocking rate and lower resource utilization. The negotiation technique proposed in this paper is based on information like look-ahead time for BA calls and negotiable requirements of IR calls. The technique allows admission of some of the IR calls that would otherwise be blocked due to resource scarcity arising from BA call activation. Simulation results show that the proposed negotiation based call admission control model achieves the desired objective of higher resource utilization and lower call blocking rate.

1 Introduction

For years, bandwidth reservation is one of the most important problems in network management, specially when the network is designed to provide guaranteed Quality of Service (QoS). In general two types of resource reservation in computer networks are distinguished i) Instantaneous Request (IR) reservation and ii) Book-Ahead (BA) reservation. IR reservation is made immediately after the call acceptance while in BA reservation resource reservation is confirmed well ahead of usage time. BA reservation is highly attractive for high bandwidth requiring time-sensitive applications which require strict quality of service. Applications like multi-party video conferencing, video on demand, live broadcast of TV programs, medical applications like remote surgery or telemedicine, teleteaching, distance learning, grid computing, distributed simulations etc. require book-ahead reservation [1-4]. Resource sharing between IR and BA reservation imposes a number of problems because of their dissimilar style of resource reservation. One of the problems is preemption of on-going IR call connections used to make the resources available for BA calls. In a QoS-enabled network, high number of preemption of on-going calls results in high user dissatisfaction because of disruption of service continuity. Recent studies [5] show that uninterrupted service is a very important metric for qualitative QoS perceived by users.

A number of research works have been conducted to reduce preemption rate in a QoS-enabled network in the context of BA reservation. Schelen *et al.* [3] proposed a

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 117–124, 2005.

Constant Look-Ahead Time (CLAT) based model to reduce the number of preemptions by introducing the concept of look-ahead time. Look-ahead time is defined as the pre-allocation time, i.e., the time for starting to set aside resources for advance reservations so that there is no resource scarcity at the stating time of a BA call. Greenberg *et al.* [2] proposed an approximate interrupt probability based admission control scheme. The scheme shows that resource sharing between IR and BA calls achieves better network performance than strict partitioning of resources proposed in [6]. This model heavily depends on the prediction accuracy of IR call duration. Accurate prediction of call holding time has been considered as a very complex issue and this is why most of the recent works assume the call holding time of an IR call as open ended [3, 4]. Lin *et al.* [1] proposed an Application Aware Look-Ahead Time (AALAT) based call admission control model which considers different look-ahead time for different applications. AALAT model also depends on correct prediction of call duration. More importantly AALAT model is not suitable for a medium to large size network as it requires the traffic pattern at each link to find look-ahead time. These considerations make the CLAT model more generalized and suitable for a computer network of any size. However, CLAT model uses constant value of look-ahead time which is not justified in consideration of dynamicity of a real-time computer network. Ahmad *et al.* [4] proposed a Dynamic Look-Ahead Time (DLAT) based call admission control model that considers network dynamicity and achieves better network performance than CLAT based CAC model. All of the look-ahead time based models block an incoming call that arrives within the look-ahead time and is likely to result in over-allocation of resources upon activation of BA calls. None of the works in literature has considered the usage of negotiation on service level agreement (SLA) in terms of bandwidth demand and call duration for possible admission of IR calls that would otherwise be blocked. Although negotiation and re-negotiation of QoS are not completely a new technique and have been applied in a QoS-enabled network for long [7-9], all of them are based on available bandwidth information. The work presented in this paper shows a novel approach of negotiation of SLA for calls that arrive in look-ahead time and are blocked in other schemes to leave enough resources for BA calls. The proposed scheme offers those IR calls a chance for admission at negotiated level without compromising BA requirements. Simulation results show that negotiation of SLA for calls arriving within look-ahead time is a promising technique to achieve better network performance in the form of utilization and call blocking rate.

2 Problem Definition

A BA call needs to announce two extra parameters, it's starting time and call holding time in addition to the nominal QoS parameters like bandwidth demand, packet loss ratio, end to end delays, jitter etc. Call Admission Control (CAC) algorithm designed for book-ahead calls checks whether there will be enough resources for that BA call at that particular starting time and for the period of the announced duration at each node along the path from source to destination. A BA call is required to make the request well in advance to its actual starting time. Resource usage time for an IR call is immediate and call holding time is open ended. An IR call request only expresses the

QoS parameters for Service Level Agreement (SLA). Once the call is admitted, the SLA is valid for the whole lifetime of the call. Problem arises when a BA call becomes active at certain point of lifetime of an IR call and there are no resources available to support the BA call. Since the service for BA call is already confirmed in advance and the application is highly time-sensitive, the only option is to preempt resources from some IR calls and make room for BA call. Preemption of IR calls disrupts service continuity. Preemption probability which indicates the ratio of preempted IR calls to accepted IR calls is thus one of the metrics that measures the users' perceived QoS.

3 Call Admission Models to Reduce Preemption Rate

CLAT based call admission scheme proposed by Schelen *et al.* [3] uses a single constant value of look-ahead time for call admission decision of IR calls. Resources are set-aside for the look-ahead period immediately before the activation of a BA call and this reduces the number of IR call preemptions. DLAT model proposed by Ahmad *et al.* [4] calculates look-ahead time taking the dynamicity of traffic pattern and network state into consideration. It dynamically updates the value of look-ahead time at regular time interval. Look-ahead time is calculated by the following equation:

$$LAT(t,s_i) = \frac{A(s_i) + R(t) + (1+l)\sigma(\mu_{IR}) - C}{\mu_{IR}\lambda_{IR}(1-b)} + \sigma\left(\frac{1}{\lambda_{IR}}\right)c \tag{1}$$

Here $LAT(t,s_i)$ is the look-ahead time w.r.t traffic condition at current time t and BA activation time s_i ($t<s_i$). $A(s_i)$ is the aggregate bandwidth reserved for BA calls to be activated at time s_i, $R(t)$ is the aggregate bandwidth used by IR calls at time t, μ_{IR} is the mean bandwidth demand of IR calls, λ_{IR} is the mean arrival rate of IR calls, l is the normalized BA limit which determines maximum allowable aggregate BA load, b is the call blocking probability for IR calls, $\sigma(.)$ is the standard deviation and c(>1) is a tuning parameter. The value of 'c' influences the look-ahead time and can be tuned to achieve the desired preemption probability.

$LAT(t,s_i)$ is calculated at regular intervals of operating time for a number of entries in the book-ahead table. For a particular entry it is calculated only when the term $A(s_i) + R(t) + (1+l)\sigma(\mu_{IR}) - C$ in Eq. (1) is found positive, otherwise it is set zero. IR calls are checked against the following rule at call admission time.

$$C > \max_{s_i \in LAT} (r + R + A(s_i)) \tag{2}$$

At each interval only those entries are taken into calculation for which the following rule satisfies

$$t > s_i - \frac{A(s_i) - A(t)}{(1-b)\mu_{IR}\lambda_{IR}} \tag{3}$$

A detailed description of the model and algorithm can be found in [4].

4 Proposed Model for SLA Negotiation of IR Calls

QoS requests are quantitatively described in terms of technical specification like end to end delay, jitter, packet loss rate etc. A set of such parameters along with guarantees about reliability are called a Service Level Agreement (SLA). A SLA works as a contract between a client and a service provider. Once a SLA is set up it is expected to remain stable in a QoS-enabled network that provides high quality service.

In previous section, it was shown that an IR call that arrives within the look-ahead time and causes over provisioning of bandwidth at the activation of nearby BA call is blocked (Eq. 2) to reduce high preemption rate. Call admission decision in this case is based on the assumption of open ended IR call duration. However, in most of the practical cases, users may have some perceived value on call duration when they wish to access the network services. It is thus highly likely that a good number of users will be satisfied if they are allowed to access the resources until the activation of BA calls. In such case it is necessary to negotiate the SLA based on the information about the duration for which the network can guarantee to provide strict QoS at that stage. Moreover, it is possible to adjust the bandwidth requirement for a number of particular types of applications like guaranteed express data transfer (bulk banking data) or non real-time variable bit rate data transfer (e.g., MPEG files). Bandwidth requirement for some applications can again be lowered (e.g., by proper method of transcoding of multimedia data) to some level so that there exists no chance of resource scarcity even after the activation of BA calls. Of course this will cause degradation in quality and it is thus important to negotiate on the quality that the network is able to provide at the current stage. In this paper, we consider these two issues: i) negotiation on call duration and ii) negotiation on bandwidth requirement for the blocked calls. The proposed model considers call admission control as follows:

Step 1: Determine if the bandwidth usage, after adding the new load r of an IR call arriving at time t, will exceed the available link capacity C during the look-ahead time LAT.

$$C > \max_{s_i \in LAT} (r + R + A(s_i))$$

Step 2: If it exceeds the link capacity upon activation of a BA call at time s_i, then negotiate with the client. If negotiation fails then block the call, otherwise accept the call.

Negotiation can be done in two ways:

i) *Lifetime*: If the client is satisfied with the call duration (s_i-t) with reliable QoS, then accept the call. If the client is unsure, but the amount of data to be transferred Z is known then the client application calculates call duration d as follows

$$d = Z/C_{max}$$

where C_{max} is the maximum transmission capacity supported by sending and receiving device given that $C_{max} < C-(A(t)+R(t))$.

If $d < (s_i-t)$ then accept the call, otherwise go to the next step.

ii) *Bandwidth*: Negotiate on the loss in quality which will occur due to allocation of lower than requested bandwidth (within $C\text{-}(R+A(s_i))$, the bandwidth that remains available after the activation of BA call at time s_i) determined by a proper technique (e.g., transcoding for multimedia). If the client agrees to the offered quality under constrained bandwidth then accept the call, otherwise block the call. Flowchart of the proposed model is given in Fig. 1.

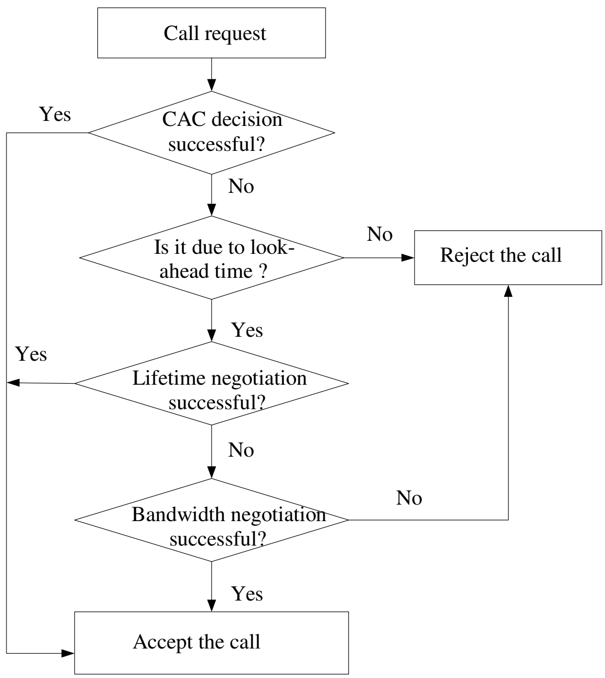


Fig. 1. Flowchart for SLA negotiation of IR calls

5 Simulation Results

Simulation was conducted with the similar single bottleneck topology used in a number of research works [1, 3, 4]. The capacity of each link is assumed to be 15 Mbps. Arrival of IR and BA calls connection is assumed to follow Poisson distribution with mean arrival interval of 7s and 60s respectively. Call holding time of each type of call is assumed to be exponentially distributed with a mean lifetime of 300s. Bandwidth demand of IR and BA calls is assumed to be exponentially distributed with a mean of 256 kbps and 2.25 Mbps respectively. As a preemption strategy, IR calls in order of least time in the network are preempted as it minimizes the amount of wasted throughput [2]. Probability of a client to agree to the offered call duration is considered as 0.2, probability of a client's ability to increase its transmission rate is assumed to be 0.05 and probability of a client's ability to change it's bandwidth requirement to some lower level is taken as 0.2 for the results shown in Fig. 2-3. The impact of other probability values on network performance is also investigated and reported later in this section. The value of tuning parameter ‘c’ in DLAT model is considered as 9.0 for the results shown in Fig. 2-5. Constant value of look-ahead time at each BA limit in CLAT model is adjusted to achieve the same preemption probability as achieved by DLAT model at that BA limit. This makes the platform to compare different models keeping the preemption probability same.

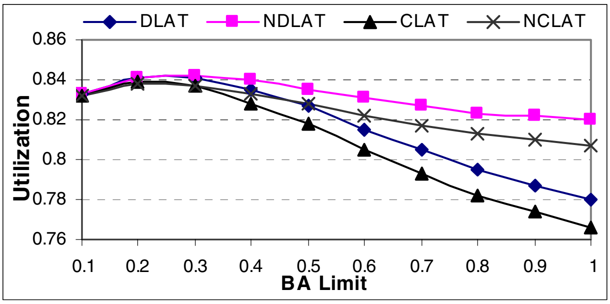


Fig. 2. Bandwidth utilization in different models for different BA limits

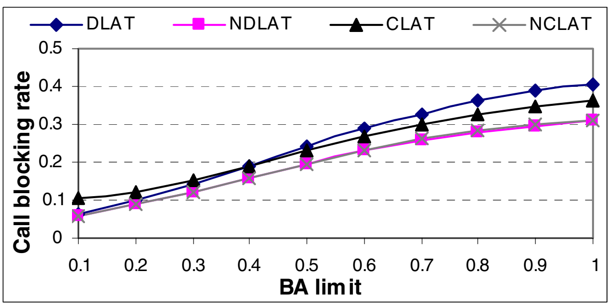


Fig. 3. IR call blocking rate in different models for different BA limits

Figure 2 shows that when the proposed negotiation technique is applied with DLAT (NDLAT) and CLAT (NDLAT) models, utilization improves quite significantly. This is illustrated by the highest utilization achieved by NDLAT model for all BA limits. For moderate BA limit the relative improvement (NDLAT vs DLAT, NCLAT vs CLAT) is more than 1% and at higher BA limits (>0.8) it is very close to 4%. This is because a large number of calls which would otherwise be blocked for appearing within look-ahead time under the CAC rule (Eq. 2) are now accepted when negotiation on lifetime and bandwidth is applied. Ability to accept more calls has another advantage of low call blocking rate and this is shown in Fig. 3. Figure 3 indicates that call blocking rate improves to a great extent when negotiation on lifetime and bandwidth (NDLAT, NCLAT) is applied. Relative improvement (NDLAT vs DLAT) is highly significant for most of the BA limits and at higher BA limits (BA limit >0.8) the improvement is as high as 10%. NCLAT model is also found to outperform CLAT model by more than 4% for BA limit value 0.5 and higher. Further observation confirms that there is very little difference in achieved preemption probability when negotiation technique is applied. The reason is that preemption probability depends on the technique to determine look-ahead time which remains the same in negotiated scheme.

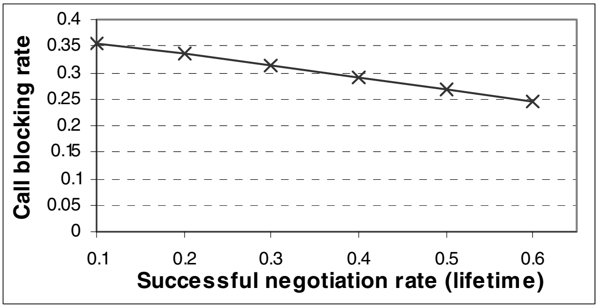


Fig. 4. IR call blocking rate at different lifetime negotiation rate in NDLAT model

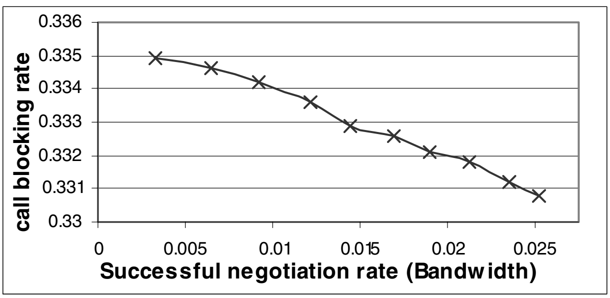


Fig. 5. IR call blocking rate at different bandwidth adaptation rate in NDLAT model

The impact of lifetime negotiated call acceptance rate on network performance was also investigated keeping other factors unchanged. It is observed that as more users agree to access the network resources within time constraints (completion before activation of BA calls), link utilization increases quite sharply. When acceptance of calls within lifetime constraint increases, it is expected that the call blocking rate will drop and the same is observed in Fig. 4. Call blocking rate drops quite shapely for increasing negotiated call acceptance rate. It is also found that preemption probability increases slightly with increasing lifetime negotiated call acceptance rate. This is because acceptance of more calls with constraint lifetime (negotiated lifetime) leaves relatively small amount of resources for calls entering the system without lifetime constraint (open ended lifetime). This indicates that the mean bandwidth of calls entering the system without negotiation is smaller now and under such situation when a BA call becomes active and requires preemption of calls in order of Last In First Out fashion, relatively large number of calls need to be preempted because of their smaller size. This is why preemption probability increases slightly with increasing lifetime negotiation rate.

The impact of changing bandwidth negotiated call acceptance rate on utilization, preemption probability and call blocking rate was also investigated. It is observed that utilization improves with increasing acceptance rate. As the bandwidth negotiated call acceptance rate increases call blocking rate decreases (Fig. 5) and preemption probability increases.

6 Conclusion

This paper presents an effective technique for SLA negotiation of Instantaneous Request (IR) calls based on information in relation to Book-Ahead (BA) calls in a QoS-enabled network that supports both BA and IR reservation. A BA call upon its activation often causes preemption of many on-going IR calls on resource scarcity. To maintain desired level of service continuity it is very important to maintain a low preemption rate in a QoS-enabled network. Look-ahead time based call admission control models are found to successfully reduce high preemption rate at the cost of lower utilization and higher call blocking rate. This paper shows that IR calls blocked by look-ahead time based CAC models can be admitted if the information about guaranteed IR call lifetime and bandwidth is used for SLA negotiation. Simulation results show that when the look-ahead time based CAC models are complemented by the proposed negotiation technique, resource utilization increases and call blocking rate decreases quite significantly.

References

1. Y.Lin, C. Chang, and Y. HSU, "Bandwidth brokers of instantaneous and book-ahead requests for differentiated services networks," *ICICE Trans. Commun.*, vol. E85-B, no.1, pp. 278-283, 2002.
2. A.G. Greenberg, R.Srikant, and W. Whitt, "Resource sharing for book-ahead and instantaneous-request calls," *IEEE/ACM Trans. Networking*, vol.7, pp.10-22, 1999.
3. O. Schelen and S. Pink, "Resource sharing in advance reservation agents," *Journal of High Speed Networks*, *Special issue on Multimedia Networking*, vol. 7, no. 3-4, pp. 213-218, 1998.
4. I. Ahmad, J. Kamruzzaman, and S. Ashwathanarayaniah, "Dynamic look-ahead time in book-ahead reservation," *Proc. IEEE ICON* 2004, pp. 566-571, Singapore, 2004.
5. M. Campanella, P. Chivalier, N. Simar, "Quality of Service Definition," *http://www.dante.net/sequin/QoS-def-Apr01.pdf.*
6. D. Ferrari, A. Gupta, and G. Ventre, "Distributed advance reservation of real-time connections," *Proc. NOSSDAV, Lecture Notes in Computer Science*, pp.15-26, Durham, 1995.
7. T.F. Addelzaher, E.M. Atkins, and K.G.Shin, " QoS negotiation in real-time systems and its application to automated flight control," *IEEE Transaction on Networks*, vol. 49, no. 11, pp. 1170-1183, November 2000.
8. A.L. Chan, and K.L.E. Law, "QoS negotiation and real-time renegotiations for multimedia communications," *Proc. Intl. Conf. on Comp. Comm. and Networks*, pp. 522-525, 2002.
9. I. Ahmad, J. Kamruzzaman, and S. Aswathanarayaniah "An efficient technique for bandwidth allocation by dynamic pricing," *Proc. IASTED Int. Conf. Comm. Internet and Inf. Tech, CIIT2003,* pp. 491-496, USA 2003.

Enhancing QoS Through Alternate Path: An End-to-End Framework

Thierry Rakotoarivelo[1,2,3], Patrick Senac[2,3], Aruna Seneviratne[4], and Michel Diaz[3]

[1] University of New South Wales, Sydney NSW 2052, Australia
thierry@mobqos.ee.unsw.edu.au
[2] ENSICA, 1 Place Emile Blouin, 31056 Toulouse, France
senac@ensica.fr
[3] LAAS-CNRS, 7 Avenue du Colonel Roche, 31077 Toulouse, France
diaz@laas.fr
[4] National ICT Australia[1], Locked Bag 9013, Alexandria NSW 1435, Australia
aruna.seneviratne@nicta.com.au

Abstract. In the next generation Internet, the network should not only be considered as a communication medium, but also as an endless source of services available to the end-systems. These services (i.e. Overlay Applications) would be composed of multiple cooperative distributed software elements that dynamically build an ad hoc communication mesh (i.e. an Overlay Association). In this paper we propose and evaluate a collaborative distributed method to provide enhanced QoS between end-points within an overlay association.

1 Introduction

In the last few years, there has been a steady increasing demand for mobile network-enabled devices. These devices collectively form a pervasive networking environment around the user. A possible approach to ensure low device cost could be to limit the available resources on the device, and rely instead on the network to provide them. Following this approach, Service Providers at the edge of the network would provide end-systems with distributed applications, computing or storage capabilities. These services would be composed of multiple cooperative distributed software elements, performing elementary tasks, and communicating with each other [1, 2]. In a particular instance, these distributed application elements dynamically build an ad hoc communication mesh that forms an overlay network above the existing infrastructures. In this context, we use the expression "Overlay Application" to refer to the distributed application composed by such elements. Within an overlay application, data flows no longer travel between just two end-points, but may instead *traverse* multiple peer end-points (hosting processing application elements). This defines a new peer-to-peer communication scheme different from the traditional point-to-point, or point-to-multipoint. For example, multimedia flow in a complex

[1] National ICT Australia is funded through the Australian Government's *Backing Australia's Ability* initiative, in part through the Australian Research Council.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 125–132, 2005.

distributed conference service (i.e. an instance of overlay application) might pass through several peers hosting elementary processing elements (e.g. stream extractions, language translation, etc…). Current overlay application architectures implement such communication need with a juxtaposition of independent point-to-point connections using underlying traditional transport services. However, we could consider these connections as a unique entity: an Overlay Association, which would provide a transport layer abstraction to this new communication scheme. Using this concept, we could design mechanisms to manage as a whole the Quality of Service (QoS) experienced by the end-user(s) of the conference service. Such mechanisms would perform global resource optimization for an overlay association, as opposed to a non-optimal composition of local resource management decisions. Because of its critical adaptation role between the application and the network, the transport layer is the most appropriate place to deploy these functions. Such a transport layer would provide a unified means to manage the communication needs of overlay applications, and reduce application complexity. We introduced such framework in [3].

In this paper we propose a method to provide enhanced QoS between two end-points of an overlay association. This method can then be used in conjunction with algebraic properties and composition rules of QoS metrics, as discussed in [4], to guarantee enhanced QoS to an entire overlay association. For simplicity, we apply our method to only one additive QoS metric: the one-way delay. However, it could be adapted to accommodate other types of QoS metrics. Harnessing the conclusions from [5], we propose a distributed and collaborative method to find and construct QoS enhanced alternate Internet paths. This method is deployed on end-points at the transport level within our overlay framework. It confines the complexity at the edge of the network, hence requiring no modification on the existing routers, and no central administrative entity or third party QoS brokers.

Our contribution is twofold. First, in section 2 we confirm the feasibility of QoS enhancement through alternate paths at the scale of the European Internet, and we further investigate some characteristics of these paths. Second, in section 3 we provide a distributed scalable scheme to discover and deploy such paths. In section 4, we analyze the performance of our scheme. Finally, we present some related works in section 5, and conclude this paper in Section 6.

2 Providing Enhanced QoS Using Alternate Paths

In [5, 7], for a given directed Internet communication, the authors demonstrate the existence of alternate paths that provide in up to 80% of the cases a better QoS than the default Internet path. The existence of these *better* paths is mainly due to the Border Gateway Protocol (BGP) operations. For various reasons (e.g. economic partnership) network administrators may not consider QoS optimization as a primary factor when implementing BGP "routing policies". Therefore these policies may generate non QoS-optimal default end-to-end Internet paths. For a given pair of hosts, one could create an alternate path by selecting and composing consecutive end-to-end paths. As this alternate path may transit through different network components/links than the default one, it may experience better QoS.

Our trace-based simulation environment is based on end-to-end one-way delay measurements between 47 peers on the European Internet. These measurements are provided by the RIPE NCC Test Traffic Measurement (TTM) project [6]. We retrieved three 24h data sets in 2004: May 25, June 17, and July 12. These data sets provide about 2 one-way delay measurements per minute per directed pair of nodes.

2.1 Existence of QoS Enhanced Alternate Paths

We first considered alternate paths via one "relay" node (2-hop paths). We will analyze 3-hop paths at the end of this section. For each data set, we generated 1000 3-tuples <*src, dst, timestamp*> composed of a random source and destination nodes, and a random time in second. For each of these 3-tuples, we retrieved the corresponding one-way delay measurements (*delayref)* experienced on the default Internet path. Then, we executed a "brute force" search algorithm on the entire node set to find alternate paths with a lower one-way delay. Based on other works on TCP implementation [8], we account for the processing delay at the "relay" node by adding 1ms to the alternate path's delay before comparing it to *delayref*. On average over the three data sets, for 50.6% of the 3-tuples, there exists at least one alternative path that provides a better one-way delay than the default Internet path. This is consistent with the results presented in [5, 7].

2.2 Analysis of QoS Enhanced Alternate Path

We evaluated the following three characteristics: the number of *better* alternate paths that exist for a given 3-tuple, the gain obtained by using these paths, and the duration for which that gain holds (i.e. gain constancy). The following results are from the May 25 data set (2 other sets have similar results).

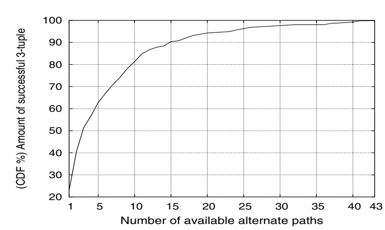


Fig. 1. CDF of available "better" alternate path among the "successful" 3-tuples

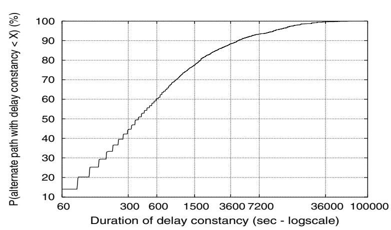


Fig. 2. CDF of alternate paths with delay constancy shorter than of equal to X

Figure 1 shows that around 76.97% of the "successful" 3-tuples (i.e. the ones with existing better alternate path) have more than 1 available *better* alternate paths (i.e. $P(1<X) = 76.97\%$). The mean number of available *better* alternate paths is about 6.47, with a standard deviation of 7.59. The existence of multiple *better* alternate paths for a given "successful" 3-tuple makes it easier for a distributed framework to discover at least one of them. Figures 3 and 4 present the one-way delay gain obtained through

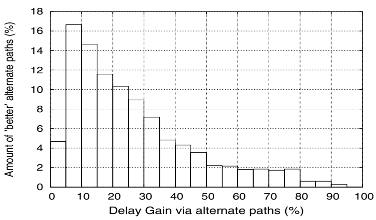


Fig. 3. Distribution of delay gain from "better" alternate paths (5% bins)

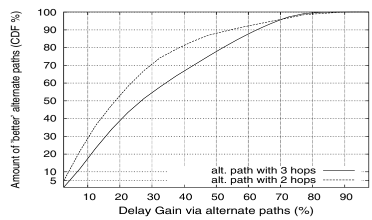


Fig. 4. CDF of delay gain from "better" alternate paths (5% bins)

2-hop alternate paths. The values are grouped in bins of 5%, hence the origin at 5% and 1% on figure 4. The mean gain on *better* 2-hop alternate paths is about 26.07% (compared to default Internet paths), with a standard deviation of 19.71. According to these results, 47.6% of *better* alternate paths provide a gain of more than 20%. Figure 2 presents the distribution of alternate paths that have their delay constancy shorter than or equal to a given value in seconds. Since default Internet paths follow the same trend, we can infer the duration of the gain constancy. Around 57% of the alternate paths have a delay constancy that lasts more than 5min. These results are consistent with [9]. Moreover in [10], the authors show that 53% of Internet streams last between 2s and 15min. Given these results, we can infer that most *better* alternate paths would offer gains with sufficient durations to benefit the majority of Internet streams. In a final experimentation, we found that there exist 42.4% of *better* 3-hop alternate paths (i.e. better than default IP path). We then successively ran the 2-hop and 3-hop search algorithms. The total percentage of "successful" 3-tuples grows by 1.7 point to reach 52.5%. Therefore the vast majority of 3-tuples with *better* 3-hop alternate paths have also 2-hop ones. Figure 4 shows that in more than 90% of the cases, the gain via the 3-hop paths is less than via the 2-hop paths. The benefit from deploying alternate paths with more than 2-hops is relatively small.

3 Finding QoS Enhanced Alternate Internet

In an overlay application, any hosts perform both as a client and a server. This is typically a peer-to-peer environment, and it seems natural to adopt a peer-to-peer approach to establish and manage overlay associations and related QoS. Overlay applications would then benefit from enhanced QoS without involving a central management entity (potential single point a failure). As there might potentially be thousands of overlay applications deployed at the same time, the proposed discovery scheme has to scale without penalizing other network users. These considerations led the design of a distributed peer-to-peer scheme, where the transport entity of a host involved in an overlay application would collaborate with other peers to gain partial knowledge of the network connectivity parameters, and to find *better* alternate paths towards other hosts involved in the same overlay application.

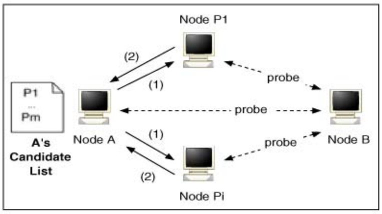


Fig. 5. First Search Level

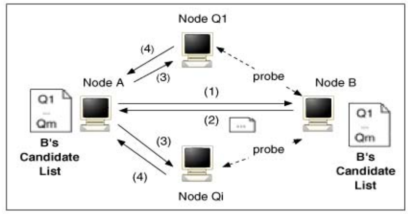


Fig. 6. Second Seach Level

During its initialization phase, a transport entity (TE_A) builds its initial partial knowledge of the network: a fixed size list (L_A). This list contains a selection of the currently known peers that have the best QoS values on the default Internet path from TE_A (e.g. the lowest one-way delay). To build its initial list, TE_A contacts a small set of bootstrap peers, requests their own lists (i.e. bootstrap lists), evaluates the QoS parameter on the default Internet paths towards nodes from these bootstrap lists, and selects the nodes with the "best" QoS values. To discover these bootstrap peers, TE_A can use a peer-to-peer directory service such as Chord [11]. For example, TE_A could join a Chord ring, and ask a fixed number of its Chord successors for their own lists. Then it evaluates its default one-way delay towards the nodes within these bootstrap lists, and incorporates in L_A the ones with the lowest values. As TE_A will discover other nodes via application requests, it will update its list accordingly. The size of the candidate list and number of bootstrap nodes are important scalability parameters. We fixed the list size to log N (N=number of participating peers). During initial deployments, peers could agree on an upper bound value for N, and fix their list size (and those of subsequent participants) accordingly.

Upon receiving an application request for a QoS enhanced path, TE_A executes a cooperative controlled-flooding algorithm to discover an alternate path that would best accommodate the required QoS. This algorithm is composed of two search levels, the second one being executed only if the first one is not successful (i.e. it failed to discover *better* alternate paths). It assumes that there are low cost techniques to evaluate a certain QoS parameter on a directed Internet path [12].

Figure 5 describes the first search level for a communication path between node A and B. TE_A probes node B to get the delay value (*delayref*) on the default Internet path from A to B. Then TE_A simultaneously sends a request (1) to all the nodes in its candidate relay list ($P_1 \ldots P_m$) asking them to evaluate the delay on their default Internet paths to node B. TE_i (P_i's transport entity) checks its available resources to assess its capacity to participate in an alternate path from TE_A. This task requires an admission control function, not discussed in this paper. If TE_i accepts to be a relay, it replies to TE_A with its delay value to TE_B, and keeps a temporary resource reservation, waiting for TE_A's path selection. Non-willing TE_is return an infinite delay value. When TE_A receives back these delay values (2), it computes the overall delay on each candidate alternate paths. It selects the values smaller than *delayref*, and compares them to retain the minimum one. If such value exists, TE_A designates the corresponding path as the alternate path to reach TE_B. In this case, the first search level is successful. The message cost to discover the alternate path is then equal to 2 * log N. Furthermore, it could be possible to design a delay estimate caching mechanism in each transport entity, hence removing the need to re-evaluate paths leading to already visited nodes. The admission control function on each node P_i together with TE_A's minimum delay path selection insure load balancing among the candidate relay nodes. Figure 6 describes the second search level. Upon failure of the previous search level, TE_A sends a request to TE_B (1) asking for its candidate relay list (2). Similarly to the previous search level, TE_A evaluates the delay values on the candidate alternate paths through the Qi nodes (from TE_B's list), (3) and (4). An alternate path is then selected. The message cost for this second search level is equal to 2 * (1 + log N). The first search level evaluates possible alternate paths by trying known first hops with lowest delay value on the directed path A to B. The second search level tries known last hops with lowest delay value on the directed paths towards B. To strictly do so, one might notice that TE_A needs to know the existence of these possible last hops. Moreover, TE_B only knows the hops with the smallest delay value on the directed paths from B (i.e. B's candidate list). IP routing mechanisms do not guarantee delay symmetry. However if delays on both ways of a given path were highly correlated, then TE_A could use B's candidate list. We computed a correlation coefficient of 0.83282 that supports this assumption.

Once TE_A has discovered and selected a best alternate path via a node P_i, it notifies TE_i and TE_B. TE_i turns its temporary resource reservation to a permanent one, and creates the necessary states to relay traffic from node A to B. TE_A monitors the experienced QoS. Upon eventual QoS degradation, it can discover another alternate path or use a cached alternate path discovered in the previous search phase. There are other important issues regarding the proposed scheme that we will investigate in our future works, such as security and admission control procedures for candidate relay nodes, or QoS management at the entire overlay association.

4 Simulation Results and Analysis

For each 3-tuple, we randomly selected 2 to 4 bootstrap nodes to build the initial candidate relay lists (with N=47, the list size is fixed to 6, i.e. $Log_2 47$). Then for each category of bootstrap node number, we executed our scheme's first level search algorithm followed by the second search level on the remaining unsuccessful 3-tuples. We averaged the results over 10 trials to account for the randomly selected bootstrap nodes. Table 1 shows that for a number of 4 bootstrap nodes, if for a given3-tuple there exist alternate paths with enhanced QoS, our proposed scheme has about 88% chance of discovering at least one of them. This result offers a satisfactory performance/cost trade-off considering the high cost of the brute force algorithm and the poor performances of the random one. From table 1, we can also infer that, in 86,5% of our scheme's successful cases, the message cost is 2 * log N. It is equal to 2 * (1 + 2 * log N) in the remaining 13.5%. One limit of our experiment is the fact that the TTM nodes are implicitly located on different networks. Building candidate lists with nodes located on the same network greatly diminishes the chance of finding

Table 1. Performance comparison for the search algorithms

	Message Cost	Discovered "successful" 3-tuples Bootstrap nodes 2	3	4
Brute Search	2 * N	100		
Search with random list	2 * log N	46.2 %	46.1 %	46.3 %
Only first search level	2 * Log N	61.8 %	71.1 %	76.1 %
Complete scheme	2*Log N to 2*(1+ 2*Log N)	77.6 %	82.8 %	88.0 %

alternate paths. A prototype of our scheme should have means to ensure the topological diversity of the nodes composing the candidate lists. Another limit is the small set of nodes, this does not allow extensive scalability test. However, it still provides a good realistic "snapshot" of the European Internet connectivity parameters. Topology generators do not provide a better alternative, since none of them accurately model end-to-end one-way delay (see [13] for such model). Finally, our experimentations do not take in account the dynamic evolution of the network. For each trial, the environment is not modified to reflect the resources being used by a previously selected alternate path. To overcome these biases, we plan to develop of a test-bed prototype in our future work.

5 Related Work

The DETOUR project [5] demonstrates the benefits of alternate Internet paths. Based on offline analysis of measurements across the North American wide area networks, the authors found that in up to 80% of the cases, for a given pair of nodes, there exists an alternate path that provides significantly superior QoS than the default Internet path. In [7], the authors proposed an application architecture to discover and deploy such paths. It relies heavily on a central entity (QRE), raising some scalability, robustness and administrative issues. Therefore, it might not be suitable for enhancing the QoS of distributed overlay applications. RON [14] is an application architecture that improves communication reliability and QoS by using alternate paths among RON nodes. It requires the presence and management of dedicated RON nodes in different Internet routing domains. For this reason, this framework might not be suitable to environments made of thousands different Internet domains. QRON [15] is based on hierarchically organized Overlay Brokers (OBs) located on different ASes. Third parties manage the OBs, and "sell" QoS enhanced alternate paths to users. In our peer-to-peer approach, end-hosts directly discover alternate paths without paying any third-party.

6 Conclusion and Future Work

We proposed a scheme to provide QoS enhanced communication path between two peers on the Internet. It is a part of a QoS management module within a overlay network transport layer framework. First, we analyzed some characteristics of QoS enhanced alternate Internet paths: in about 50% of the cases, there exists at least one alternate path providing in around 47.6% of the cases a significant durable gain of more than 20%. Second, we proposed a scalable distributed method to discover and construct such paths. This method is executed on the end-hosts at the edge of the network. If there exists any *better* alternate paths between two peers, it would find at least one of them in 77.5 - 88% of the cases. It does not guarantee 100% success. However, it empowers end-users, and does not rely on any central managements or third party brokers. We will continue our investigation of alternate path discovery methods, and our study of an overlay transport framework.

References

[1] X. Fu, et al. Cans: Composable, adaptive network services infrastructure. In Proc. of the UNSENIX Symposium on Internet Technologies and Systems, 2001.
[2] B. Raman, et al. The SAHARA model for service composition across multiple providers. IEEE Pervasive Computing 2002.
[3] T. Rakotoarivelo, et al. TOP: a Transport Overlay Protocol for Peer-to-Peer Applications. In Proc. of International Conference on Internet Computing. 2004
[4] Z. Wang and J. Crowcroft. QoS routing for supporting multimedia applications. IEEE Journal on Selected Areas in Communications, 1996.
[5] S. Savage, et al. The end-to-end effects of internet path selection. In Proc. of ACM SIGCOMM Conference, 1999.
[6] RIPE NCC Test Traffic Measurements project, see http://www.ripe.net/test-traffic/
[7] R. Beyah, R. Sivakumar and J. Copeland. Application layer switching: A deployable technique for providing quality of service. In Proc. of IEEE GLOBECOM, 2003.
[8] D. Clark, et al. An analysis of TCP processing overhead. IEEE Communication, 1989.
[9] Y. Zhang, et al. On the Constancy of Internet Path Properties. In Proc. of ACM SIGCOMM Internet Measurement Workshop, 2001.
[10] N. Brownlee and K.C. Claffy. Understanding Internet traffic streams: Dragonflies and tortoises. IEEE Communications, 2002.
[11] I. Stoica, et al. CHORD: A Scalable Peer-to-Peer lookup service for Internet Applications. In Proc. of ACM SIGCOMM, 2001.
[12] K.P. Gummadi, S. Saroiu and S.D. Gribble. King: Estimating Latency between Arbitrary Internet End Hosts. In Proc. SIGCOMM Internet Measurement Workshop, 2002.
[13] C.J. Bovy, et al. Analysis of end-to-end delay measurements in Internet. In Passive and Active Measurement (PAM), 2002.
[14] D. G. Andersen, et al. Resilient Overlay Networks. In Proc. ACM SOSP, Canada, 2001.
[15] Zhi Li and P. Mohapatra. QRON: Qos-aware routing in overlay networks. IEEE Journal on Selected Areas in Communications, 2004.

A Comparison on Bandwidth Requirements of Path Protection Mechanisms

Claus G. Gruber

Insitute of Communication Networks,
Munich University of Technology, Munich, Germany

Abstract. A large variety of resilience mechanisms are known today. However, often the used resilience mechanisms are not adapted to the network operator's and customer's needs. For this, a detailed analysis and a comparison of resilience mechanisms and capacity requirements is needed. In this paper we analyze the capacity requirements of three widely used shared path protection mechanisms with each other: shared global path protection, shared local link protection and shared local to egress protection. Additionally we present an optimization approach based on linear programming to obtain optimal working and resilience path configurations for a given network structure and apply these optimizations to different case study networks.

1 Introduction

Next to traditional Quality of Service (QoS) requirements as delay, delay variation (jitter), bandwidth and packet loss probability, fast and efficient resilience mechanisms are one of the key requirements to today's networks. Broadband technologies (e.g. ADSL, cable modem, PLC, FTTH) in combination with file sharing applications have and will further increase the amount of transported traffic in IP networks. In contrast to that however, the earnings of network providers have decreased due to global competition.

Thus, it is even more important today - than it was already in the past - to optimize a network and especially its resilience characteristics to fulfill the customer's requirements while reducing the overall network cost.

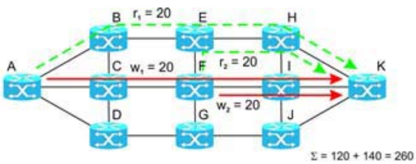
In order to choose suitable resilience mechanisms a comparison of characteristics of different approaches is required. The resilience mechanisms are dependent on the used forwarding mechanisms. Today, the most common Intra-Domain routing protocols use shortest path destination based routing (e.g. OSPF [1] and IS-IS [2]). Recently, Multi Protocol Label Switching (MPLS) [3] is deployed more often in IP networks in addition to these conventional shortest path based routing protocols due to the support of Virtual Private Networks (VPN) and the high flexibility considering traffic engineering and resilience. Thus, in this paper we will focus on path based resilience mechanisms that can be performed with MPLS.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 133–143, 2005.

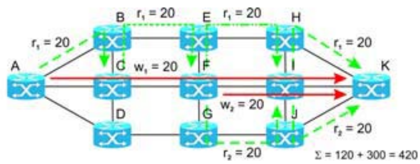
2 Protection Mechanisms

2.1 Shared Global Path Protection

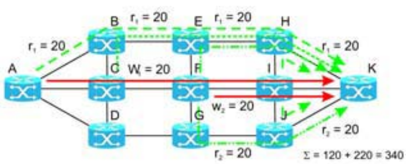
Figure 1(a) depicts an example configuration of two demands that are protected by a global path protection mechanism. Each working path transports 20 capacity units (w_1, w_2). If a failure occurs along the working path of one of the demands (e.g. link A-C, C-F, F-I, I-K) the nodes adjacent to the failure detect the failure and send a failure notification message to the source of the demand. The source node is then able to detour the traffic onto failure-free resilience paths. A sharing of backup resources is possible if working paths are routed disjoint and cannot be affected by (probable) simultaneous failures. Both demands of Figure 1(a) traverse the links F-I and I-K. Thus, if one of these links fails, both working paths are affected. Sharing of backup resouces is not possible in this constellation and 40 capacity units need to be reserved on link E-H and H-K for protection.



(a) Shared Global Path Protection



(b) Shared Local to Egress Protection



(c) Shared Local Link Protection

Fig. 1. Example of a path configuration for shared global path protection, shared local link protection and shared local to egress protection. The capacity requirements (working+protection) of the example configurations are shown in each figure

2.2 Shared Local Link Protection

Backward signaling takes its time and the reaction time upon a failure can be reduced if the detecting node (in front of the failure) detours the traffic immediately. In local link protection traffic traversing the failed link is detoured around the failure. The backup paths start in front of the failure and end at the other side of the failure. A sharing of resources can be performed if disjoint routed working paths can use the same protection capacities on detour links. Additionally one detour path is sufficient for all working paths traversing a failed link. Thus, the number of required backup paths in the network is small.

2.3 Shared Local to Egress Protection

Local link protection reduces the convergence time since no signaling need to be performed. However, if the network is high capacitated (i.e. few spare capacity is available) or sparsely meshed the local detour might be too long or no path can be found to return to the opposite side of the failure.

Local to egress protection combines the advantages of global path protection and local link protection with each other. Similar to local link protection the traffic is detoured locally at the node in front of the failure and no signaling is required. However, the backup path does not need to return to the working path but targets the sink of the demand (Figure 1(c)). With this, an increased flexibility and more paths are possible for the detour. The added delay that is caused by the detour is furthermore potentially smaller compared to local link protection. However, a combined detour of all affected paths traversing the failed link using only one resilience path is not be possible for paths with different destinations.

3 Considerations About Capacity Requirements

Figure 2 depicts a comparison of a path configuration example to show the relations between the capacity requirements of the resilience mechanisms. The left upper part shows a working and a protection path for global path protection. The left middle and left lower part shows a working and a protection path for local link protection.

The backup paths of shared global path protection are routed in a disjoint manner and quasi parallel to the working path. In local link protection, however, the traffic is detoured in front of the failure towards the opposite side of the failure. Thus, links which are parallel to the working path need also to be traversed.

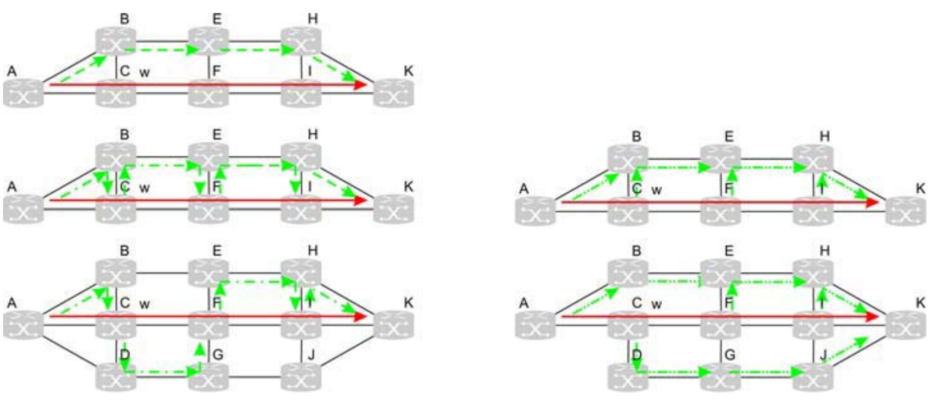


Fig. 2. Resilience resources comparison configuration. Left: Shared local link protection requires more capacity than shared global path protection. Right: Shared local to egress protection requires maximum the capacity of shared local link protection, however, more than shared global path protection

When concatenating the parallel links of all local link detours, a path disjoint to the working path can be formed. Additionally, detour links from and back to the working path are required. Independently of the topology and capacity constellation, a global protection path can thus be formed by concatenating local link protection paths. The required resilience capacity for shared global path protection is thus less or equal to the required resilience capacity of shared local link protection.

The right side of Figure 2 shows a comparison of local link and local to egress protection paths. Local to egress protection paths are allowed to return to the working path after a detour around the failure. No additional traffic has to be reserved for resilience purposes on the downstream side (after the failure) of the working path since the working traffic is detoured. The working capacity can be reused to transport the traffic towards the sink of the demand. Thus, local link protection requires at least the amount of resilience capacity used for local to egress protection. However, if the sum of required resilience capacity can be reduced by using other paths, local to egress protection requires less capacity than local link protection. Compared to global path protection however, additional capacity is required for local to egress protection to detour the traffic from the working path to the parallel backup capacities.

As a summary we can categorize the required protection capacities as follows:

Table 1. Protection capacity requirements of the investigated resilience mechanisms

`global path protection`	$\leq$	`local to egress protection`
`local to egress protection`	$\leq$	`local link protection`

The required capacity (sum of working and resilience capacity) of protection mechanisms is dependent on network structure, network dimensioning and the location of working and backup paths. Thus, in order to obtain the absolute and relative differences in capacity requirements results from optimal network configurations are required.

4 Optimization Equations

In this section we present a mathematical formulation based on linear programming for the optimal configuration of working and protection paths that are able to survive a single bidirectional link failure. We assume that the physical topology, demands and demand relations for a network are known and a dimensioning and configuration of working and backup paths are to be obtained.

As shown in Section 2 the protection mechanisms under investigation differ in the location of the detour and the possibility and location of the return to the working path. To highlight the small differences we divided the optimization problem in small modules. Additionally, this modularization allows the joint as well as the independent optimization of working and protection paths. A joint optimization results in less overall required capacity [4, 5]. However, considering on-line planning and an on-line reconfiguration of networks an independent optimization is required.

The network is modeled as a directed graph $G = (V, E)$. Where V represents the set of nodes and $E \in (V \times V)$ the set of edges of the network. Each edge is represented by a pair of counter-directional links. $DR \in (V \times V)$ denote the demand relations between two nodes whereas F denotes the set of failure patterns (bidirectional link failures). In the following, the superscript of a variable shows the set of numbers to which it belongs: $D \in R^+$, $I \in Z^+$ and $B \in boolean\ \{0; 1\}$.

4.1 Module Routing of Working Paths

We denote the variables $WPE^D_{d,e}$ as the working traffic of a demand d on a physical edge e. The outgoing (incoming) traffic of a node n for a demand d is represented by the variables $OWPN^D_{d,n}$ ($IWPN^D_{d,n}$). Its amount is equal to the traffic that is routed on outgoing (incoming) edges of the node for a demand d (equations (1) and (4)).

$$OWPN^D_{d,n} = \sum_{e \in out(n)} WPE^D_{d,e} \quad (1)$$

$$D^D_d >= OWPN^D_{d,n} \quad (2)$$

$$OWPN^D_{d,n} = 0 \quad (3)$$

$$IWPN^D_{d,n} = \sum_{e \in in(n)} WPE^D_{d,e} \quad (4)$$

$$D^D_d >= IWPN^D_{d,n} \quad (5)$$

$$IWPN^D_{d,n} = 0 \quad (6)$$

Equations (2) and (3) or (5) and (6) respectively are applicable if the node n is the source or the target of the demand. Routing loops are prevented and it is assured that the demand value D^D_d is routed between the two nodes. The relaxation of the strict equality ($\geq$ instead of $=$) is used to facilitate the work of the optimizer to find feasible results in a smaller amount of time during the solving process. To further satisfy a flow conservation, the incoming and the outgoing traffic of a demand d for a physical node n need to be equal if the node is not the source or the target of the demand (equation 7).

$$IWPN^D_{d,n} = OWPN^D_{d,n} \quad (7)$$

4.2 Module Basic Resilience

The module 'Basic Resilience' introduces variables and equations common to all resilience mechanisms. If a particular resilience mechanism has to be applied, we add some more specific equations and variables.

The variable $RPEF^D_{d,e,f}$ refers to traffic on a physical edge e used to protect demand d in case of failure pattern f. The sum of all outgoing (incoming) traffic out of (in) a physical node n that is used to protect traffic demand d for a failure pattern f can be calculated as follows:

$$ORPNF^D_{d,n,f} = \sum_{e \in out(n)} RPEF^D_{d,e,f} \quad (8)$$

$$IRPNF^D_{d,n,f} = \sum_{e \in in(n)} RPEF^D_{d,e,f} \quad (9)$$

Additionally, equation (10) prevents the routing of traffic on a failing egde ($e \in f$):

$$RPEF^{D}_{d,e,f} == 0 \tag{10}$$

4.3 Global Path Protection

If an edge e along a working path fails the traffic need to be detoured from the working path to the backup path. In global path protection the source node of the demand detours the traffic on a (disjoint) parallel path towards the target of the demand.

For the source node of the demand the outgoing traffic on the resilience paths need thus, be greater or equal to the traffic on the failed working path ($e \in F$):

$$ORPNF^{D}_{d,n,f} \geq WPE^{D}_{d,e} \tag{11}$$

$$IRPNF^{D}_{d,n,f} \geq WPE^{D}_{d,e} \tag{12}$$

For the target node of the demand the incoming traffic used to protect demand d need to be greater than the affected working traffic.

If the node n is neither the *source* nor the *target* of the demand d again flow conservation is required. The incoming detoured traffic is equal to the outgoing traffic along the resilience path (14).

$$IRPNF^{D}_{d,n,f} = ORPNF^{D}_{d,n,f} \tag{13}$$

To avoid a presence of loops the incoming (outgoing) traffic on a resilience path in (out of) a node n need to be zero if the node is the source (target) node of demand d (15).

$$IRPNF^{D}_{d,n,f} = 0 \tag{14}$$

$$ORPNF^{D}_{d,n,f} = 0 \tag{15}$$

4.4 Local to Egress Protection

In local to egress protection the traffic is detoured around the failure locally at the source of the failure and targets the sink of the demand.

We can reuse equations (11) and (14) if node n is in front of the failure and equations (12) and (15) if node n is the sink of the demand. Again for flow conservation we additionally need equation (13).

4.5 Local Link Protection

Similarly to local to egress protection in local link protection traffic is detoured in front of the failure. However, the traffic is reverted back onto the working path at the other end of the failure.

Equations (11) and (14) are required if the node n is the source node of the failed edge e and equations (12 and (15) if the node is the target node of failed edge ($e \in F$).

Beside these equations we need (13), if the node is not adjacent to the failure.

4.6 Capacity Calculation

The working capacity on an edge e (WCE_e^D) can be calculated as the sum of all working paths traversing this edge.

$$WCE_e^D = \sum_{d \in DR} WPE_{d,e}^D \tag{16}$$

If the resilience capacity cannot be shared between different working paths, the maximum required resilience capacity on an edge for the failure patterns (RCE_e^D) can be calculated as shown in equation (17).

$$RCE_e^D = \sum_{d \in DR} RPEF_{d,e,f}^D \tag{17}$$

If resilience capacity can be shared with each other, i.e. the protected working paths are routed disjoint to each other and cannot be affected simultaneously by the same failed link, the required resilience capacity can be reduced:

$$RCEW_{d,e}^D \geq RPEF_{d,e,f}^D \qquad \forall f \ in \ F \tag{18}$$

$$DRCE_e^D = \sum_{d \in DR, d \text{ is dedicated}} RCEW^D d, e \tag{19}$$

$$SRCEF_{e,f}^D \geq \sum_{d \in DR, d \text{ is shared}} RCEWF_{d,e,f}^D \tag{20}$$

$$SRCE_e^D \geq SRCEF_{e,f}^D \tag{21}$$

$$RCE_e^D = SRCE_e^D + DRCE_e^D \tag{22}$$

The variable $RCEW_{d,e}^D$ models the real required resilience capacity on edge e for demand d and any failure pattern (18). The variable $DRCE_e^D$ models the real required resilience capacity on the edge e for all *dedicated* demands and any failure pattern (19). The variable $SRCEF_{e,f}^D$ models the real required resilience capacity on the edge e for all *shared* demands d in case of a failure pattern f (20). The variable $SRCE_e^D$ denotes the real required resilience capacity on the edge e for all *shared* demands d and any failure pattern (21). The variable RCE_e^D models the real required resilience capacity on the edge e for all demands and failure patterns. It is the sum of all *dedicated* and *shared* resilience capacities (22).

4.7 Calculation of Capacity of the Network

The total working capacity of the network (WCN^D) is the sum of working capacities on the edges:

$$WCN^D = \sum_{e \in E} WCE_e^D \tag{23}$$

The total resilience capacity of the network is the sum of resilience capacities on the edges.

$$RCN^D = \sum_{e \in E} RCE_e^D \tag{24}$$

Finally, the total capacity used in the network can be calculated as a sum of all working and resilience capacities.

$$CN^D = WCN^D + RCN^D \tag{25}$$

5 Case Study

The models of Section 4 are formulated in ILOG Concert technology 2.0 using ILOG CPLEX 9.0 [6] with the internal barrier algorithm as MILP solver.

We dimension eight networks that vary in nodal degree and demand pattern to represent real life network structures and demand types:

- A pan-European network (COST 239 [7]) with 11 nodes and 26 ducts and four different bidirectional demand patterns yielding 40, 60, 82 and 112 demands.
- Seven random generated networks having 10 nodes and nodal degrees between 2.2 and 3.8 that are generated according to [8]. Demand structure (*A-A*) is a fully meshed homogeneous demand matrix on which 5 GBit/s are sent from each node to each other. Demand structure (*A-1*) is a centralized demand matrix on which all nodes send 5 GBit/s to one node only. Finally, demand structure (*1-A*) is a broadcast-like demand matrix on which one node sends 5 GBit/s to all other nodes in the network.

5.1 Capacity Requirement Comparison

Figures 3(a) and 3(b) depict the resulting total required capacity of the 10 node example networks. Although the chosen working paths are optimized concurrently by the solver the required working-capacity sum is almost equal for all considered case studies. However, differences are in the required amount of resilience capacity.

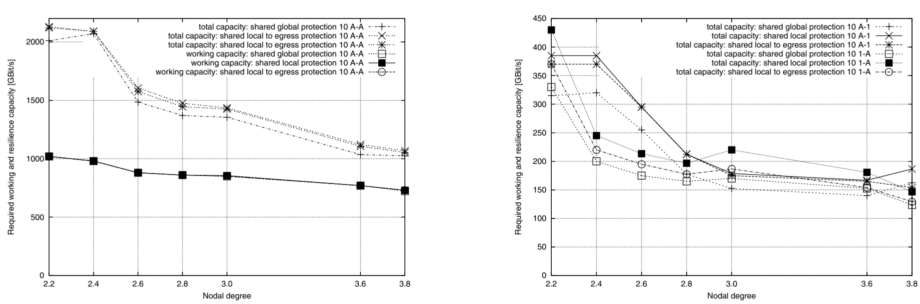


Fig. 3. Required capacity for shared global path protection, shared local link protection and shared local to egress protection of the seven random generated networks with 10 nodes, given nodal degree and demand pattern (*A-A*), (*A-1*) and (*1-A*)

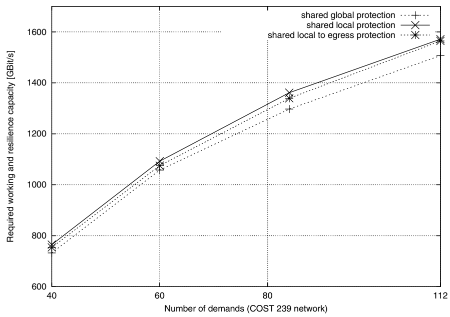


Fig. 4. Required capacity for shared global path protection, shared local link protection and shared local to egress protection of the COST239 network having 20, 30, 42, and 56 bidirectional demands

All case-study results are according to the theoretical deliberations of Section 3: Shared global path protection requires less capacity than protection with local to egress or local link protection mechanisms.

A capacity of 2010 GBit/s is sufficient to route and protect single link failures with global path protection for the network with nodal degree 2.2 and demand pattern (A-A). 110 Gbit/s in addition (2120 GBit/s) are sufficient to protect the network with shared local to egress protection and another 10 GBit/s (2130 GBit/s) are sufficient for local link protection. With this, the differences between the resilience mechanisms are quite small compared to the total sum of required capacity in our case studies.

The absolut values decrease even further with an increase in the nodal degree of the example networks. E.g. for nodal degree 3.0, demand pattern (A-A) and shared global path protection, a capacity sum of 1355 GBit/s is sufficient to protect the network while a capacity of 1024.58 GBit/s is sufficient for a nodal degree of 3.8. This capacity reduction of almost 100% (from 2010 to 1024.58 GBit/s) can be explained with the increased number and possibilities to chose working and protection paths and the ability to split backup traffic onto multiple smaller protection paths with higher sharing characteristics. The network structure (and with it the chosen working and protection paths) are transformed from an almost pure ring-like-structure with around 100% redundancy (nodal degree 2.2) to a mesh network (nodal degree of 3.8).

Figure 4 shows the optimization results of the COST239 network. Similar to the results of the random generated networks the required capacity for shared global path protection is smaller than the amount for local to egress protection that is itself smaller than the amount for local protection.

The relative differences between the three protection mechanisms, however, is rather small. For nodal degree of 2.2 of the random network the difference is around 6% only and stays within 1% to 9% for all other nodal degrees. An overview about the additional relative required capacity for shared local link and shared local to egress protection can be seen in Figure 5.

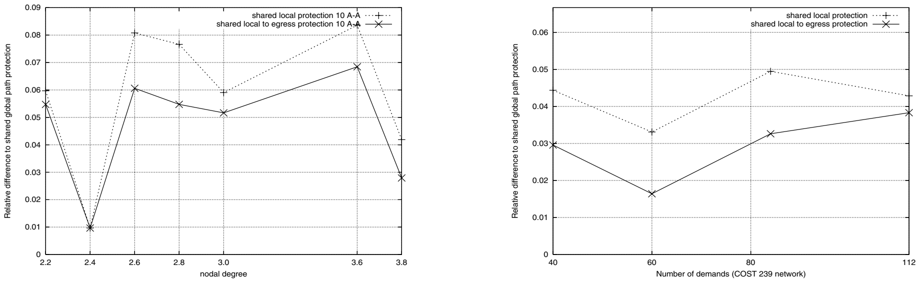


Fig. 5. Relative capacity requirement difference compared to shared global path protection for the random generated networks with different nodal degrees and demand pattern (A-A) and the COST239 network with different demands

6 Conclusion

We have investigated the capacity requirements of three widely used protection mechanisms: Shared global path protection, shared local link protection and shared local to egress protection. We have analyzed their protection path behavior for a protection of single link failures and have categorized their capacity requirements. To strengthen our theoretical deliberations we presented an optimization approach based on linear programming to be able to calculate optimal configurations and applied the optimization on existing and randomly generated networks.

The key findings are that global path protection mechanisms require less capacity compared to local to egress or local link protection mechanisms. The difference in the total required capacity, however, is quite small and in the order of some percent for the example networks. Considering the granularity of deployed hardware (interface cards and trunks) the differences and resulting cost-savings diminish even further [4].

Thus, capacity requirements seem to be no tie-breaker when deciding which shared resilience mechanism should be used in future networks. Other characteristics like the protection speed, manageability and complexity should be taken into account.

Since local protection mechanisms are able to detour traffic in the range of 50-100 ms [9, 10, 11] they may be advantageous compared to the rather slow (100s of ms) protection speeds of shared global path protection mechanisms.

A further interesting open issue remains, however: Does the amount of sharing of ressources differ between the resilience mechanisms? If this is the case, it might have a drastic impact on multiple failure survivability.

References

1. J. Moy. *OSPF Version 2*, Request For Comments 2328, IETF, April 1998.
2. *Intermediate System to Intermediate System Intra-Domain Routeing Exchange Protocol for use in Conjunction with the Protocol for Providing the Connectionless-mode Network Service (ISO 8473)* ISO DP 10589, February 1990,

3. E. Rosen, A. Viswanathan, and R. Callon. *Multiprotocol label switching architecture*, Request For Comments 3031, IETF, January 2001.
4. S. Orlowski and R. Wessäly, *Comparing restoration concepts using optimal network configurations with integrated hardware and routing decisions*, Proceedings of the Fourth International Workshop on the Design of Reliable Communication Networks (DRCN), Banff, CA, 2003
5. Dominic A. Schupke, Claus G. Gruber, and Achim Autenrieth, "Optimal configuration of p-cycles in WDM networks," *IEEE ICC 2002, New York*, 2002.
6. ILOG *CPLEX Version 9.0, Concert Version 2.0, http://www.ilog.com*
7. P. Batchelor et al *Ultra high capacity optical transmission networks*, Final report of action COST 239, 1999
8. H.F. Salama *Multicast Routing for Real-Time Communication On High Speed Networks* Dissertation, Raleigh, NC, 1996.
9. *A comparison of MPLS with KING Hammock Routing*, Internal report of the project KING, 2003
10. A. Autenrieth *Differentiated Resilience in IP based Multilayer Transport Networks.* Dr.-Ing. thesis, Munich University of Technology, Munich, Germany 2003.
11. W.D. Grover *Mesh-based Survivable Transport Networks: Options and Strategies for Optical, MPLS, SONET and ATM Networking* Prentice Hall PTR. 2003.

Quality of Service Solutions in Satellite Communication

Mathieu Gineste[1,2] and Patrick Sénac[1,3]

[1] ENSICA, DMI 1 Place Emile Blouin 31056, Toulouse Cedex, France
{mgineste, senac}@ensica.fr
[2] LIP6, 8 rue du Capitaine Scott, 75015 Paris, France
[3] LAAS/CNRS, 7 avenue du Colonel Roche, 31077 Toulouse cedex 04, France

Abstract. This paper presents architectural solutions and results from an ESA (European Space Agency) study intending to mitigate the limitations of the current TCP/IP protocol stack in a satellite environment. Focus is done on the Quality of Service solutions to improve the fair sharing of the return link (based on the DVB-RCS standard) and to optimize this link utilization. A QoS-based approach (XQoS) is presented and adapted to the satellite context to map the user and application requirements toward services and resources available at lower layers. Experimental results of the QoS oriented architecture, done on a satellite emulation platform, are presented to demonstrate its effects on the communication.

1 Introduction

In the last years, TCP/IP family protocols have been deployed in almost all the network communicating entities, and as a consequence, most applications have been developed on top of this protocol stack. These protocols were originally designed to fit the networks and applications used at that time, namely the wired networks, characterized by low delay and losses due to congestion and applications characterized by elastic time constraints (e.g. file transfer). Recently, multimedia applications have been evolving and taking an important role in the multi-domain information exchange on the Internet. This applications have new requirements in terms of Quality of Service (e. g. real-time constraints, large bandwidth, low jitter…). The wired network protocols are not deterministically responding to their QoS requirements related to order, reliability, bandwidth, time and synchronization constraints. Yet, the decrease of equipment costs has permitted to increase the networks' capacity. Thus, in order to reduce congestions and bottlenecks, over-provisioning of resources is often the solution used. In the same time, new technologies are offering broadband access to end users (such as DSL and cable).

However, this wide-bandwidth wired network cannot be implemented in all areas. Therefore, wireless technologies such as satellite with its wide broadcasting capacities appear as a fundamental component of a definitively heterogeneous Internet. But, two main aspects differentiate the satellite link from a traditional wired network:

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 144–152, 2005.

– First, the characteristics of the transmission through the satellite link, in terms of delay and loss models: the delay is much longer than in a wired network and losses are not due generally to congestion but to link errors.
– Then, the characteristics of the return link of the satellite: This link has generally a low amount of available bandwidth due to its cost and so appears as a scarce resource that has to be managed optimally.

Consequently, the current TCP/IP protocol stack will not be able to fit well in satellite communication due to the new characteristics introduced by satellite links.

In addition, the Best Effort behavior of the IP protocol will not permit to share properly and efficiently resources of the return link's bottleneck and in congestion situations, this will have dramatic effects on real-time applications. In this case, over-provisioning would not be possible due to the high cost of these satellite resources.

The current inefficiency of the TCP/IP stack for satellite links in the Internet have lead ESA (European Space Agency) to promote research in this area and to support the TRANSAT (Transport protocol and Resource mANagement for mobile SAtellite neTworks*)* project leaded by Alcatel-Space. The goal of this project, of which some solutions are described in this paper, is to adapt the current TCP/IP protocol stack to the satellite link constraints and to optimize the satellite resources usage.

Next sections are organized as follow. A general presentation of the TRANSAT project and its architecture is done. In section 3, a QoS specification language (XQoS), taking into account application requirements and services availability, is described. The integration of this language in the TRANSAT architecture and the QoS solutions are presented as well in this section. Finally, experimental results evaluating this solutions are shown and conclusions are presented.

2 TRANSAT Project General Presentation

TRANSAT is an ESA (European Space Agency) study leaded by ASPI (Alcatel Space Industry) and involving the University of Helsinki (Finland), ENSICA (France), INRIA (France), ISD (France). The study focuses on the Internet broadband

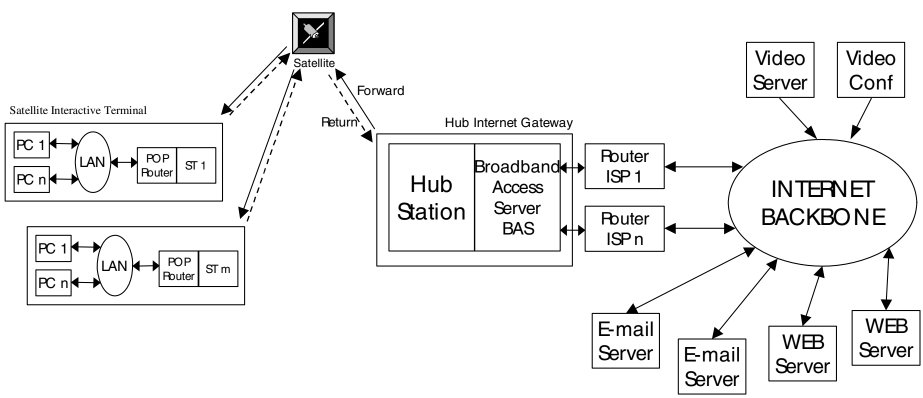


Fig. 1. TRANSAT System Architecture

access via a geostationary bent pipe satellite. The architecture (Figure 1) is based on the DVB-RCS standard (on the return link) in order to provide broadband internet access to end users anywhere in the satellite coverage. The purpose of TRANSAT is to evaluate some improvement methods to offer DVB-RCS users a similar level of performances and functions achieved in pure terrestrial broadband access. The context of the study as we can see on this first figure is to improve performance of a LAN network accessing the Internet via a satellite link and so to optimize the resource usage of this link. We have particularly focused, in the study, on the return link of the satellite which has generally a low amount of bandwidth available in its contract of service (i.e. the Service Level Agreement passed between the Satellite Terminal and the Hub Station) and, due to the cost of this bandwidth, needs to be used efficiently.

The study has mainly focused on TCP enhancements knowing satellite link constraints and Quality of Service of the protocol stack.

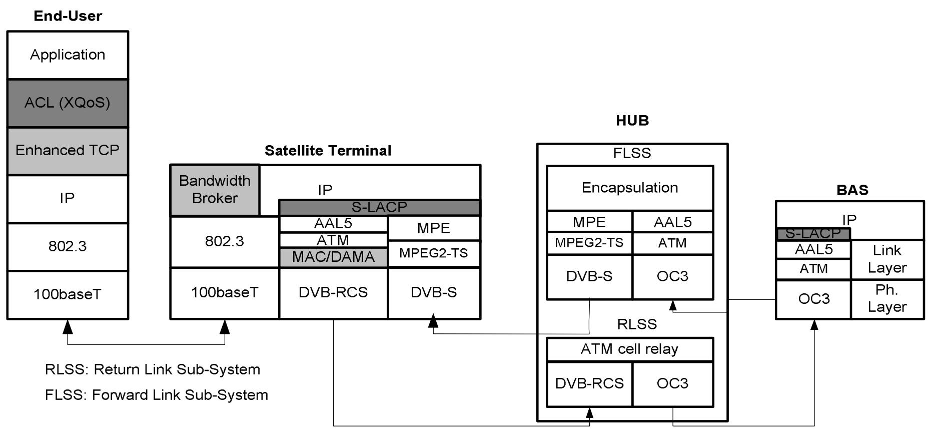


Fig. 2. TRANSAT Protocol Stack

The global aim is to propose the definition and design of a QoS-oriented architecture adapted to the satellite link. Figure 2 presents TRANSAT protocol stack architecture: Three layers of the protocol stack have been modified: TCP on end-systems, IP on the Satellite Terminal (with a Bandwidth Broker addition) and MAC/DAMA on the ST (Satellite Terminal). Moreover, two layers have been added compared to the standard stack: ACL the Abstract Communication Layer that is intended to process QoS oriented mechanisms and can be seen as a QoS management and control plane, S-LACP Satellite Link Aware Communication Protocol, standing between the IP and the MAC/DAMA layers on the ST and on the BAS. The role of these new and enhanced layers will be detailed and in particular, the QoS aspects of this architecture namely ACL and the IP layer as well as MAC/DAMA on the Satellite Terminal.

2.1 TCP and S-LACP

TCP has well-known potential limitations when used in a satellite environment ([1] [2]) due to the fact it has been originally designed with terrestrial network constraints in mind not taking into account delay and losses characteristics introduced by the satellite. Consequently, while in a wired network losses are mainly due to congestion, implying stringent bandwidth reduction applied on TCP flows by the congestion control mechanisms, it's different in a satellite environment where losses often comes from packets' corruption and in this case the sending rate does not need to be decreased. In addition to that, the large network delay, introduced by the satellite link, degrades badly the application performance when retransmissions, following upon losses, are done. The goal of the TRANSAT study is to mitigate those effects while keeping the end-to-end aspect of TCP that can be essential for the good functioning of some applications. To achieve this goal, two concurrent enhancements have been studied:

1. To use possible TCP enhancements already available in standard tracks
2. To design a specific protocol (S-LACP: Satellite Link Aware Communication Protocol) working below IP level on the ST and on the BAS in order to correct link errors as much as possible.

First, concerning the **TCP enhancements**, an optimal customization of the protocol was targeted integrating the more adapted mechanisms in satellite communication and tuned thanks to a test campaign. The retained features were proposed in IETF standards but not used in Satellite TCP (RFC 2488) such as the TCP initial window size increase, use of Conservative SACK algorithm, the Limited Transmit, the Ack-every-Segment during Slow Start, the Control Block Interdependence (CBI), the Fast Retransmit Time Out (F-RTO) and the Explicit Congestion Notification (ECN) with RED. The test campaign of this enhanced TCP showed major performance improvements compared to the standard TCP implementation and even compared to the Satellite TCP version. A full description results is detailed in [3].

Another major problem with TCP is its inability to distinguish between packet losses due to link errors and packet losses due to congestion. In both cases, TCP reduces its transmission rate, while it is only necessary in the second case. We proposed satellite link enhancements with the addition of **S-LACP** protocol combined with TCP enhancements detailed before. The S-LACP role is to reduce the error rate perceived by the transport layer, using a combination of both FEC (Forward Error Connection) and ARQ (Automatic Repeat reQuest) modes, in order to minimize additional delay due to retransmissions. In the TCP performance tests, we noticed that S-LACP approach turned out to be beneficial, in particular on a noisy satellite link [4]. S-LACP with only ARQ mechanisms helps recovering most errors in the low, medium and high error conditions. FEC coding, applied on retransmitted ARQ packets, gains significant improvement in TCP performance on very noisy satellite links. In other cases, the overhead introduced by FEC coding compensates for its benefits.

All TCP customizations and the addition of S-LACP intend to improve the performance of the transport layer over satellite links when applications are using this protocol for their transport. However, TCP is not used by every applications, moreover, Quality of Service will not be guaranteed by these new mechanisms. There is still a need for a management of the Quality of Service that could guarantee an optimal mapping from the application requirements toward the underlying layers and the respect of this QoS but without important modifications of the standard protocol stack and ideally in a transparent way to the applications. This is the intent of the Abstract Communication Layer (ACL) presented in the following section.

2.2 ACL (Abstract Communication Layer), IP and MAC/DAMA on the ST

The Abstract Communication Layer is a new layer intended to insure the mapping, the management and the control of the Quality of Service in order to guarantee an optimal use of the satellite resource and in the same time the respect of the applications QoS needs and priorities. This Layer ACL can be seen as a control plane of the Quality of Service that will take advantage and manage all the services available in the communication system. It is represented on Figure 2 between the Application and the Transport Layers on the end-system, because this is a privileged location to gather QoS requirements of the application and possibly of the end-user.

A major challenge in a satellite environment is to share properly and efficiently access to satellite bandwidth, particularly on return links. The goal is to have a relevant mapping from the application toward the MAC/DAMA layer of the satellite which is the only solution to both respect the application QoS constraints and optimize the resources utilization. At the Network layer the DiffServ approach [5] is the approach chosen to differentiate the applications' flows from one another in respect with their priority using several Classes of Service. This approach also offers a scalability which could permit to extend the Diffserv domain in an easier way than with the IntServ approach. Consequently, we have added DiffServ facilities at the IP layer of the ST (which is the edge router). We have used a bottom up approach to define this QoS mapping, in order to ensure its feasibility: first, we proposed a pertinent mapping from the DiffServ Classes of Service toward the DVB-RCS capacity requests offered by the Satellite Terminal (ST); and then, ACL would help choosing the appropriate Class of Service knowing the QoS needs of the applications and the availability of resources. To maintain this resources availability and to configure dynamically the DiffServ node, a Bandwidth Broker is implemented on the ST.

This considerations lead us to take a more general question into account, not specific to the satellite access but that is critical in this case: this is how we could retrieve, present and take into account the application requirements in term of Quality of Service and then map these needs optimally toward the lower layers knowing the available services and resources offered by the communication service. In the following chapter we describe first our solution to this problem in a general framework and how to apply it specifically to the satellite access case.

3 XQoS Presentation and TRANSAT QoS Architecture

3.1 QoS Considerations and XQoS Language

Many applications, and particularly multimedia applications, have now specific QoS (Quality of Service) constraints that need to be respected for them to work properly.

Due to their expansion, these multimedia applications were part of our target in the TRANSAT project. QoS requirements are not deterministically respected by wired networks, moreover the new characteristics and constraints of the satellite link make the respect of these requirements even more challenging. TCP and UDP standard transport protocols are offering an everything-or-nothing service and definitely not a service with a QoS description in accordance with new applications' needs. Consequently, adding the standard transport protocols (.i.e. TCP and UDP) to the best effort network service as well as the best effort access to the satellite link, does not permit to take into account the QoS required by applications. Although the QoS parameters required by the multimedia applications are well known and some form of guaranteed QoS is available at transport and network layers (such as IP QoS [5],[6] and alternative transport protocols [7],[8]), there is no standard QoS specification enabling to deploy the underlying mechanisms in accordance with the application QoS needs. Therefore, there is a need for a standard QoS specification, and an associated framework, that may be employed to map the application requirements to the specific transport, network and data link services available in the communication systems [9], in other words a QoS control plane. We have proposed a QoS specification language to describe applications' needs and available network services called XQoS (XML-based QoS Specification Language) [10]. An associated approach permits to derive these requirements into a configuration of network, transport and system services and mechanisms.

XQoS is an XML-based language, based on TSPN (Time Stream Petri Network) formal model. Four types of description can be defined to express: the application QoS specifications (parameters specific to various flows composing a given session), the media type definitions (parameters generic to a given media or codec), the communication services (parameters characterizing available services and mechanisms), the resources (parameters to describe resources on a given device).

A model is also presented to map the high level descriptions of Quality of Service (as expressed by users or standard applications in the session negotiation using for instance SDP) toward precise QoS parameters that could be used to configure network services and mechanisms (such as bandwidth of flows participating in the session, delay and jitter required, synchronization constraints between flows, admitted loss rates…). Then a mapping and policing methodology based on PCIM [11] permits to find an optimal mapping solution between the application requirements and the available communication services and resources. The approach consists in discarding the non viable solutions and to propose ordered combinations of available services.

3.2 TRANSAT QoS Architecture Integrating XQoS Approach

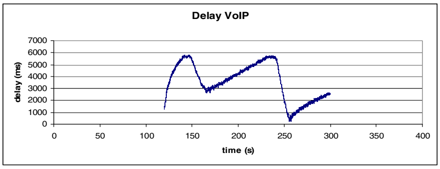
The main goal of the Quality of Service oriented architecture introduced in TRANSAT is to use optimally the satellite return-link resources that are costly and generally not very abundant. Consequently, the role of ACL and the Bandwidth Broker is to share properly and efficiently access to the available resources of the satellite depending on application needs, taking into account their QoS constraints and priorities as well as the user preferences. In addition, the satellite access specificities need to be taken into account: the DVB-RCS (Digital Video Broadcasting-Return Channel System via Satellite) standard, used for the communication on the return link of the satellite, proposes four main classes of traffic assignment to access the satellite, characterized by guaranteed bandwidth or not and various jitters and delays. Thus, a mapping needs to be done from the application toward the Physical Layer and resources. To ensure the feasibility of the mapping, we had a bottom up approach, from the access mechanism of the satellite toward the applications QoS needs. A DiffServ node is used at the IP layer of the ST to differentiate the flows in order to set priorities between them and control their bandwidth use. A QoS controller (the Bandwidth Broker) is implemented on the ST to do the admission control of the flows and to redirect them toward the corresponding IP and DVB-RCS classes by managing and configuring dynamically the DiffServ node. So, we first map DiffServ classes toward the DVB-RCS access mechanisms depending on their characteristics and the DiffServ classes properties: A high priority class of the DVB-RCS access scheme is fully guaranteed, thus EF (Expedited Forwarding), corresponding to high priority traffic not admitting long delay or variable jitter, is associated with this traffic class. A second type of traffic in DVB-RCS is guaranteed up to a maximum but introduces more delay and jitter. This traffic is associated to the AF (Assured Forwarding) IP type of traffic which permits to assure a high delivery of packets up to the max bandwidth and marks packets with a higher drop precedence if the maximum is reached. In congestion situations, these packets will be dropped first. Then, all the DVB-RCS non guaranteed types of traffic are associated to the Best Effort (BE) IP type of traffic with a lower priority than AF traffic, having a lower priority than EF traffic. No per-flow admission control is done for BE traffic type, but, its global rate is limited to the remaining bandwidth not used by high priority traffic.

The Service proposed by the QoS Controller, is described using XQoS language and passed to the ACL every time it requests it, when a session starts. The description includes the guaranteed bandwidth, the delay, the jitter and the reliability proposed by each DiffServ class of traffic at the time it is requested. The ACL after gathering QoS information about the session and user preferences using the XQoS framework will map these application requirements toward the optimal service proposed by the QoS Controller (following the XQoS policing methodology) and request bandwidth reservation. The QoS Controller is doing a per flow admission control, applies packets marking, and updates the DiffServ node to make sure the user and the application remain in profile. This approach assures an optimal usage of the satellite resources taking into account applications and users requirements in terms of QoS.

4 Experimental Results

The graphs on Figure 3, represent the measured delay of a Voice over IP (VoIP) application on the return link of a satellite emulation platform, running along with concurrent flows and having together a sending rate above the rate limit of the link.

On the first graph, where no QoS is implemented (corresponding to current satellite architectures), the delay perceived by the VoIP application is dependent on the concurrent flows and in particular TCP flows increasing progressively their rate and then reducing it upon packet losses. The delay reaches 6 seconds which is not compatible with the interactivity and time constraints of a VoIP application.



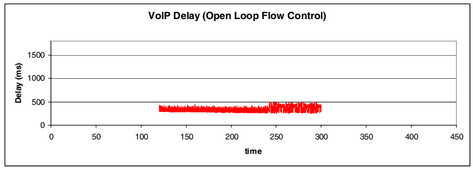


Fig. 3. VoIP one-way Delay with no QoS and with the QoS oriented architecture

On the second graph, the QoS architecture case, admission control is done for the VoIP flow and its corresponding rate is reserved in high priority type of traffic at the IP level of the Satellite Terminal and so on the return link access of the satellite. The delay is constant, around 300 ms, not disturbed by concurrent traffic and acceptable for voice applications.

5 Conclusions

Theory and experimental results, demonstrate that the current behavior of the standard TCP/IP protocol stack combined with the characteristics of satellite links do not permit to assure a fair bandwidth allocation and more generally does not provide the specific Quality of Service required by each flow composing applications. To overcome these issues, we have developed a Quality of Service oriented architecture, based on the XQoS specification language, capable to take into account the QoS requirements of the applications and derive them toward the underlying layers to provide the best service as possible while keeping the high utilization of the costly satellite bandwidth. As part of this architecture, a network service has been proposed, including admission control mechanisms based on the available bandwidth of the satellite link and flow differentiation based on DiffServ principles. To improve the TCP performance over satellite, an enhanced TCP layer has been proposed as well as a new link layer minimizing packet losses due to link corruption on the satellite link.

References

1. M. Allman, et al, Ongoing TCP Research Related to Satellites, RFC 2760, February 2000
2. M. Allman, et al, Enhancing TCP Over Satellite Channels using Standard Mechanisms, RFC 2488, January 1999
3. D. Astuti et al, TCP Performance Simulation Study in TRANSAT Satellite Architecture, 03
4. M. Kojo et al, Improving TCP Performance over Wireless WANs using TCP/IP-Friendly Link Layer, ICETE 2004
5. S. Blake, et al, An Architecture for Differentiated Services, Oct.1998. RFC 2475.
6. R. Braden, D. Clark, S. Shenker. Integrated Services in the Internet Architecture: an Overview, June 1994. RFC 1633.
7. Eddie Kohler, et al, Datagram Congestion Control Protocol (DCCP), July 2004.
8. R. Stewart, et al, Stream Control Transmission Protocol. October 2000. RFC 2960.

9. Huard J.F. and Lazar A.A., On QOS Mapping in Multimedia Networks, 21th IEEE Annual International COMPSAC '97, Aug. 13-15, 1997.
10. E. Exposito , M. Gineste , et al, XQOS: XML-based QoS Specification Language The 9th International Multi-Media Modeling Conference January 2003, Taiwan.
11. B. Moore et al, Policy Core Information Model, RFC 3060.

QoS-Oriented Packet Scheduling Schemes for Multimedia Traffics in OFDMA Systems

Seokjoo Shin[1], Seungjae Bahng[2], Insoo Koo[3], and Kiseon Kim[3]

[1] Dept. of Internet Software Engineering, Chosun University, Korea
[2] Dept. of Electrical Engineering, University of Hawaii, USA
[3] Dept. of Infor. and Comm., GIST, Korea

Abstract. In this paper, we propose packet scheduling disciplines for multimedia traffics in the contexts of OFDMA systems. For real time traffics, packet loss fair scheduling (PLFS), in which the packet loss of each user from different real time traffics is fairly distributed according to the QoS requirements is applied. A simple priority order queue mechanism is applied for non-real time traffics. In addition, to compensate fast and slow channel variation we employ link adaptation technique such as AMC. From the simulation results, our proposed packet scheduling scheme can support QoS differentiations with guaranteeing short-term fairness as well as long-term fairness for various multimedia traffics.

1 Introduction

Scheduling algorithms provide mechanisms for bandwidth allocation and multiplexing at the packet level. Many scheduling algorithms, capable of providing certain guaranteed QoS, have been studied for wireless networks. In the earliest-due-date-first (EDD), each packet from a periodic traffic stream such as real time services is assigned a deadline and the packets are sent in order of increasing deadlines [1]. The principle of EDD is based on the priority-order-queue-mechanism. The real time (RT) traffics such as voice and video streaming are very delay sensitive, but can stand a certain level of packet loss. The service-oriented fair packet loss sharing (FPLS) algorithm is introduced in [2], where TD/CDMA system is considered. The basic idea of the FPLS is that the packet loss of each user is controlled according to the QoS requirements and the traffic characteristics of all the mobile users sharing the same frequency spectrum in the cell. From the results, FPLS provides a higher spectral efficiency than GPS algorithm proposed in [3]. In [2], however, the authors did not consider the link adaption techniques such as adaptive modulation and coding (AMC) in a cellular environment. In order to evaluate the performance of a packet scheduling algorithm deployed in wireless environment more reliably, the wireless channel environment should be considered.

In this paper, we propose a QoS-guaranteed packet scheduling schemes for multimedia traffics. Packet loss fair scheduling (PLFS), in which the packet loss of each user from different RT traffics is fairly distributed according to the tolerable

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 153–160, 2005.

packet loss requirements of all the user equipments (UEs) sharing the same frequency spectrum in a cell is applied for RT traffics. On the contrary, a simple scheme based on priority ordering of each user is applied for NRT traffics. Since RT and NRT traffic has quite different QoS characteristics, it could be more benefit to make distinct decision rule for each. Therefore, we suggest a new frame structure with separating these two traffics and propose two scheduling scheme, respectively.

The OFDM/FDMA (OFDMA: Orthogonal Frequency Division Multiple Access) system is considered in this paper, since the most suitable modulation choice for beyond 3G mobile communication systems seems to be OFDM due to its high data rate transmission capability with sufficient robustness to radio channel impairments. In addition, the link adaptation techniques have been widely studied to overcome low wireless channel efficiency. Adaptive modulation and coding (AMC) is one of the compromising techniques providing flexibility to choose an appropriate modulation and coding scheme (MCS) for the channel conditions based on either UE measurement reports or network determined. We adapt AMC technique to the proposed packet scheduling algorithm according to the received signal-to-interference ratio (SIR) of each UE.

2 System Model

We consider an OFDMA cellular system with packetized transmission. One central base station (BS) and multiple distributed users are set to be a cell. The basic frame structure of the considered OFDMA system is shown in Fig. 1 in the context of downlink in a cellular packet network, where the time axis is divided into a finite number of slots within a frame and the frequency axis is segmented into subchannels that consist of multiple subcarriers. In Fig. 1, the basic resource space of packet transmission is defined as basic unit (BU) which corresponds to a slot in time and a subchannel in frequency. Therefore, there are $N_{slot} * N_{sub}$ BUs in a frame, where N_{slot} is the number of slots in a frame and N_{sub} is the number of subchannels in frequency. In this system, maximum N_{sub} users can transmit their packets simultaneously in each slot without intra cell interference.

BU_{ij}, where $0 \leq i \leq N_{slot} - 1$ and $0 \leq j \leq N_{sub} - 1$, delivers a certain amount of information bits defined as $C(BU_{ij})$. $C(BU_{ij})$ is highly dependent on the channel condition of its assigned UE. The instantaneous system capacity, $R_c(t)$, is represented as follow:

$$R_c(t) = F_s \Sigma_{i=0}^{N_{slot}-1} \Sigma_{j=0}^{N_{sub}-1} C(BU_{ij}) \qquad bit/s \tag{1}$$

where F_s is the number of frames in a second. The system capacity, $R_c(t)$, changes in time randomly.

In addition, one frame duration is divided into three parts: T_{RT} for RT traffics, T_m for marginal region (in this region both RT users and NRT users can transmit their packets), and T_{NRT} for NRT traffics. The boundaries are movable dependent on the previous RT traffic usage. Since some margin is required in the current channel usage of RT traffics, we simply apply the movable boundary as follow

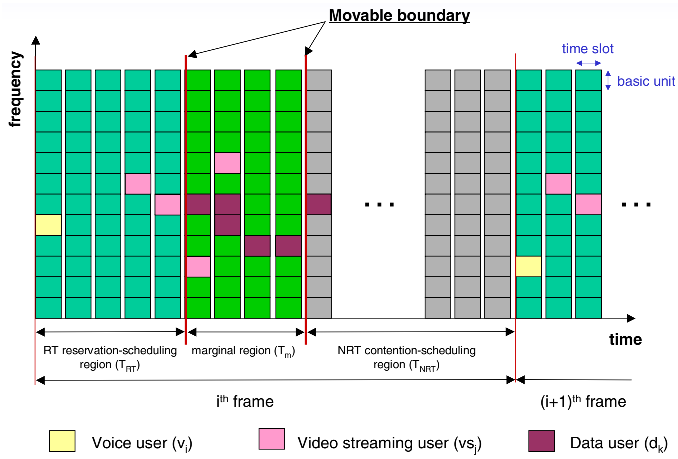


Fig. 1. The proposed frame structure of OFDMA

$$boundaries = \begin{cases} \sum_{i=1}^{N_{RT}} K_{BU}^{i} \mp \alpha, & if \sum_{i=1}^{N_{RT}} K_{BU}^{i} > \alpha \\ \sum_{i=1}^{N_{RT}} K_{BU}^{i} + \alpha, & else > N_{av} \end{cases}, \tag{2}$$

where K_{BU}^{i} is $|B_i/\bar{k}_i|$, in which $\bar{k}_i$ is the average number of transmitted MAC PDUs in a BU for RT user i and B_i is the number of MAC PDUs in a buffer of RT user i.

The packet scheduler schedules up to N_{sub} users among all of active users in every slot instant. Each UE has its own buffer to queue the packets for transmission. The packet size of all buffers is assumed to be identified. At i_{th} time slot, the packet scheduler selects the N_{sub} highest priority buffers after calculating the priority of each buffer based on the uplink feedback information and buffer management information. The scheduling discipline is described more details in the next section.

Under the wireless environment, we adapt AMC technique to the proposed packet scheduling algorithm with assigning K MCS levels depending on UE measurement reports. In the frequency selective fading environments, UE measurement reports of different subchannels have different values. The user k receives SNR_k values, $SNR_k^0, \cdots, SNR_k^{N_{sub}-1}$, from predetermined pilot signals corresponding to each subchannel. SNR is measured as the ratio of pilot signal power to background noise when we assume that there is no other-cell interference at all. More specifically, the SNR of the n^{th} subchannel allocated for the k^{th} user can be represented as

$$SNR_k^n = \frac{P_p h_{k,n}^2}{N_0 \frac{B}{N_{sub}}} \tag{3}$$

where $h_{k,n}$ is random variable representing fading the k^{th} users' and n^{th} subchannel. P_p is the transmitted power of the pilot signal. N_0 is the noise power

Table 1. Transmission mode with convolutionally-coded modulation

Index	SNR_{req}	Packet/BU	Modulation	Coding rate
AMC_1	1.5 (dB)	1	BPSK	1/2
AMC_2	4.0 (dB)	2	QPSK	1/2
AMC_3	7.0 (dB)	3	QPSK	3/4
AMC_4	10.5 (dB)	4	16QAM	1/2
AMC_5	13.5 (dB)	6	16QAM	3/4
AMC_6	18.5 (dB)	9	64QAM	3/4

spectral density and B is the total bandwidth of the system. The channel gain, $h_{k,n}^2$, of subchannel n of user k is given by:

$$h_{k,n}^2 = |\alpha_{k,n}|^2 \cdot PL_k \tag{4}$$

Here PL_k is the path loss for the user k and defined by:

$$PL_k = PL(d_0) + 10\beta \log(\frac{d_k}{d_0}) + X_\sigma \tag{5}$$

where $\alpha_{k,n}$ is short scale fading for user k and subchannel n. d_0 and d_k are the reference distance and distance from BS to user k, respectively. β is path loss component and X_σ represents Gaussian random variable for shadowing with standard deviation σ.

To reduce the signaling overhead, UE selects $N(< N_{sub})$ SNR_ks for the feedback of channel quality information (CQI). After receiving UE's CQI index, BS allocates the appropriate BUs to the UE if the user is selected for the scheduling in the current slot.

The MCS level is classified by required SNR strength, SNR_{req}, and maps to the number of packets in a BU. The mapping between MCS level and the number of packets is shown in Table 1 when we assume that all subchannel are allocated with equal power i.e., $1W$. From the channel condition of user k, $\bar{A}_k$ is defined as moving average of the number of transmittable packets in a BU from the previous trials with window size WS.

3 Proposed Scheduling Algorithms

3.1 For Real Time Traffics

Among the diverse objectives for fairness to incorporate multimedia services, we focus on the fair QoS-guarantee scheduling rules for RT traffics. The proposed PLFS algorithm is concerned with satisfying the different required QoS evenly for the RT traffics. A fairness guarantee for RT traffics is assumed to be achieved when the current packet loss rate (PLR) is distributed proportionally-equal for all users at any time instant. In other words, our proposed scheme ensures short-term fairness as well as long-term fairness simultaneously.

PLR occurring in real-time traffics is defined as the sum of packet error rate (PER) resulting from the channel impairments and packet dropping rate (PDR) calculated from packets exceeding the required maximum delay, D_{max}, in each traffic. PLR should be less than a certain determined threshold, $PLR_{req,i}$, for user i, that is,

$$PLR_i(t) = PER_i(t) + PDR_i(t) \leq PLR_{req,i} \tag{6}$$

Priority order queue mechanism is applied for supporting PLR of each active user fairly. PLFS rule schedules the highest priority user among all users. After that, the scheduler selects the next highest priority user continuously until all subchannels in a slot are occupied. The scheduler updates priority of each active user in every time slot before scheduling. For a predetermined set of parameters related to the required QoS of each user, the rule is given by,

$$j = max\left[\left(\frac{A_k(t)}{\bar{A}_k}\right) \cdot \left(\frac{PLR_i(t)}{PLR_{req,i} \cdot D_{max,i}}\right)\right] \tag{7}$$

where $A_k(t)$ is the state of the channel in terms of the MCS level of user k at time t. The key feature of this algorithm is that a scheduling decision depends on both current channel conditions and current packet loss of different users. The term, $\frac{A_k(t)}{\bar{A}_k}$ becomes the proportionally fair queuing presented in [4], while $\frac{PLR_i(t)}{PLR_{req,i} \cdot D_{max,i}}$ renders the scheduling rule be the packet loss fair queuing and can provide QoS differentiations between different users.

3.2 For Non-real Time Traffics

Since the NRT traffics are not delay sensitive, we assume that the NRT packets are scheduled only if when there is available slots after scheduling RT traffics as shown in Fig. 1. Between NRT users, the scheduler schedules user according to the priority order queue mechanism based on queue length and waiting time of each user. For giving fair transmission opportunity of each user, the rule is given by,

$$j = max\left[b_i(t) \cdot \frac{A_k(t)}{\bar{A}_k} \cdot W_i(t)\right] \tag{8}$$

where $b_i(t)$ is queue length of user i and $W_t(t)$ is the head-of-the-line packet delay of user i at time t.

4 Simulation Environments

For the system level simulation, we consider two types of RT traffics such as voice and video streaming and a NRT traffic such as WWW. A voice source creates a pattern of talkspurts and gaps that are assumed to have exponentially distributed duration. These are assumed to be statistically independent of each other. The mean duration of the talkspurts and gaps are 1sec and 1.35sec, respectively. The source data rate of voice traffic is assumed to be $16kbps$. A video

Table 2. *Simulation parameters*

Parameters	Value
Number of subcarriers in OFDM	1536
Number of subchannels	12
Frame length (ms)	20
Slots per frame	20
Maximum packet loss rate (PLR_{req})	voice= 10^{-2}, video streaming= 10^{-3}
Packet size ($byte$)	44 (including $4bytes$ header)
Maximum packet delay (ms)	variable
Cell radius (km)	1
User distribution	Uniform
BS transmission power	$12W$
Path loss model	α=4, σ (dB)=8

streaming is modelled as VBR (variable bit rate) characterized by Pareto distribution. The modelling is composed of continuous video-frames, where each video-frame is divided into the fixed number of video-packets [6]. Moreover, the size of a video-packet is determined by Pareto distribution. The parameters of video streaming model is described in [6] where the generation rate is $32kbps$. Note that a variable length video-packet is segmented into the fixed-length MAC PDUs before being stored the scheduler buffer.

On the other hand, WWW is modelled as ABR (available bit rate) based on Pareto distribution. We follow the parameters basically proposed by ETSI. In this model, the session is defined, in which the average number of packet calls per session is 5. The size of packet within a packet call is characterized by Pareto distribution where mean value is $480bytes$ and the maximum packet size is $11kbytes$. A generated WWW packet from the aformentioned model is segmented into the fixed-length MAC PDUs in advance the scheduling decision instant.

In this paper, we consider the radio cell with N_{user} active UEs and a centralized BS. The BS has packet scheduler in its MAC layer to obtain a high statistical multiplexing gain, where N_{user} buffers are directly connected to the packet scheduler. We assume that the fixed length packets (or MAC PDUs) are stored into the scheduler buffers.

In a wireless cellular network, channel state varies randomly in time on both slow and fast scale. As slow channel variation depends on mostly user location and interference level, the normalized power from BS is adapted to the diverse MCS level in AMC technique according to the received SNR_k for a user k.

The set of the simulation parameters is listed in Table 2.

5 Simulation Results

Packet loss rate (PLR) and transmission delay are considered as the performance measures for the RT and NRT traffics, respectively.

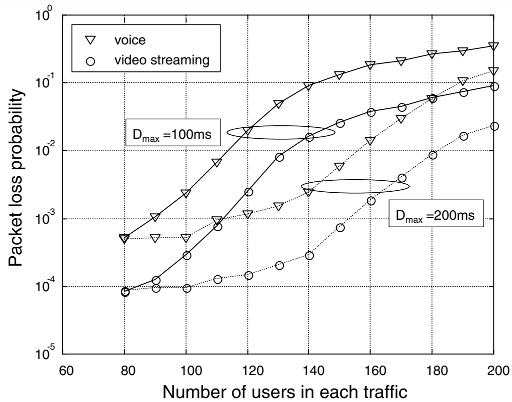


Fig. 2. PLR vs. the number of concurrent voice and video users when D_{max} ==100 and 200ms, respectively

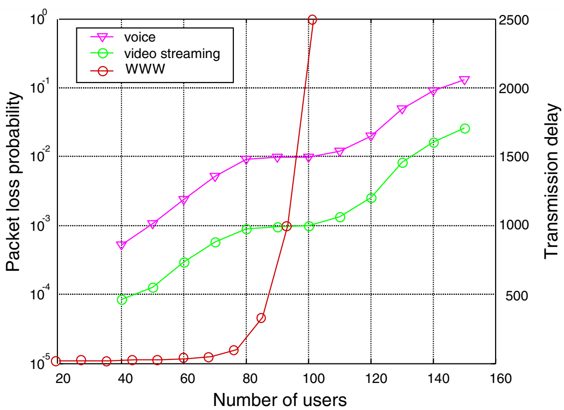


Fig. 3. PLR and delay performances of the RT and NRT traffics when D_{max}=100

Through a computer simulation, PER is ignored from the assumption that the channel condition is well estimated and predicted. Fig. 2 and Fig. 3 show that the performances of the proposed scheme when the system supports equal number of users in both RT traffics (N_{voice}=N_{video}) simultaneously.

When we assume that there are only RT traffics in system, Fig. 2 shows the PLR performace of each RT traffic. From the figure, the proposed PLFS algorithm for RT traffics gives fair resource sharing in terms of PLR requirements of real-time traffics. PLR experienced in each traffic user at a time is maintained to be fairly distributed for all different real-time traffics. For a given PLR requirements when D_{max}=100ms as an example, the number of concurrent voice and video users are same to be 115. Note that the higher D_{max} is, the more users can be supported.

Fig. 3 shows the PLR of RT traffics and delay performance of NRT traffics from the our proposed MAC frame structure and scheduling algorithms. Note that there is flat region in graphs of RT traffics. The reason is that the boundaries are moving to the end of frame according to the increase of PLR of RT traffics. On the other hand, the delay of NRT traffic is rapidly increased from the start point of flat region due to the lack of its bandwidth.

6 Conclusions

In this paper, a new frame structure and scheduling schemes have been proposed for multimedia traffics based on OFDMA system. For RT traffics, we proposed PLFS algorithm, in which the packet loss of each user is fairly distributed according to the QoS requirements. Fair scheduling algorithm was suggested for NRT traffic. From the computer simulation, the results verify the PLFS to give fair resource sharing between real-time users. In multimedia case, RT traffics are well supported for their QoS requirements fairly between users in the expense of the quite high delay performance of NRT users.

References

1. C. Lin, J. Layland, "Scheduling algorithms for multiprogramming in a hard real-time environment," *J. ACM*, vol. 20, pp. 46–61, Jan. 1973.
2. V. Huang and W. Zhuang, "Fair packet loss sharing (FPLS) bandwidth allocation in wireless multimedia CDMA communications," Proc. Int'l Conf. 3G Wireless and Beyond, pp. 198–203, May 2001.
3. A. Elwalid and D. Mitra, "Design of generalized processor sharing schedulers which statistically multiplex heterogeneous QoS classes," Proc. IEEE INFOCOM, vol. 3, pp. 1220–1230, Mar. 1999.
4. A. Jalali, R. Padovani, R. Pankaj, "Data throughput of CDMA-HDR a high efficiency - high data rate personal communication wireless system," VTC2000, vol. 3, pp. 1854–1858, Apr. 2000.
5. D. J. Goodman, R. A. Valenzuela, K. T. Gayliard, and B. Ramamurthi, "Packet reservation multiple access for local wireless communications," *IEEE Trans. Commun.*, vol. 37, no. 8, pp. 885-890, Aug. 1989
6. 1xEV-DV Evaluation Methodology-Addendum (V6), WG5 Evaluation AHG, Jul. 2001.

Packet Delay Analysis of Dynamic Bandwidth Allocation Scheme in an Ethernet PON

Chul Geun Park[1], Dong Hwan Han[2], and Bara Kim[3]

[1] Department of Information and Communications Engineering,
[2] Department of Mathematics,
Sunmoon University, Asan-si, Chungnam, 336-708, Korea
{cgpark, dhhan}@sunmoon.ac.kr
[3] Department of Mathematics, Korea University, Anam-dong,
Seongbuk-ku, Seoul, 136-701, Korea
bara@korea.ac.kr

Abstract. In this paper, we deal with the packet delay of a dynamic bandwidth allocation(DBA) scheme in an Ethernet passive optical network(EPON). We focus on the interleaved polling system with a gated service discipline. On the access side, input traffic may be aggregated from a number of users. So we assume that input packets arrive at an optical network unit(ONU) according to Poisson process. We use a continuous time queueing model in order to find the mean waiting time of an arbitrary packet. We give some numerical results to investigate the delay performances for the symmetric polling system with statistically identical stations.

1 Introduction

The passive elements of an EPON consist of a single, shared optical fiber, splitters and couplers. The active elements consist of an optical line terminal(OLT) and multiple optical network units(ONUs). The EPON is a point to multi-point network in the downstream direction and a multi-point to point network in the upstream direction[1]. The OLT resides in the local exchange, connecting the access network to the Internet and allocates the bandwidth to the ONUs by a polling scheme to transmit the Ethernet packet to the OLT. The ONU is located at the end user location and provides the interface between the OLT and customer networks to send broadband voice, data, and video traffic. In an EPON, the process of transmitting data downstream from the OLT to multiple ONUs is broadcast in variable length packets according to the IEEE 802.3[2].

In the upstream direction, the ONU should share the channel capacity and resources. We know that the limitation of TDMA approach is the lack of statistical multiplexing gain. It also true that some time slots remain unutilized even if traffic load is very high[3]. A dynamic scheme that reduces time slot size when there are no data would allow the excess bandwidth to be used by other ONUs. The obstacle of implementing such a scheme is in the fact that the OLT

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 161–168, 2005.

does not know exactly how many bytes of data each ONU has to transmit. Fortunately, Kramer et *al.*[3,4] proposed an OLT-based interleaved hub-polling scheme to support dynamic bandwidth allocation(DBA) and proposed a DBA scheme with a maximum transmission window size limit to avoid the monopolization of ONUs with high data volume. They also investigated how an EPON transmission mechanism(multi-point control protocol) can be combined with a strict priority scheduling algorithm specified in the IEEE 802.1D[5]. All of the previous studies[2-5] dealt with packet delay analysis of static and dynamic bandwidth allocation schemes by only simulation. Park et *al.*[6] used the linear system for the queue length distributions in order to investigate the mean packet delay by the queueing approach.

In this paper, we try to analyze the mean packet delay of a DBA scheme with a gated interleaved polling algorithm by using the closed form solution for the queue length distributions. Polling algorithms have been employed in many communication systems with data link protocols. From the viewpoint of queueing theory, it is a cyclic server system with multiple queues. The classification of polling models is with respect to the types of service discipline: (a) exhaustive, (b) gated and (c) limited[7]. In the gated service discipline, the server continues to serve only those packets which are waiting at the station when it is polled.

2 System Model for DBA Scheme

We consider an access network consisting of an OLT and N ONUs connected by an EPON. From the access side, traffic arrive at an ONU from a single user or a local area network(LAN), that is, traffic may be aggregated from a number of users such as voice, data and video terminals. Hence we assume that input traffic arrive at the ONU according to Poisson process. Ethernet packets should be waited in the first stage buffer of the ONU until the ONU is allowed to transmit all packets of the second stage buffer(Fig. 1). The transmission of the packets from the second stage buffer is triggered by the Grant message arrived from the OLT. The Grant message contains the granted window size allowed to be sent by the ONU.

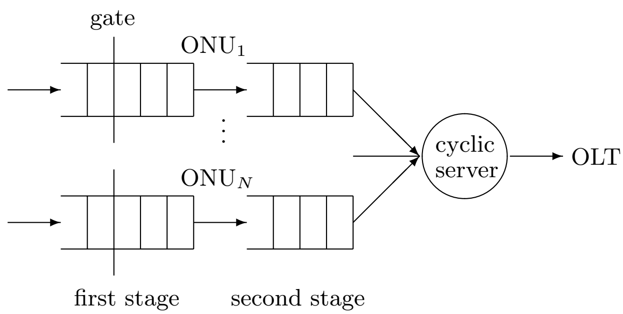


Fig. 1. Gated polling model of DBA scheme with two-stage buffer at an ONU

When the Grant message arrives at the ONU, the second stage buffer starts to transmit its packets to the OLT as well as the gate of the first stage buffer is closed. At the end of its transmission window, the ONU generates a Report message containing the number of bytes that remain in front of the gate in the first stage buffer and send it to the OLT. At the same time, the packets ahead of the gate in the first stage buffer are advanced into the vacant space in the second stage buffer and wait for the transmission of the next polling cycle. The Report message sent by the ONU tells the OLT its transmission window size when the Grant arrives. Simultaneously, the first stage buffer keeps receiving new data packets from its users.

Since the OLT knows how many bytes it has the authorized ONUi to send, it knows when the last bit from the ONU will arrive. Then knowing round trip time(RTT) for the next ONU$i+1$, the OLT can schedule a Grant to the ONU$i+1$, such that the first bit from the ONU$i+1$ will arrive with a small guard time after the last bit from ONUi. The guard time provides protection for fluctuations of RTT and control message processing time of multiple ONUs. The Requests(therefore data) from each ONU arrive in the same order(round robin) in every polling cycle. The Grants are scheduled with regard to the corresponding RTT and granted window sizes. The interested view of OLT scheduling is how the OLT should determine the granted window size. Kramer et *al.*[4] defined a few approach the OLT may take in making its decision: (i) Fixed, (ii) Limited, (iii) Gated, (iv) Constant Credit, (v) Linear Credit and (vi) Elastic. For the nice mathematical manipulation, we focus on the gated service discipline (iii).

3 Queueing Model and Analysis

In this section we deal with continuous time queueing model of the gated interleaved polling scheme with the first stage infinite queue(buffer in Fig. 1) and the second stage queue for packet transmission in order to investigate the packet delay distribution of the DBA algorithm in an EPON. We assume that the packets arrive at each station i(ONUi) according to a Poisson process with rate λ. The lengths of packets at station i are assumed to be independent and identically distributed with a general distribution function. For simplicity, we assume that the system is symmetric in the sense that the arrival rate and the packet length distribution are independent of the corresponding station. Let $B^*(s)$ be the Laplace-Stieltjes transform(LST) of the packet length distribution and let b and $b^{(2)}$ be the mean and the second moment respectively.

Stations are served in cyclic order of $1, 2, \cdots, N, 1, 2, \cdots$. The guard time (switchover) between station i and station $i+1$ is assumed to be independent and identically distributed with a general distribution function whose LST is denoted by $R^*(s)$. Let r and $r^{(2)}$ be the mean and the second moment of the switchover time distribution respectively. We also assume that packet arrival times, packet lengths and switchover times are independent. Further we assume that the stability condition $\rho \equiv N\lambda b < 1$ holds. Let C be a cycle time at steady state. Then we have

$$\rho = \frac{E[\text{service time during a cycle}]}{E[C]} = \frac{E[C] - Nr}{E[C]}.$$

From the above relation, the mean cycle time is obtained by $E[C] = \frac{Nr}{1-\rho}$.

Now, we derive the mean waiting time $E[W]$ of an arbitrary packet. By the pseudo conservation law[8], we have

$$\rho E[W] + \frac{\rho b^{(2)}}{2b} = \frac{\rho b^{(2)}}{2b(1-\rho)} + E[V|\text{switchover}], \tag{1}$$

where $E[V|\text{switchover}]$ is the expectation of the unfinished work in the system at an arbitrary time in switchover. Thus the mean waiting time $E[W]$ is obtained if we know $E[V|\text{switchover}]$. In what follows we focus on the derivation of $E[V|\text{switchover}]$.

Let $X_i(t)$ and $Y_i(t)$ be the numbers of packets in the second stage queue and the first stage queue at time t, respectively. Let τ_i be an epoch at steady state when a switchover from station $i-1$ to station i begins. Hereafter we admit $i-1$ to denote N for $i=1$ and $N+1$ to denote 1. Denote by $F_i(x_1, y_1; \cdots; x_N, y_N), i = 1, \cdots, N$, the joint probability generating function(PGF) of $(X_1(\tau_i), Y_1(\tau_i); \cdots; X_N(\tau_i), Y_N(\tau_i))$ by $F_i(x_1, y_1; \cdots; x_N, y_N)$, i.e.,

$$F_i(x_1, y_1; \cdots; x_N, y_N) = E\Big[\prod_{j=1}^{N} (x_j^{X_j(\tau_i)} y_j^{Y_j(\tau_i)})\Big].$$

We find a relation between $F_i(x_1, y_1; \cdots; x_n, y_n)$ and $F_{i+1}(x_1, y_1; \cdots; x_n, y_n)$. Let $\bar{\tau}_i$ be the end epoch of the switchover time that begins at τ_i. Hence a service period of station i begins at $\bar{\tau}_i$. Since the service period $(\bar{\tau}_i, \tau_{i+1})$ of the station i is the duration while $X_i(\tau_i)$ packets are served in the second stage buffer and the service time of each packet has the LST $B^*(s)$, we have

$$E[e^{-s(\tau_{i+1} - \bar{\tau}_i)} | X_1(\tau_i), Y_1(\tau_i); \cdots; X_N(\tau_i), Y_N(\tau_i)] = (B^*(s))^{X_i(\tau_i)}. \tag{2}$$

Let $A_j(t_1, t_2)$, $j = 1, \cdots, N$, denote the number of packet arrivals to station j during (t_1, t_2). Then, by (2), the conditional PGF of the numbers of packet arrivals during the service period $(\bar{\tau}_i, \tau_{i+1})$ of station i, given $(X_1(\tau_i), Y_1(\tau_i); \cdots; X_N(\tau_i), Y_N(\tau_i))$, is

$$\begin{aligned} &E[y_1^{A_1(\bar{\tau}_i, \tau_{i+1})} \cdots y_N^{A_N(\bar{\tau}_i, \tau_{i+1})} | X_1(\tau_i), Y_1(\tau_i); \cdots; X_N(\tau_i), Y_N(\tau_i)] \\ &\qquad = B^*(\sum_{j=1}^{N} (\lambda - \lambda y_j))^{X_i(\tau_i)}. \end{aligned}$$

Since the PGF of the numbers of packet arrivals during $(\tau_i, \bar{\tau}_i)$ is

$$\begin{aligned} &E[y_1^{A_1(\tau_i, \bar{\tau}_i)} \cdots y_{i-1}^{A_{i-1}(\tau_i, \bar{\tau}_i)} x_i^{A_i(\tau_i, \bar{\tau}_i)} y_{i+1}^{A_{i+1}(\tau_i, \bar{\tau}_i)} \cdots y_N^{A_N(\tau_i, \bar{\tau}_i)}] \\ &\qquad = R^*(\lambda - \lambda x_i + \sum_{j=1, j\neq i}^{N} (\lambda - \lambda y_j)), \end{aligned}$$

the conditional joint PGF of packet arrival numbers during (τ_i, τ_{i+1}) is given by

$$E[y_1^{A_1(\tau_i,\tau_{i+1})} \cdots y_{i-1}^{A_{i-1}(\tau_i,\tau_{i+1})} x_i^{A_i(\tau_i,\bar{\tau}_i)} y_i^{A_i(\bar{\tau}_i,\tau_{i+1})} y_{i+1}^{A_{i+1}(\tau_i,\tau_{i+1})} \cdots y_N^{A_N((\tau_i,\tau_{i+1})} | X_1(\tau_i), Y_1(\tau_i); \cdots ; X_N(\tau_i), Y_N(\tau_i)]$$
$$= R^*(\lambda - \lambda x_i + \sum_{j=1, j\neq i}^{N} (\lambda - \lambda y_j)) B^*(\sum_{j=1}^{N} (\lambda - \lambda y_j))^{X_i(\tau_i)}. \quad (3)$$

By the gated interleaved policy, we obtain $X_i(\tau_{i+1}) = Y_i(\tau_i) + A_i(\tau_i, \bar{\tau}_i)$. This equation indicates the fact that the number $X_i(\tau_{i+1})$ of the packets in the second buffer at station i at time τ_{i+1} is the sum of the number $Y_i(\tau_i)$ of the packets in the first buffer at station i at time τ_i and the number of packet arrivals to station i during $(\tau_i, \bar{\tau}_i)$. Therefore, together with (3), we obtain

$$F_{i+1}(x_1, y_1; \cdots ; x_N, y_N) = R^*(\lambda - \lambda x_i + \sum_{j=1, j\neq i}^{N} (\lambda - \lambda y_j)) F_i(x_1, y_1; \cdots ; x_{i-1}, y_{i-1}; B^*(\sum_{j=1}^{N} (\lambda - \lambda y_j)), x_i; x_{i+1}, y_{i+1}; \cdots ; x_N, y_N). \quad (4)$$

We assert that the following equation holds for a symmetric system

$$F_2(x_1, y_1; \cdots ; x_N, y_N) = F_1(x_2, y_2, x_3, y_3; \cdots ; x_N, y_N; x_1, y_1). \quad (5)$$

For a proof of the above equation, we can refer to [9]. By (4) evaluated at $i = 1$ and (5), we have

$$F_1(x_2, y_2; x_3, y_3; \cdots ; x_N, y_N; x_1, y_1)$$
$$= R^*(\lambda - \lambda x_1 + \sum_{j=2}^{N} (\lambda - \lambda y_j)) F_1(B^*(\sum_{j=1}^{N} (\lambda - \lambda y_j), x_1; x_2, y_2; \cdots ; x_N, y_N). \quad (6)$$

Differentiating (6) with respect to x_i and y_i, $i = 1, \cdots, N$, at $x_1 = \cdots x_N = y_1 = \cdots y_N = 1$ yields

$$\begin{aligned}
&f_{(N,1)} = f_{(1,2)} + \lambda r, \\
&f_{(i-1,1)} = f_{(i,1)}, \quad 2 \leq i \leq N, \\
&f_{(N,2)} = f_{(1,1)} \lambda b, \\
&f_{(i-1,2)} = f_{(i,2)} + f_{(1,1)} \lambda b + \lambda r, \quad 2 \leq i \leq N,
\end{aligned}$$

where $f_{(i,1)} \equiv \frac{\partial}{\partial x_i} F_1(1, 1; \cdots ; 1, 1)$, $f_{(i,2)} \equiv \frac{\partial}{\partial y_i} F_1(1, 1; \cdots ; 1, 1)$. By solving the above equations, we obtain

$$f_{(i,1)} = \frac{\lambda r N}{1 - \rho}, \qquad f_{(i,2)} = \frac{\lambda r (N - i + \rho)}{1 - \rho}, \qquad i = 1, \cdots, N.$$

Hence the mean number of packets in the system at the beginning epoch of an arbitrary switchover is given by

$$\sum_{i=1}^{N} (f_{(i,1)} + f_{(i,2)}) = \frac{\lambda r N (3N + 2\rho - 1)}{2(1 - \rho)}.$$

Since the mean number of packets arrived at the system during the elapsed switchover time is $N\lambda r^{(2)}/(2r)$ at an arbitrary epoch in switchover, the mean number of packets in the system at an arbitrary epoch in switchover is

$$\sum_{i=1}^{N}(f_{(i,1)} + f_{(i,2)}) + \frac{N\lambda r^{(2)}}{2r} = \frac{\lambda r N(3N+2\rho-1)}{2(1-\rho)} + \frac{N\lambda r^{(2)}}{2r}.$$

Thus we obtain

$$E[V|\text{switchover}] = b\left(\frac{\lambda r N(3N+2\rho-1)}{2(1-\rho)} + \frac{N\lambda r^{(2)}}{2r}\right). \tag{7}$$

Substituting (7) into (1) yields

$$E[W] = \frac{\rho}{1-\rho}\frac{b^{(2)}}{2b} + \frac{r(3N+2\rho-1)}{2(1-\rho)} + \frac{r^{(2)}}{2r}.$$

By Little's formula, the mean number $E[L]$ of packets in a station(two buffers) at an arbitrary time is given by

$$E[L] = \lambda(E[W] + b).$$

4 Numerical Results

In this section we present some numerical results to show the performance of the proposed DBA scheme. There are several factors that can affect the performance of the scheme, such as packet arrival rate, switchover time and the number of stations. We consider the symmetric gated interleaved polling model of the DBA scheme in an access network consisting of an OLT and N ONUs connected by an EPON. We use 1 Gbps as the rate of the upstream link from an ONU to the OLT. The trimodal packet size distribution has been demonstrated in access networks[5,10]. We assume that three modes correspond to the most frequent packet sizes: 64 bytes(47%), 582/594 bytes(15%) and 1518 bytes(28%). In addition, we consider the other packet sizes of 300(5%) and 1300 bytes(5%) and inter-frame gap 20 bytes. Thus we assume that the packet size distribution has 84 bytes(47%), 320 bytes(5%), 608 bytes(15%), 1320 bytes(5%) and 1538 bytes(28%). The respective service times are 0.67, 2.56, 4.86, 10.6 and 12.3 [μs].

Fig. 2 illustrates how the traffic intensity has influence on the mean waiting time $E[W]$ of an arbitrary packet and the mean cycle time $E[C]$ of the polling system. In this figure, we choose $N = 16$ as the number of ONUs and $r = 5[\mu\text{s}]$ as the guard (switchover) time. The parameter λ is adjusted to achieve the desired traffic intensity $\rho = N\lambda b$. We can see that the mean waiting time and the mean cycle time becomes strictly larger as the traffic intensity becomes heavier.

Fig. 3 shows the effect of the guard time between two consecutive windows for the case of $\rho = 0.4, 0.5$ and 0.6 with the fixed parameter $N = 16$. We can see that the mean waiting time and the mean cycle time increase linearly when the guard time increases from $r = 1$ to 5.

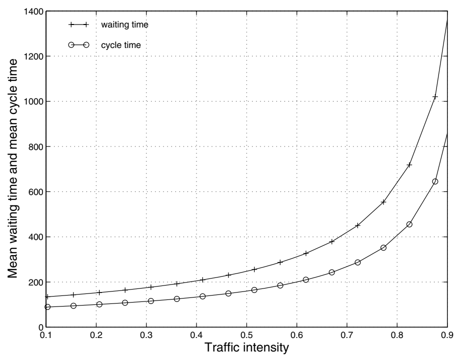


Fig. 2. Mean waiting time and cycle time versus traffic intensity

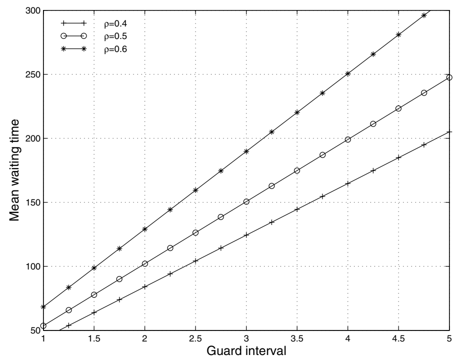


Fig. 3. Mean waiting time versus guard time

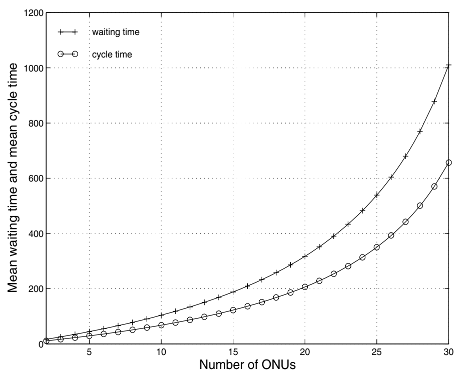


Fig. 4. Mean waiting time and cycle time versus number of ONUs

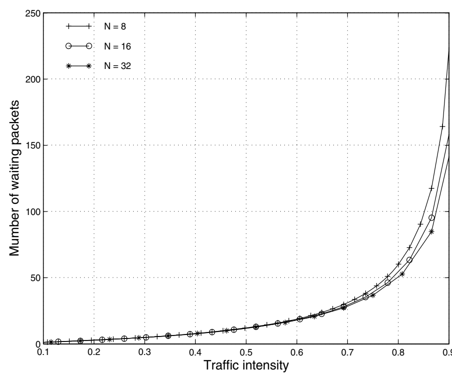


Fig. 5. Mean number of waiting packets versus traffic intensity

Fig. 4 shows the effect of the number of ONUs on the mean waiting time and the mean cycle time. The parameters $r = 5$ and $\lambda = 0.04$ are fixed, but the traffic intensity $\rho = N\lambda b$ varies between 0.05 and 0.77, when the number of ONUs varies from $N = 2$ to 30. We can see that the mean waiting time becomes exponentially larger according to the number of ONUs.

Fig. 5 shows the mean number of waiting packets in two queue versus traffic intensity for three cases of $N = 8$, 16 and 32 with the fixed parameter $r = 5$. We can see that the mean number of waiting packets increases sharply when the traffic intensity is in the heavy load above $\rho = 0.8$. We can also see that the mean number of waiting packets in each ONU is in reverse proportion to the number of ONUs that the polling system has.

5 Conclusion

In this paper, we study the packet delay performance of dynamic bandwidth allocation scheme in an EPON. To do this, we analyzed a gated interleaved polling system with an infinite waiting queue. We introduce a new interleaved polling system with two stage queueing buffers in order to model a dynamic bandwidth allocation scheme in an EPON. In addition, we also introduce a novel mathematical development to analyze the symmetric gated polling system with two stage queue. We gave some examples to show the effect of the traffic intensity, the number of ONUs and the guard time on the mean waiting time, the mean cycle time and the mean number of waiting packets in the two queues. We need further studies to deal with the number of waiting packets in the respective queues and the asymmetric gated polling system with the limited service discipline.

Acknowledgement

This research was supported by University IT Research Center Project.

References

1. Web ProForum Tutorials, Ethernet passive optical networks, http://www. iec.rog, The International Engineering Consortium
2. G. Kramer, B. Mukherjee and G. Pesavento, Ethernet PON(ePON): design and analysis of an optical access network, Phot. Net. Commun. **3** (2001) 307–319
3. G. Kramer and B. Mukherjee, Interleaved polling with adaptive cycle time (IPACT): Protocol design and performance analysis, Tech, rep. CSE-2001-4, Dept. of Comp. Sci., UC Davis, Aug. (2001)
4. G. Kramer, B. Mukherjee and G. Pesavento, Interleaved polling with adaptive cycle time (IPACT): A dynamic bandwidth distributin scheme in an optical access network, Phot. Net. Commun. **4** (2002) 89–107
5. G. Kramer, B. Mukherjee and G. Pesavento, Supporting differentiated classes of services in Ethernet passive optical networks, J. of Optical Networking **1** (2002) 280–290
6. C.G. Park, S.Y. Shim and K.W. Rim, Packet delay analysis of interleaved polling algorithm for DBA scheme in an EPON, The Third APIS, Istanbul, Turkey, January 13-14 (2004) 234–239
7. H. Tagaki, Analysis of polling systems, The MIT Press series (1986)
8. O.J. Boxma, Workloads and waiting times in single-server systems with multiple customer classes, Queueing Systems **5** (1989) 185–214
9. C.G. Park, B. Kim and D.H. Han, Queueing analysis of gated polling system for dynamic bandwidth allocation scheme in an EPON, J. Appl. Math. & Computing **16** (2004) 469–481
10. D. Sala and A. Gummalla, PON functional requirements: services and performance, Presented at the IEEE 802.ah meeting in Poland, Ore. (2001)

Inter-domain Advance Resource Reservation for Slotted Optical Networks

Abdelilah Maach[1], Abdelhakim Hafid[2], and Jawad Drissi[3]

[1] SITE, University of Ottawa, 800 King Edward PO. Box 450,
Ottawa, On, K1N 6N5, Canada
amaach@site.uottawa.ca
[2] Network Research Laboratory, University of Montreal,
Pavillon André-Aisenstadt H3C 3J7, Canada
ahafid@iro.umontreal.ca
[3] Department of Computer Science, Texas State University - San Marcos
601 University Drive, San Marcos, Texas 78666-4616
Jd30@txstate.edu

Abstract. In inter-domain optical networks, the major issues are bandwidth management and fast service provisioning. The goal is to provide optical networks with intelligent networking functions and capabilities in its control plane to enable rapid optical connection provisioning, multiplexing and switching at different granularity levels, especially wavelength and time slots. In this paper, we propose a new mechanism, for providing resource reservation in advance across multiple domains. The user specifies the service he/she requires, the start time of the reservation, and the duration of the reservation. The proposed scheme provides the user with the resources that can be reserved at the start time and other times in the future carefully selected. This is in opposition to existing approaches that respond with either an acceptation or a rejection of the request. We performed simulations to evaluate the proposed advance reservation scheme. The simulations results show lower user request blocking probability when using the proposed scheme.

1 Introduction

With the development being made in the networking technology, a new generation of applications (e.g., Grid applications) is emerging, requiring more and more resources. Furthermore, these applications have different needs and may require specific quality of services (QoS). The network has to guarantee QoS parameters by reserving adequate resources to avoid variations of the transmission quality.

Optical networks deploying dense wavelength division multiplexing (DWDM) [1, 2] and time division multiplexing [3, 4] are gathering more interest in research and industry. Their ability to establish connections between sources and destinations with the exact amount of bandwidth make them a pioneer of another generation of backbone networks providing services like bandwidth on demand [5]. Unlike routed wavelength optical networks [6, 7] where the bandwidth granularity is the whole

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 169–177, 2005.

wavelength, the time slotted optical networks use the concept of time slot. Thus, the resource reservation is based on time slots allocation.

To enable high scalability and large geographical coverage, many networks are connected together. Many protocols are developed, such as Border Gateway Protocol (BGP) [8, 9], to ease the communication between the different domains and provide reachability information. In this context, users submit connection requests only at the time when the connection should be established based on the information gathered by the inter-domain protocols. For each request, the network must decide immediately whether to accept or reject the request. However, with this model [10] there is always the uncertainty of whether the user will be able to establish the desired connection at the desired time; the user is provided with a limited choice which is either acceptation or rejection; indeed, the user is provided with no choice. Furthermore, it makes it difficult, or rather infeasible, for the network providers to minimize the blocking probability of the user requests (and thus increases its revenues) by rearranging/adjusting the resources allocation without degrading the agreed upon QoS [11]; indeed, resources rearrangements usually require traffic rerouting which usually causes data losses.

To overcome this undesirable situation, there is a need (a) to support advance reservation [12, 13, 14] of time slots; and (b) to provide the user with more choices than the simple accept/reject choice. In this case, the user requests the allocation of a certain number of time slots at a given start time for a given duration. In response, the allocation manager checks resources availability across all the involved networks, computes, and presents the user with the number of time slots that can be reserved at the requested start time and a certain times in the future carefully selected. With this model, the agreed upon reservations between the user and the network provider are confirmed before the start time; the user is "guaranteed" that the requested service can take place at the desired time at the agreed upon QoS (i.e., number of time slots). However, if the user requirements cannot be met (because of shortage in resources) The user and the network provider have enough time to engage a negotiation to reach a mutual agreement on an another start time and number of times slots to be reserved for the requested duration.

In this paper, we propose a novel scheme for an inter-domain advance reservation. The users generate requests, each specifying source and destination nodes, bandwidth requirements in terms of time slots, the starting time, and the duration. The proposed scheme computes the number of time slots that can be reserved, across all optical networks involved in supporting the request, at the requested start time and the number of time request that can be reserved at future times for the requested duration. These times are carefully computed to present the user with a minimum of "real" choices; "real" means that the choices do not contain redundant and/or inadequate choices. The paper is organized as follows. Section 2 presents the inter-domain advance reservation scheme and the algorithms used to compute the choices, called proposals, to be presented to the user. The performance analysis of the proposed scheme is presented in Section 3. Section 4 concludes the paper.

2 Inter-domain Advance Time Slot Allocation

The reservation in advance allows users to schedule and reserve the resources necessary for a future connection. The reservation results in an establishment of lightpaths (or TDM channels in case crossed networks are deploying TDM), from source to destination crossing many domains. In this paper we assume that the networks are deploying TDM.

The user requests a number of time slots from [startTime, startTime + *length*] between source and destination (the source and the destination may belong to different domains); this means that the user needs n times slots reserved over a period of time equal to length between source and destination.

In this reservation both inter-domain and intra-domain protocol are involved. The scheme we are proposing in this paper, is a gateway between inter and intra domain protocols. Indeed, the Advance Resource Manager (ARM) is in charge of exploring the resources at a future time in inter-domain links as well as the resources inside every domain that will be crossed during the connection life. The exploration and the reservation are performed inside an autonomous system by a Local Advance Resource Manager (LARM) [15]. Each LARM is associated with one network, realizes its portion of the reservation within the associated network and coordinates with other LARMs to realize the end-to-end reservation across all networks.

Upon receipt of the user request, the LARM returns a set of proposals where each proposal indicates the number of slots available for a period of *length* at some future times, carefully selected, including startTime. If n time slots are available for a period of length starting at startTime, then, only one proposal is returned i.e., the user request can be accommodated. Formally, a proposal is defined as a tuple <time, n, delay>; this means that there are n slots available over [time, time+length]; delay is the transit delay inside this network.

We assume that each network keeps track of its resources (i.e., time slots) availability presently and in the future; this can be performed by a software agent associated with the network or a central entity that maintains availability knowledge for the entire network.

LARMs realize the functionality locally within their associated communication networks, and interact/collaborate with each other to realize the end-to-end resource reservation. The set of these interactions is called *inter-LARM signaling*. In defining the inter-LARM signaling, a number of challenging issues arise: (a) the protocol should not be technology dependent; and (b) the protocol should allow for transmission of information necessary to support advance reservation. The existing signaling protocols do not tackle any of these issues. In the following, we briefly outline the salient features of our inter-LARM signaling by describing the inter-LARM interactions involved in setting up future inter-domain channels.

When ARM receives a setup request from a user, it determines the inter-network route for the channels (e.g., source routing can be used), determines the resources availability of the user network using the LARM of the user network. The LARM returns the list of proposals as described in [15]. Furthermore, ARM determines the resources availability of the link connecting this network and the next network; then, it produces a list of proposal. It computes a new list of proposals that combines both lists of proposals (LARM proposals and inter-domain link proposals). ARM propagates the setup request to the next LARM, including the route information and the combined list of proposals. Each LARM repeats this step.

The LARMs involved in a channel setup effectively commit the network resources for the channel only when all LARMs determine that resources are available in every network traversed by the channel. This occurs in the second phase of the setup where each LARM, starting from the destination LARM, sends to the previous LARM a "commit" message. In parallel with that the ARM commits the network resources on the inter-domain links. However, if the proposals provided do not meet the user requirements, an iterative mechanism is adopted to process the user request. Indeed, the allocation manager uses a k-shorted path algorithm to compute the k shortest path in terms of number of networks crossed or any other metric of interest. Then, it computes the end-to-end availability using an ARM on the 1st shortest path computed earlier. Using the end-to-end availability state, the ARM computes the end-to-end proposals and presents these proposals to the user. If the user selects one proposal, the allocation manager makes the necessary reservation (i.e., updating the availability state of the networks and links involved); otherwise, the allocation manager considers the 2nd shortest path and repeats the same process. If the user does not select one of the proposals that corresponds to the ith shortest path for $1<=i<=k$, then the user request is rejected. The value of k should be selected carefully; otherwise, the user request can be rejected while there exists an acceptable proposal to the user that was not computed (i.e., the corresponding path was not considered).

To minimize the interactions with the user, we can use a variation of the proposed approach to present the user with the best m proposals (m assumes a predefined value, e.g., the "human acceptable number" of proposals to present to the human user). This can be realized by computing the end-to-end proposals for each of the l path (e.g., l assumes a value equal to the number of possible paths between source and destination); then, the allocation manager orders all these proposals from the best proposal to the worst proposal; see [16] for more details on ordering proposals.

This variation minimizes the interactions with the user; however, it is not optimal. It uses a "brut" force (exhaustive search) to compute the m best proposals. We believe that this can be optimized using "smart" search in the end-to-end proposals space; we are in the process of investigating such a possible optimization.

3 Simulation Results and Analysis

We studied the performance of the proposed inter-domain advance reservation scheme and investigated its impact on the resource utilization of the network and the blocking probability. In this simulation we use the NSFNET network with 14 nodes (see figure 1).

We assume that each single fiber link is bi-directional, and has the same number of wavelengths operating at 50 Gbps. Each wavelength is divided into 50 small timeslots of 1 Gbps each. The propagation delay between two connected nodes ranges

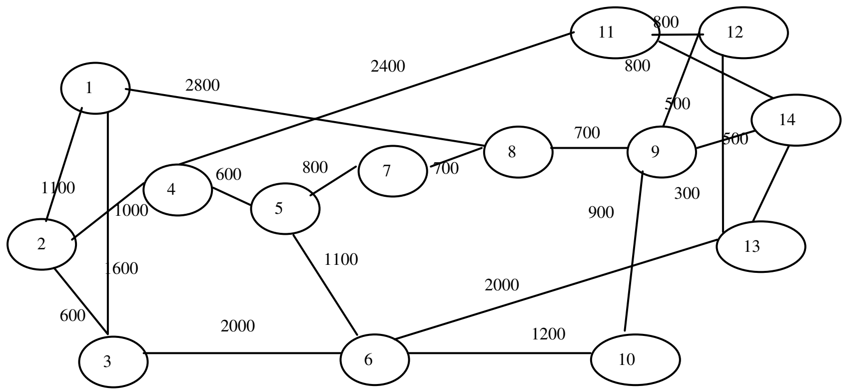


Fig. 1. NSFNET Topology with 14 Nodes

between 1.5 and 14 ms. Each Autonomous system ASi is represented by a Node Ni. We assume that at every intermediate AS, there is a chance to be delayed. 80% of the requests can get the resources they need at the desired time whereas 20% are delayed by a certain number of frames (uniformly distributed between 1 and 100).

We do not employ conventional buffers or wavelength converters in the switch. However, we assume that every switch is equipped with a slot interchanger [17]. Therefore, the resources are expressed in term of number of slots available in the given frame for a given link.

We use K-Dijkstra's algorithm (k=4 in this simulation) to get the list of the k shortest light paths between a source and a destination (this list is supposed to be provided by BGP data bases). The user request characteristics are captured by the following parameters:

- User request type: indicates the number of slots the user asks for. We assume that we have three classes 1, 2 and 3 called R1, R2 and R3 respectively. A request of type R1 asks for 20% to 40% of the frame size, a request of type R2 asks for less than 20%, and a request of type R3 asks for more than 40%. The users will request the popular class (i.e., R1), more often; the probability that a user generates a request of type Ri is given by pi. The following service request type pattern is assumed: p1=0.8, p2=0.1 and p3=0.1.
- User Request pattern in time: indicates the distribution of user requests over an interval of time; this distribution presents a peak. The peak represents a situation where the network is facing high load, the advance reservation is supposed to smooth the traffic out and reduce the blocking that may occur during the peak period. A normal distribution, characterized by its mean (3.5) and its variance (60), is selected to model the evolution of this parameter.

Besides the request type and request pattern, there are two other parameters we used in our simulation:

- Maximum delay parameter (MDP): indicates the maximum difference (between the requested start time and a delayed start time) which is acceptable to the user;

a value of 0 for this parameter means that the user does not accept any delay with respect to the requested start time. This parameter reflects the user negotiation and how long he/she accepts to delay the requested start time.

- The total number of requests which defines the network load.

The selection of the best proposal (which depends on each user) is constrained by the following policy:

- If more than one proposal satisfies the user request then the one going through the shortest path is selected.
- If there is no flow with enough capacity to carry the whole request then many flows should be used together to accommodate the request. In this case, only those with the closest time to the desired time are selected.

The goal of the experiments is to study the performance of the proposed scheme compared to immediate reservation.

The first set of simulations investigates the number of requests accepted by the proposed scheme and by the immediate reservation scheme respectively; this also reflects the number of requests that are rejected due to shortage in resources. The total number of requests generated in this simulation is 3000.

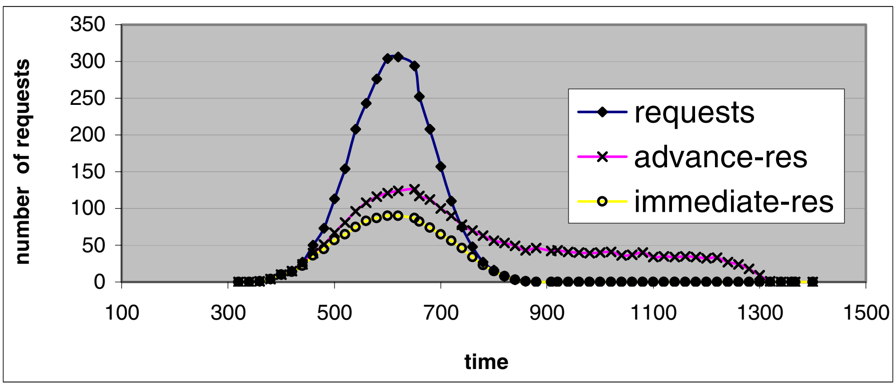


Fig. 2. Number of Requests Allocated with and without Advance Reservation

Figure 2 shows the distribution of allocated requests. Obviously, advance reservation accepts more requests. The maximum number of requests is allocated at the peak; this is due to the fact that all the network resources are available before the peak. This maximum cannot be hold; indeed, as soon as the network gets saturated a new reservation requires a departure of another request. Figure 2 also shows the requests being accommodated, with a delay, by the proposed scheme (i.e., requests that could not be served at the peak time and are rescheduled for a future reservation). The advance reservation tries to smooth out the heavy demand on bandwidth.

In the second set of simulations, we investigate the impact of our reservation scheme on the network utilization. Figure 3 shows the network utilization for immediate reservation and advance reservation with different values of MDP. Advance reservation is performing better than immediate reservation. Indeed, the advance reservation is using the resources made available after the pick. As expected when MDP increases the network utilization is improved using the proposed scheme.

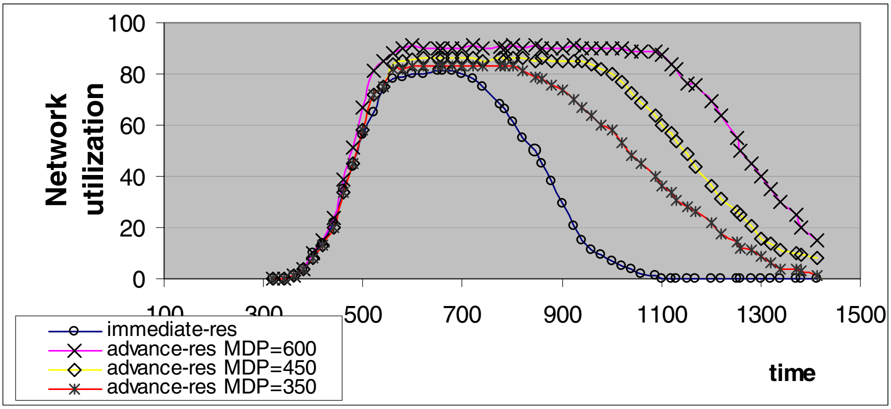


Fig. 3. Network Utilization with Immediate Reservation and Advance Reservation with different values of MDP

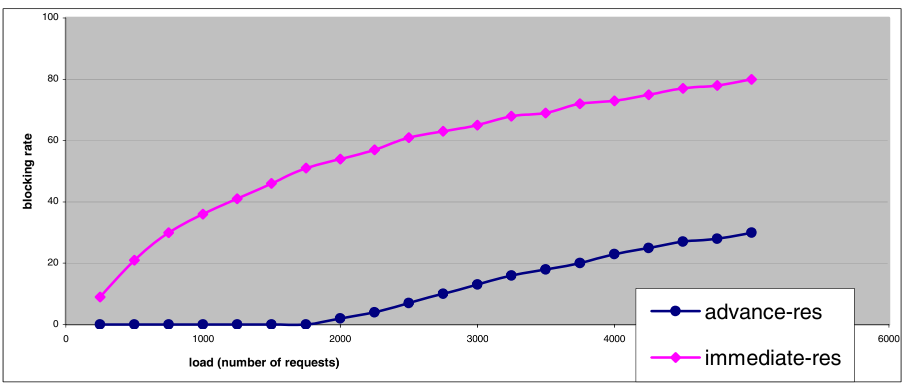


Fig. 4. Blocking Rate for Allocation with and without Advance Reservation

The advance reservation provides a flexible way to control the network utilization. Indeed, by increasing MDP, one can increase the capacity of the network. This can be used to face the high demands that may occur temporarily. In this case, the requests may suffer additional delay, which may be not suitable for a number of classes of applications. To accommodate different classes of applications, different values of

MDPs can be used; each class of applications has its own MDP. A class of applications that do not tolerate delay will have a value of 0 for the MDP.

In the third set of simulations, we investigate the blocking rate while varying the number of total requests. The blocking rate reflects the number of requests that should be rejected because of shortage in resources; it is the number of rejected requests over the total number of requests.

Figure 4 shows that when the load increases, advance reservation suffers no loss at all and keeps a zero loss until a certain load (in this case around 2000 request). This value depends on the network capacity. It could be improved whether by increasing the MPD or by enhancing the network physical resources. We observe also, that when the load is larger than this limit the loss increases linearly. This is because beyond the network capacity (the saturation load) all the requests are simply dropped. It is important to notice that, unlike advance reservation, in regular reservation the rejection starts at early stage. The performance difference between the two techniques is considerable. This is due to the fact that advance reservation uses the resources that are available after the peak. The larger the MPD the larger is this different.

4 Conclusion

In this paper, we proposed a new inter-domain reservation scheme in slotted optical networks. In this scheme the source specifies the bandwidth required to another destination, the service start time and the duration of the reservation. If reservation is not possible because of shortage in resources, other alternatives are provided to the user. Both the bandwidth (in term of number of time slots) and the start time are subject to negotiation.

Advance reservation provides the user with more choices than the simple accept/reject choice of existing approaches; the inter-domain signaling protocol, proposed in this paper to realize inter-domain reservation is network technology independent and easy to implement. Simulations show the proposed scheme allows for better resources utilization and lower blocking probability for channel requests

Currently, we are investigating issues related to the accommodation of time slots in different flows. Indeed, a time slot maybe delayed in an intermediate node and that may affect the global synchronization of resource allocation.

References

1. Song, S., Wu, Z.: A broadband integrated services network architecture based on DWDM, Proceedings of the Electrical and Computer Engineering Canadian Conference, Vol. 1 (2001) 347-352.
2. Golmie, N., Ndousse, T.D., Su, D.H: A differentiated optical services model for WDM networks, IEEE Communications Magazine, Vol. 38-2 (2000) 68 - 73.
3. Liew, S.Y, Chao, H.J.: On slotted WDM switching in bufferless all-optical networks, Proceedings 11th Symposium High Performance Interconnects (2003) 96 -101

4. Ramamirtham, J., Turner, J.: Time sliced optical burst switching, INFOCOM 2003, Vol. 3-30 (2003) 2030 – 2038
5. Bianco, A.., Galante, G., Leonardi, E., Neri, F., Nucci, A..: Scheduling algorithms for multicast traffic in TDM/WDM networks with arbitrary tuning latencies, Computer Networks: The International Journal of Computer and Telecommunications Networking, Vol. 41-6 (2003) 727-742
6. Banerjee, D., Mukherjee, B.: Wavelength-routed optical networks: linear formulation, resource budgeting tradeoffs, and a reconfiguration study, IEEE/ACM Transactions on Networking, Vol. 8-5 (2000) 598 – 607
7. Grosso, A., Leonardi, E., Mellia, M., Nucci, A.: Logical Topologies Design over WDM Wavelength Routed Networks Robust to Traffic Uncertainties, IEEE Communications Letters, Vol. 5-4 (2001) 172–174
8. Rexford, J., Wang, Z., Xiao, Zhang, Y.: BGP routing stability of popular destinations, Proceedings Internet Measurement Workshop (2002)
9. Yang, X., Ramamurthy, B.: Inter-Domain Dynamic Routing in Translucent Optical Transport Networks, The Workshop on High-Speed Networking, held in conjunction with IEEE INFOCOM (2003)
10. Barry, R.A., Humblet, P.A.: Models of blocking probability in all-optical networks with and without wavelength changers, IEEE J. Select. Areas Commun., Vol. 14 (1996) 867-878
11. Hashmani, M., Yoshida, M., Ikenaga, T., Oie, Y.: Management and Realization of SLA for Providing Network QoS, Proceedings the First International Conference on Networking-Part 1 (2001) 398- 408
12. Norden, S., Turner, J.: DRES: network resource management using deferred reservations, Global Telecommunications Conference, Vol. 4 (2001) 2299 – 2303.
13. Guerin, R.A., Orda, A.: Networks with advance reservations: the routing perspective, INFOCOM 2000, Vol. 1, (2000) 118 – 127
14. Schill, A., Kuhn, S., Breiter, F.: Resource reservation in advance in heterogeneous networks with partial ATM infrastructures, IEEE INFOCOM, Vol. 2 (1997) 611– 618
15. Maach, A., Hafid, A., Bochmann, G.v.: Resource Reservation in Advance in Optical Network, Proceedings the Third International Conference on Optical Communications and Networks (2004)
16. Hafid, A., Natarajan, N., Falchuk, B., Roy, A., Pastor, J.: A network management prototype for information delivery and planning with quality of service guarantees, 21st IEEE MILCOM, Vol. 12 (2000) 1026-1030
17. Maach, A., Zeineddine, H., Bochmann, G.v.: Bandwidth allocation scheme in optical TDM, Proceedings 7th IEEE HSNMC (2004) 801-812

Virtual Source-Based Minimum Interference Path Multicast Routing with Differentiated QoS Guarantees in the Next Generation Optical Internet

Suk-Jin Lee[1], Kyung-Dong Hong[1], Chun-Jai Lee[1], Moon-Kyun Oh[2], Young-Bu Kim[2], Jae-Dong Lee[3], and Sung-Un Kim[4]

[1] Pukyong National University,
Daeyeon 3-Dong Nam-Gu, Busan, 608-737, Korea
{stone, omnibus, leecjcpp}@mail1.pknu.ac.kr
[2] Electronics and Telecommunications Research Institute,
161 Gajeong-Dong, Yuseong-Gu, Daejeon, 305-350, Korea
{mkoh, ybkim}@etri.re.kr
[3] Kyungnam College Information & Technology,
Jure 2-Dong, SaSang-Gu, Busan, 617-701, Korea
jdlee@kit.ac.kr
[4] Corresponding Author: Pukyong National University,
Daeyeon 3-Dong Nam-Gu, Busan, 608-737, Korea
kimsu@pknu.ac.kr

Abstract. WDM networks using wavelength routing are considered to be potential candidates for the next generation backbone networks. One of the critical issues of the future network is a provision of proper QoS guarantees for a wide variety of multimedia multicast service. This paper concerns with the problem of optical multicast routing with QoS guarantees in WDM networks. Using Virtual Source (VS) nodes, we proposes a new MCRWA method for a multicast session, combining the VS-based tree method with Multi-Wavelength Minimum Interference Path Routing (MW-MIPR). This paper also proposes a QoS MCRWA method in combination with QoS constraints and a recovery strategy based on the differentiated QoS service model to provide QoS guarantee for a wide variety of multicast application in the next generation optical networks.

1 Introduction

As the Internet traffic continues to increase exponentially, WDM networks with terabits per second bandwidth per fiber become a natural choice as a backbone in the next generation optical Internet. Moreover many applications such as television broadcast, movie broadcasts from studios, and video-conferencing are becoming increasingly popular. These applications require point-to-multipoint connections among the nodes in a network. Multicast provides an efficient way of disseminating data from a source to a group of destinations, so the multicast

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 178–185, 2005.

problem in the optical networks has been studied for years and many efficient multicast routing protocols have been developed.

To support multicast at the WDM layer, the concept of the light-tree was introduced in [1], which is a point-to-multipoint extension of a lightpath (i.e., an all-optical WDM channel). The key advantage of light-tree is that only one transmitter is needed for transmission and intermediate tree links can be shared by multiple destinations. To support all-optical multicasting efficiently, some nodes in a WDM network need to have the light splitting capability. An Multicast-Capable(MC) node, however, is expensive to implement throughout the network, so the concept of sparse-splitting was first introduced in [2]. With sparse splitting, only a small percentage of nodes in the network are MC nodes, and the rest are Multicast Incapable (MI). An MI node can forward an input signal only to one of the output ports; thus it cannot serve as a branching node of a light-tree.

In order to provide the multicast services, some multicast routing algorithms to construct multicast trees were proposed[2-4]. But the previous researches had some limitations[4].

To overcome the previous studies, we proposes a new MCRWA method for multicast sessions, combining the VS-based tree generation method with MW-MIPR that chooses a route that does not interfere too much with potential future connection requests[5]. Choosing the minimum interference paths, the new algorithm provides an efficient use of wavelength number in comparison with VS-based tree generation method.

This paper also proposes a QoS MCRWA in combination with QoS constraints and a recovery strategy based on the differentiated QoS service model to provide QoS guarantee for a wide variety of multicast applications in the next generation optical networks.

The rest of the paper is organized as follows: in section 2, we review properties of previous multicast tree generation methods and limitations. In section 3 we define the new MCRWA algorithm, and section 4 takes into account differentiated QoS MCRWA problem with QoS constraints and a recovery strategy. Experiment results showing effects of a new algorithm and our conclusion are presented in section 5 and 6, respectively.

2 Previous Researches

2.1 Source-Rooted Approach

A multicast tree is constructed with the source of a session as the root of the tree. The objective here is either to minimize total cost of a tree or to minimize individual cost of paths between the source and destinations. Depending on the objective, there are two methods to construct a multicast tree.

In source-based tree [2], the destinations are added to the multicast tree in a shortest path to the source of a multicast session. These algorithms provide a computationally simpler solution to the multicast tree generation, but have some limitations[3-4].

In Steiner-based tree [3], the destinations are added to the existing multicast tree one at a time in such a way that the total cost of the tree is minimized. These algorithms are computationally expensive. Hence, heuristics are provided to choose a node to which the present node can be connected.

But the source-rooted approach has a following limitation. In a wide area network destinations of a session are distributed throughout the networks. Hence, the delay incurred in constructing the tree will be very high. There should be a simple procedure to add and delete a node from the session, because deleting or adding destinations to the existing session may change the structure of the tree.

2.2 Virtual Source-Rooted Approach

This algorithm overcomes the limitation of source-rooted approach. In VS-based method[4], some nodes in the network are chosen as VS nodes. Here VS nodes have splitting and wavelength conversion capabilities and can transmit incoming messages to any number of outgoing links on any wavelengths. These VS nodes are interconnected in such a way that a lightpath is established between every pair of VS nodes. These interconnectivities among the VS nodes are used when the multicast tree is constructed.

Comparing with Source-rooted approach, VS-based approach method has the follow advantages. Source need not know about the location of destinations. There is a maximum of three light hop distance from a source to any destinations. Hence, fairness among destination is achieved. And the procedure of dynamic addition or deletion of members in the group is simple in comparison with Source-rooted approach. Whereas VS-based tree method has a critical default such like that as the number of VS nodes increases, the overhead due to the resources reserved for paths between VS nodes also increases.

In order to overcome the limitation of VS-based method, it needs a strategy to control the traffics of paths between VS nodes.

3 VS-MIPMR Algorithm

3.1 Definition

In VS-based tree method, as the number of VS nodes increase, the overhead due to the resources reserved for paths between VS nodes also increases. Therefore it needs an appropriate strategy to use the paths between VS nodes.

The Figure 1 illustrates the new algorithm. There are two potential source-destination pairs such as (S_1 , D_{11}& D_{12}) and (S_2 , D_{21}& D_{22}). When the segment 1 is chosen for the first multicast session, another multicast session may share the same link that can lead to high blocking probability by inefficiently using the resource due to the traffic concentration on the minimum-hop paths. If the connection between (S_2, D_{21}& D_{22}) pairs is set along segment 1 selected by min-hop routing as demanded, then this route may block the previous path when the capability of segment 1 is not large enough. Thus, it is better to pick segment 2 that has a minimum effect for other future connection requests, even though

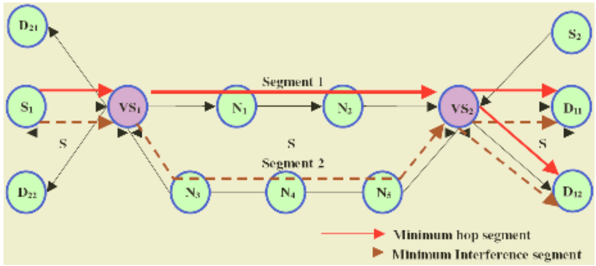


Fig. 1. Illustration of the new MCRWA algorithm

the path is longer than segment 1. Before formulation of the new algorithm, we define some notations commonly used in this algorithm as follow.

- (s, d): A source-destinations pair to want to construct the multicast tree.
- (a, b): A VS-nodes pair to require connections when constructing multicast trees.
- T_{sd}: A multicast tree constructed by minimum-hop path between the VS-nodes.
- $S_{vv}(i)$: The ith minimum-hop path connecting the path between the VS-nodes.
- α_{vv}: The weight for a segment between the VS-nodes.
- C_{vv}: The minimum-hop paths between the VS-nodes.
- F_{vv}: The set of wavelengths available in S_{vv}
- $W_{vv}(S_{vv}(i))$: The set of wavelength available in the ith path between the VS-nodes.
- $\Psi_{vv}(S_{vv}(i))$: The set of wavelengths assigned to $S_{vv}(i)$.
- $R_s(i)$: The weight for the ith path between the VS-nodes.

Based on these notations, the link weights are determined as follow:

$$MAX \sum F_{vv}/(\alpha_{vv} \cdot v_i) \tag{1}$$

$$\begin{cases} v_i(S) = 1 & if(s,d) : S_{vv}(i) \in C_{vv} \cap \{W_{vv}(S_{vv}(i)) - \Psi(S_{vv}(i))\} = \emptyset \\ v_i(S) = 0.5 & if(s,d) : S_{vv}(i) \in C_{vv} \cap \{W_{vv}(S_{vv}(i)) - \Psi(S_{vv}(i))\} \neq \emptyset \\ v_i(S) = 0 & otherwise \end{cases} \tag{2}$$

$$R_s(i) = \sum_{\forall (s,d) \in S \setminus (a,b)} \alpha_{vv} \cdot v_i(s) \qquad \forall S \in S_{vv} \tag{3}$$

Equation (1) presents the minimum interference of the wavelength path decision between the VS-nodes in order to choose the optimal path according to the present multicast session request. Equation (2) allocates the differentiated values to the ith segment between VS nodes which were determined by the previous multicast sessions and the wavelength available, according to the degree of effect of segments that have a minimum-hop number path requested by the previous multicast sessions. Equation (3) presents the summation of the differentiated values to the ith segment according to the given multicast session request.

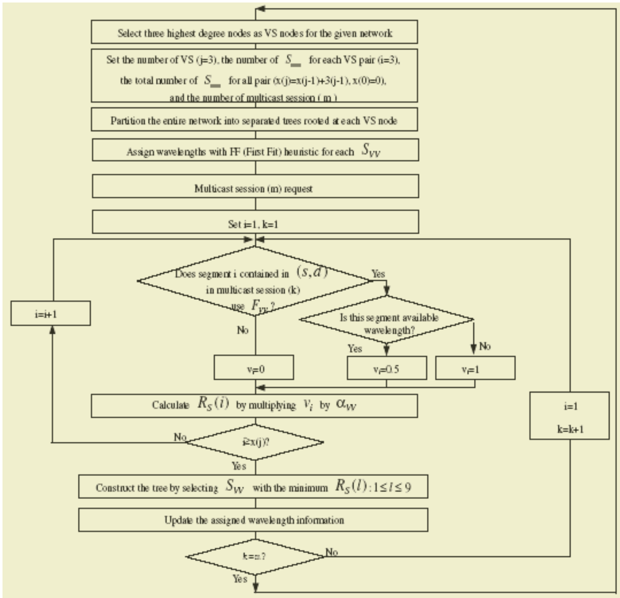


Fig. 2. Procedure of VS-MIPMR

Therefore the algorithm decides the light path that has a minimum value of segment weight $R_s(i)$. Figure 2 illustrates such a procedure of VS-MIPMR.

4 Differentiated QoS MCRWA and Recovery Mechanism

The explosive increase of traffic volumes and a variety of real-time multicast applications with the rapid development in internet technologies call for the next generation optical networks based on DWDM. One of the critical issues of the future network is a provision of proper QoS guarantees for a wide variety of multimedia multicast service[6]. In this section, we introduce QoS constraints to guarantee a satisfying QoS for multicast application services and propose differentiated QoS MCRWA with recovery schemes for each service in the next generation optical Internet.

4.1 QoS Constraints

A general classification of Internet service may be divided into differentiated service classes, i.e., premium service, assured service, and best-effort service based on the level of their QoS [7].

In this section, we provide three main approaches to QoS evaluation in order to provide with differentiated QoS in the next generation optical Internet. The first one is related to the transmission quality attribute of optical signal. In DWDM networks, optical signals passing through optical network elements undergo many undesired transmission impairment throughout their routes. The

Classification criteria	Class0	Class1	Class2	Class3	Class4
Nest Generation Internet service	Premium service		Assured service		Best-effort service
	Virtual leased line service	Bandwidth pipe for data service	Minimum rate guarantee service	Qualitative Olympic service: Gold, Silver, Bronze	
Multicasting service	HDTV, Video Conference, VoIP, Tele-Learning	Digital library, Robotics, Shared Virtual reality	Tele-Immersion Tele-Instrumentation	Data mining	The others
BER (Q)	10^{-16}(8)	10^{-14}(7.5)	10^{-16}(8)	10^{-14}(7.5)	10^{-10}(6)
el. SNR	18.06dB	17.5dB	18.06dB	17.5dB	15.56dB
OSNR (f_0=10Gbit/s)	20.67dB	20.1dB	20.67dB	20.1dB	18.17dB
PLV (4dB<Noise figure<6dB)	39.59mW <PLV <51.40mW	37.04mW <PLV <47.38mW	39.59mW <PLV <51.40mW	37.04mW <PLV <47.38mW	30.51mW <PLV <37.38mW
Resource allocation	(C band: 1530nm - 1565nm)	(L band: 1565nm - 1625nm)	(C band: 1530nm - 1565nm)	(L band: 1565nm- 1625nm)	(L band: 1565nm - 1625nm)
Recovery Scheme	Protection/segment disjoint (1:1) light tree		Protection/mixed segment disjoint (1:1 + 1:N) light tree		Restoration at IP level
Recovery time	<50msec (Detection time: <100msec)		<50msec (Detection time: <100msec)		1-100 sec (Detection time: 100msec - 180sec)

Fig. 3. Differentiated multicast QoS service model in next generation optical Internet

impaired transmission signals are transmitted to another node, and accumulated by other elements.

Therefore the optical signal can't provide sufficient QoS services throughout the networks. Especially in case of the multicast service, an optical signal undergoes severe power loss due to the splitter in MC-OXC. Such an optical signal's impairment is determined by calculating the BER in the receiving nodes. We can estimate the BER in an optical network by Q-factor as a new parameter evaluating signal quality [9]. It measures the SNR based on assuming Gaussian noise statistics on the eye-diagram. Thus, the QoS parameter related to the transmission quality attribute of optical signal can be determined [8-10]. The measured SNR must strictly comply with BER, el. SNR, and OSNR constrains for each service presented in figure 3 on all links of the selected route.

The second one is related to the resource quality attribute of optical signal. Generally the premium service must guarantee reliability when setting up the lightpath. Therefore we allocates wavelength of C-band that provides least attenuation for the premium service, so the excellent optical quality is provided for it. Then we allocate wavelength of L-band for the assured and best-effort services that require less reliability in comparison with the premium service. As a result, wavelength effectiveness and the premium service are guaranteed by assigning the previously assignment ratios to the corresponding services[5].

The last one is related to the survivability of the lightpath. In the high-speed network based on DWDM, a fault or an attack of optical signal will cause severe impairments due to the tremendous transmission quantity. Therefore it is important to provide protection and restoration mechanism to guarantee the transparence of lightpath against various problem such as cutting of lightpath and impairment of wavelength. As a result, the differentiated survivability methods based on each service type are required in the next generation optical Internet based on DWDM, as shown in figure 3.

4.2 Differentiated QoS MCRWA with Survivability

In this section, we provides the differentiated QoS MCRWA based on VS-MIPMR considering the QoS attributes that include the transmission and resource quality of optical signal and the survivability of the lightpath mentioned in section 4.1. Also we apply FF wavelength assignment method due to its simple complexity and implementation.

In order to provide optimal QoS MCRWA, we must estimate OSNR and BER mentioned in section 4.1. Such QoS parameters are obtained by measuring the power level and Q-factor on the links. Here OSNR, BER, and PLV of a link should satisfy each theldshold in figure 3.

5 Experiment Results

We conducted simulations to evaluate the performance of VS-MIPMR and differentiated QoS MCRWA. The network model used in the simulations is the NSFnet which topology consists of 14 nodes and 20 links, and we assume that the connection request arrive randomly according to the Poisson process. To prove the efficiency of VS-MIPMR algorithm proposed in section 3, we analyzed the wavelength numbers and the wavelength channels of VS-MIPMR and Virtual Source-based method; here the group size (GS) that determines the number of members to construct a multicast session is 0.3 and 0.4. Figure 4 reveals that the proposed algorithm outperforms Virtual Source-based method due to the selection of the minimum interference routes. Therefore VS-MIPMR can accomplish approximately 16-22% and 20-25% improvement of the wavelength numbers, in GS 0.3 and in GS 0.4, respectively, in comparison with that of the Virtual Source-based multicast method. Even though VS-MIPMR needs slightly more numbers of wavelength channels due to the detour paths to avoid congestion links, we can identify more and more decreasing loss of wavelength channels.

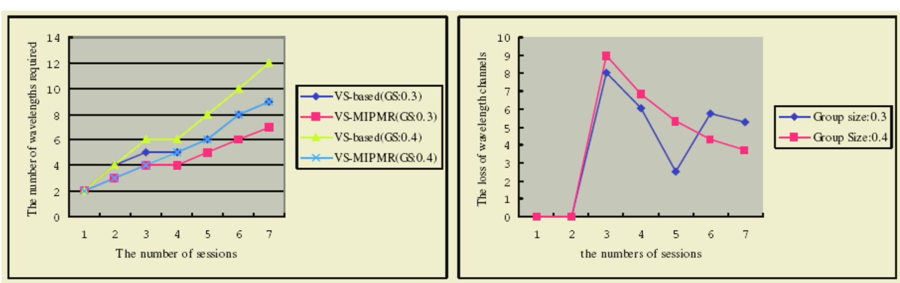


Fig. 4. The number of wavelengths and The loss of wavelength channels over session

6 Conclusion

In this paper, we proposed a new MCRWA algorithm that combines the VS-based method with MW-MIPR that chooses a route that does not interfere too much with potential future connection requests, and presented the QoS MCRWA mechanism with differentiated survivability based on the proposed VS-MIPMR.

Simulation results show that our new algorithm significantly improves the utilizations of wavelength number over sessions, comparing with Virtual Source-based method. Therefore, the proposed multicast method can be applied to Generalized Multi-protocol Label Switching (GMPLS) in the DWDM networks due to the provision of differentiated services and protection schemes.

As a future research, we will study VS-MIPMR based on various network model that have more nodes than our study, and will conduct simulations to verify the efficiency of the algorithm. In addition to verifying the efficiency, we will expand MCRWA problem for various multicast applications in a variety of service classes in the next generation optical networks.

Acknowledgment

This work was supported by grant No.(R01-2003-000-10526-0) from Korea Science & Engineering Foundation.

References

1. L. H. Sahasrabuddhe, et al.: Light trees:optical multicasting for improved performance in wavelength routed networks, Communications Magazine, IEEE, vol. 37, Issue: 2, Feb. 1999, pp. 67-73
2. Xijun Z., et al.: Constrained multicast routing in WDM networks with sparse light splitting, Journal of Lightwave Technology, Vol.18, No. 12, Dec 2000, pp.1917-1927
3. N. Sreenath, et al.: Virtual source based multicast routing in WDM optical networks, Proceedings IEEE International Conference on, 5-8 Sep. 2000, pp 385-389
4. N. Sreenath, N. K. M. Reddy, G. Mohan, and C. S. R. Murthy : Virtual source based multicast routing in WDM networks with sparse light splitting, High Performance Switching and Routing, 2001 IEEE Workshop, 29-31 May 2001, pp 141-145
5. Jong-Gyu Hwang, et al.: A RWA Algorithm for Differentiated Services with QoS Guarantees in the Next Generation Internet based on DWDM Networks, Photonic Network Communications, vol. 8, Issue 3, Nov. 2004, pp. 319-334
6. http://www.east.isi.edu/projects/NMAA/
7. S. Blake, et al.: An Architecture for Differentiated Service, RFC 2475, IETF, 1998
8. P. S. Andre, et al.: Optical-signal-quality monitor for bit-error-ratio assessment in transparent DWDM networks based on asynchronously sampled amplitude histogram, Journal of Optical Networking, vol. 1, no. 3, Mar. 2002, pp. 118-127
9. WorldCom's White Contribution COM-D 126: Proposed Optical Performance Monitoring Parameters for OTN, ITU-T SG 15 contribution, Oct. 2001
10. http://www.cisco.com/en/US/products/hw/optical/ps2011/products-data-sheet09186a008008870d.html
11. KDDI's white Contribution D.97 (WP4/15): Recent technical information on C- and L-band in Optical Transmission Systems, ITU-T SG15 Contribution, Feb. 2001

Multiple Failures Restoration by Group Protection in WDM Networks

Chen-Shie Ho[1], Ing-Yi Chen[2], and Sy-Yen Kuo

[1] Department of Computer Science and Information Engineering,
Van Nung University, Chung-Li, Tao-Yuan, Taiwan
hocs@vnu.edu.tw
[2] Department of Computer Science and Information Engineering,
National Taipei University of Technology, Taipei, Taiwan
ichen@ntut.edu.tw
[1,2] Department of Electrical Engineering,
National Taiwan University,
Taipei, Taiwan
hocs@lion.ee.ntu.edu.tw

Abstract. Single link/node failures are often occurred in daily network operation, which make them the most widely considered failure model in the literatures. Besides these conventional failures, in this paper we will focus on a new failure model, the group failure model, which occurs infrequently but is critical for seamless providing of network service. We examine the influence of this new model to general WDM network survivability mechanism, and present capacity optimization techniques for static protection and then propose the heuristics for solving the disjoint routing problems with group protection for dynamic traffic environment. The extensive simulations are conducted and the results are discussed to examine the relative influence of various network metrics for the proposed heuristics.

1 Introduction

Wavelength-routed wavelength division multiplexing (WR-WDM) network has been widely used in large-scale long-haul core networks and it is imperative that these transport networks are implemented by effective fault tolerance mechanisms to minimize the huge avenue loss due to unpredictable failures [1][2][3]. The survivability mechanisms against different failure models in optical layer can be classified into two categories: the dynamic lightpath restoration and the preplanned protection scheme. These methods are either link/path-based or segment-based implementation [4]-[10]. All these strategies can be treated as special cases for group (sub-mesh)-based scenarios. The goal of group switching is attempting to make more efficient management for these hierarchical protection autonomous areas and completely exploits the potential resource within the protection group. Group switching can be viewed as the partition of the mesh transport network into groups to enhance the management efficiency and failure restoration. Restoration can be provided by either the client service layer or directly the optical layer. In this paper, we only consider the protection and restoration

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420,, pp. 186–193, 2005.

in physical WDM layer. In group switching restoration scenario, the traffic connection path is divided into several segments with unequal hop-length and located into several distinct group areas. The path segmentation will be determined by group partition criteria. The group partition methods can be static or dynamic, which has the different influence to the quality of survivability. This paper will mainly consider the dynamic traffic condition on the wide area wavelength routed mesh topology based backbone environment with sparse wavelength conversion capability.

The survivability mechanisms proposed to date are mainly for single link or node failure, and for multiple failures which are random distributed among different portions in the network. The effect of multiple failures can be handled as a series accumulation of single failure and recovered one after the other in restoration process.

There is possibility in real world that the failures will occur within fixed range of zone due to bomb/missile or terrorist attack. This type of new failure model, the *group (clustering) failure*, is still an undiscovered failure object in the literature. The characteristics of group failure can be described as follows: (1) The failures belonging to the same group occurred simultaneously, which are not the same case as in substantial generation of successive single failure. All the traffic associated with the destroyed nodes or links will be influenced and must be recovered if the quality of protection for service should be guaranteed. (2) Since all the network resources in the affected group zone will be viewed as in the destroyed status, the survivable routing phase in RWA process for each reliable connection request arriving will be critical and should be designed depending on some service model to provide some degree of restorability. (3) The link/node failures locate on the bounded failed area will cause the differentiate restorability automatically. The size of the affected failure zone will have the different influence on affected connections passing through the destroyed area. (4) The recovery order for the group failure will result in different service availability, hence different recovery efficiency.

Among the existed literatures, in [11] the authors describe a hierarchical classification of two-link failures in optical networks and use the vulnerability metric to evaluate the effects of different identified failure models. The results in [12] show that the protection switching efficiency can be improved by reconfiguration in multiple failure scenarios, especially the shared path protection scheme gains high improvement against two-link failures in vulnerability measure by on-line reconfiguration which only sacrifices less extra capacity cost. The second link failure is assumed to occur long enough after the first to allow normal recovery to complete but before any physical repair can be accomplished. In [13] there presents three lookback strategies to recover from two-link failures by considering the relationship among protection paths for distinct links belonging to some working paths. A protection path computation model is also proposed for achieving 100% restorability there. In [14] the author considered the influence on p-cycle selection and the restorability to dual fiber duct failures, especially the optimal capacity design to accommodate two failures located within the p-cycle. It is discouraged if applying the schemes developed above to the group failure condition directly due to the difference in natural feature between there two types of target failure models. Extensive consideration and additional novel protection/restoration methods for group failure model are desired to be proposed.

2 An Example

We give an example in Fig. 1 to illustrate the protection switching concepts. Assuming that there is a connection request between *(s,d)* = (A,J). After setup process the working path for this request is determined to A-D-G-H-J. The restoration rerouting paths corresponding to each protection scenario are listed in table 1. The protection path in path protection scheme is chosen by shortest node-disjoint path. The alternative rerouting path in link protection scheme is determined link by link in the same manner. The segment restoration paths are found by equal partitioned on the working length and assumed that the shared protection is allowed. In the sub-mesh protection strategy, the protection group is assigned according to current network status dynamically and assumed to node set {A,D,G,H,J,I,E,B}. If the group failure occurred after the bomb attack covers the node set (D,G,E,H), that is, the affected network nodes associated with their attached links are all destroyed and loss of functionality, then the restoration paths on all the schemes above are unable to recover the lost traffics. In path protection, the group failures cause the multiple failure occurrences in both disjoint route due to the node set in the working/protection path pair overlapped with the node set of the group failure. The same reason to the unavailability also can apply to other cases of protection methods. The link protection with dedicated disjoint route each link maybe has more chance to avoid the failure clustering effect but the shared protection is applied in general because of limited resource on network capacity and topology. The segment protection also failed in this case even with different partition scenario on the working path. Finally, the sub-mesh protection will fail to recover from the failure if the group set still remains as fixed, but the restoration process can be activated and completed if the group merging procedure proposed in [15] is with another distinct protection group (B,E,I,K,L,F,C). That is , the restoration path after the group merging will be A-B-C-F-L-K-I-J, which is another node disjoint path with the working path.

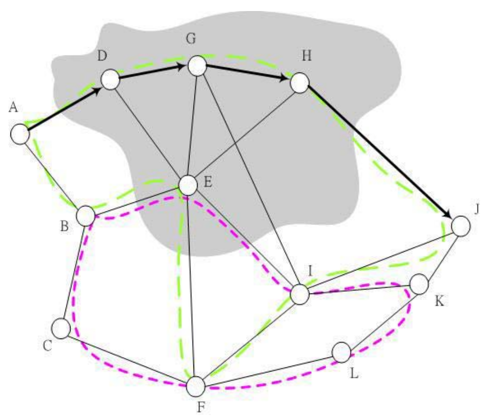


Fig. 1. Illustration of the group-based protection mechanism

Table 1. Protection paths in different protection switching scenarios

Normal working path	(A,D,G,H,J)
Path protection path	(A,B,E,I,J)
Link protection path	$(A,B,E,D)_1,(D,E,G)_2,(G,E,H)_3,(H,E,I,J)_4$
Segment protection path	$(A,B,E,G)_1,(G,E,I,J)_2$
Sub-mesh protection	(A,B,E,F,I,K,J)

The rest of this paper is organized as follows. In section 3 we describe the heuristics on static and dynamic group partition schemes and related algorithms. The results of the simulation experiments are discussed in Section 4, and Section 5 summarizes this paper.

3 Dynamic Group Protection for Group Failures

It is intuitive to solve the group failure survivable routing problem by selecting the maximum separating disjoint path pair for each reliable connection request. We define the diversion degree to be the distance of separation between two lightpaths, which can be evaluated by counting the node-pair distance for every node located on these two paths level by level. For speeding up the calculation process, the distance table which can be incorporated into the adjacency matrix of the network model will be provided. The RWA optimization procedure can de completed with 2-step way, that is, first finding the maximum disjoint routes which meet the diversion degree constraints, and then the capacity optimization for shared protection will be performed. The shared protection mechanism will be default utilized in our model by efficiency and realistic consideration. We omit the formulations of LP model for the group failure protection based on a group protection scenario here due to the space limitation.

The node-disjoint path pair finding under multiple constraints is difficult and the linear programming version solution for group failure is computational inefficient. Accordingly, the heuristics to solve the RWA problem for a connection request, especially by the group partition scenario, will be appropriate in this case. The group partition can be implemented by the static or dynamic manner. The basic idea behind the static partition is to divide the network topology into several sub-areas (regions) by geographical classification. The physical distances and the crucial city points will be incorporated into consideration together. (The traffic flows passing cities which are vulnerable to be attacked will be rerouted by the politic decision, although it is not the actual case in practical.) The routing of working path will be made crossing multiple regions and partitioned into working segments automatically. Each segment in a region will use the region boundary on both sides of the segment as the candidate working and protection path pair. The RWA process continues to examine if these path pair meet the capacity and wavelength requirement. The path finding process will try to discover the alternate route within the region or region-by-region. We illustrate the case of static route selection in Fig. 2. The path segment A-D-F of working path A-D-F-G will be contoured by static group partition A-B-H-I-J-F and A-E-F. There provides some degree of survivability due to their separation in geographical. The traffic flows will be distributed and delivered among these two spare paths according to free capacity ratio or inject all the flows to one of these paths depending on the capacity availability. The protection path A-C-I-J-F will be examined if there doesn't have enough resource available on A-B-H-I-J-F. It is intuitively that the region determination step can be activated from the smallest face of graph and expanding incrementally. Or, the partitions of the network can be initialized by choosing the OXCs which equipped with the wavelength conversion capability as the boundary of each partition due to more efficient resource availability. The LP formulations for the static RWA calculation are not presented here because of the space limitation.

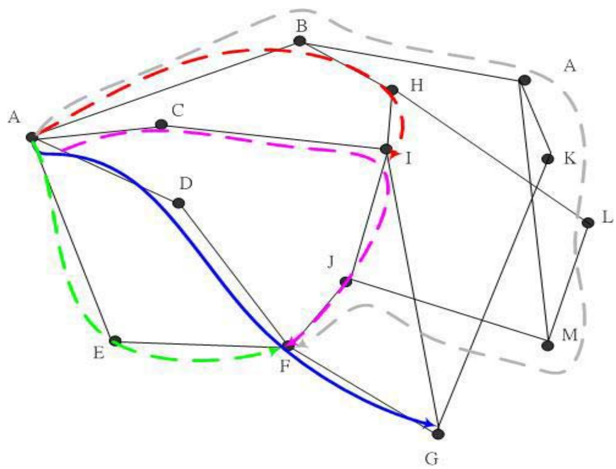


Fig. 2. Static route selection for group failure recovery

In dynamic group protection scenario [15], the protection group are created and adjusted dynamically according to the network resource and current flow status. There is no spare resource pre-configuration and the resource information is exchanged on the fly. If a working path passes through a dynamic group, the protection contour which is created by preceding connection will be used as the backup route for the current working path or segment. The backup route will change on some links because of the execution of capacity adjustment process (the expansion or shrinking process described in [15]). The protection groups can be shared or not. On survivable routing for dynamic traffic flows under group failure situation, the route selection will depend on the current group status in the network to choose the maximum disjoint group route pair as the working and protection paths. The group distance and group member information can be afforded by dynamic status exchange by modifying the control protocol proposed in [15]. The RWA procedure can be made in joint or separate two-way manner after the routes determination which according to the guidance rules as follows.

- The maximum node-disjoint routing algorithm will be executed if there is no group exists in the network.
- The protection path will be chosen along the boundary of the opposite side of the group the group if the working path of the connection is allocated along one side of the group.
- The maximum group-disjoint routing algorithm will be activated if the route pair must pass through two or more group in the network.

One can realize how the partitions are performed will determine the diversity of the disjoint path pair and the restorability. The rules described above attempt to achieve the maximum possibility to avoid and escape from the influence under group failure condition.

4 Simulation Study

We evaluate the effectiveness of the proposed static and dynamic group routing algorithms by performing the simulations. We list the characteristics of simulated networks in Table 2. Each link in the networks is assumed to be a bi-directional duplex channel consists of 16 wavelength channels. The connection requests arrive at a node as a Poisson process with exponentially distributed holding time with unit mean. We use 4 metrics, connection blocking probability, restoration time, restorability by recovery ratio and influence of group sharing to measure the performance of the proposed algorithms. The blocking probability is evaluated by the success ratio that the spare path could be designated during working path establishment phase. We also distinguish the blocking probability as 2 components in which one resulted from the routing failure and the other resulted from the failure of wavelength assignment.

Table 2. Sample network topology information and statistical simulation results

Target network	Node number	Edge number	Average nodal degree	Average group size
(a) Germany	73	130	3.56	9
(b) French	122	214	3.51	7
(c) Poland	47	70	2.98	12
(d) Spain	40	60	3.0	8

We plot the blocking probability versus loading factor (in Erlang unit) in Fig. 3(a). About 82% of the blocking from the 76% acceptance ratio produced by routing failure under heavy loads, which implies the maximum distance disjoint routing tends to be impossible on the bound of large distance guarantee to attack involving larger range of area. The automatic quality of protection for reliable service guarantee can be provided if we change the routing policy to be in the adaptive manner. The limitations on group size and path length will further degrade the probability. For verifying the efficiency of restoration, we apply 4-types of failures with different cluster size which covers from 1 to 5 nodes. From the plotting of restorability analysis in Fig. 3(b) and

3(c), the restoration time grows less than exponentially because of the balance of dynamic resource releasing and bounded group boundary size. Advanced modification on group selection policy and control protocol will improve the recovery efficiency. The influence on group sharing is plotted in Fig. 3(d), we note that the sharing degree will decrease the acceptance and the restorability performance little bit due to the lower separation degree and more switching overhead. The degree of sharing on different groups will be a tradeoff between the resource utilization and survivability when design the survivable RWA algorithms.

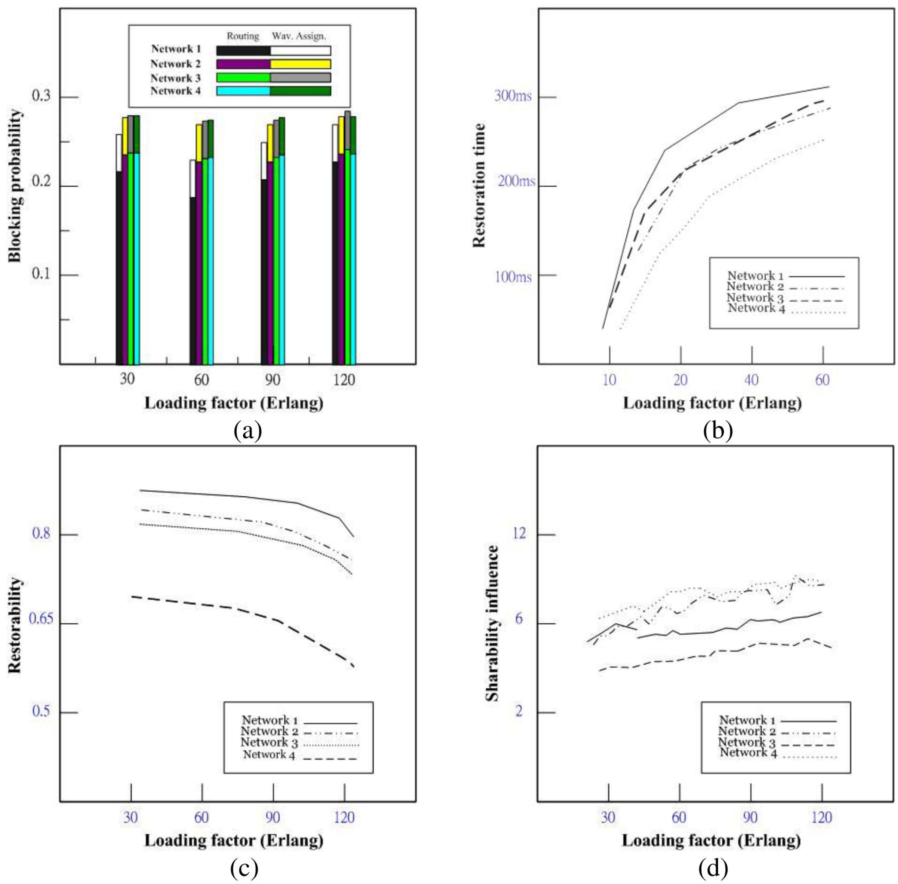


Fig. 3. Various simulation parameters vs. loading factors

5 Conclusion

This paper studied the protection and restoration mechanisms for the proposed new group failure model which mainly based on the group partition strategy to the WDM networks. The formulations of linear programming for survivable routing have also been developed. For the dynamic traffic demands, the adaptive protection group partitions specified by available network capacity and traffic patterns is more flexible and can be properly allocated for the group failure conditions. We extended the sub-mesh restoration strategy with fixed size and variable partition to achieve maximum distance disjoint routing to recover from the fixed range of clustering failure sets. The simulation results reveal that the group failure indeed have more influence on service availability if there lacks of deeper considerations for it, and the group partition policy will affect the survivability which can help to speed up the restoration process. The adaptive routing algorithms are under developed to achieve cost-efficient RWA operation for reliable service connection demands.

References

1. H. Zang, J. P. Jue, and B. Mukherjee: A review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks, Optical Networks Magazine, Vol. 1, No. 1 (2000) 47–60
2. S.Ramamurthy and B. Mukherjee: Survivable WDM mesh networks, Part II–Restoration, Proc. ICC, Vol. 3 (1999) 2023–2030
3. S. Ramamurthy and B. Mukherjee: Survivable WDM mesh networks, Part I–protection, Proc. IEEE INFOCOM, (1999) 744–751
4. B. T. Doshi, S. Dravida, P. Harshavardhana, O. Hauser, and Y. Wang: Optical network design and restoration, Bell Labs. Tech. J., Jan (1999) 58–84
5. G. Mohan, Arun K. Somani: Routing Dependable Connections with Specified Failure Restoration Guarantees in WDM Networks, Proc. IEEE INFOCOM, Vol. 3 (2000) 1761–1770
6. V. Anand, et al.: Sub-path protection: a new framework for optical layer survivability and its quantitative evaluation, UB CSE Tech. Report, Jan (2002)
7. C. Qu, et al.: Sub-path protection for scalability and fast recovery in WDM mesh networks,
8. Proc. OFC'02', Anaheim, CA, Mar (2002), 495–497
9. P. H. Ho and T. M. Hussein: A framework for service-guaranteed shared protection in WDM mesh networks, IEEE Communication Magazine, Feb (2002), 97–103
10. M. Medard, S. G. Finn and R. A. Barry: WDM loop back recovery in mesh networks, IEEE INFOCOM, (1999), 744–751
11. W. D. Grover: Distributed restoration of the transport network, Telecommunications Network Management into the 21st Century, NJ: IEEE Press,(1994) 337–417
12. S. S. Lumetta, M. Medard: Classification of two-link failures for all-optical networks, IEEE INFOCOM, (1999), 744–751
13. Sun-il Kim, S. S. Lumetta: Evaluation of protection reconfiguration for multiple failures in optical networks, IEEE INFOCOM, (1999), 744–751
14. Hongsik Choi, S. Subramaniam and Hyeong-Ah Choi: On double-link failure recovery in WDM optical networks, IEEE Communication Magazine, Feb (2002), 97–103
15. D. A. Schupke: The tradeoff between the number of deployed p-cycles and the survivability to dual fiber duct failures, IEEE Communication Magazine, Feb (2003)
16. C. S. Ho, I. Y. Chen and S. Y. Kuo: Improvements on dynamic sub-mesh restoration scheme in dense WDM networks, ICOIN'04, Busan, Korea, Feb (2004)

Wavelength Assignment in Route-Fixed Optical WDM Ring by a Branch-and-Price Algorithm*

Heesang Lee[1], Yun Bae Kim[2], Seung J. Noh[3], and Sun Hur[4]

[1] Sungkyunkwan University, Suwon, Korea
leehee@skku.edu
[2] Sungkyunkwan University, Suwon, Korea
[3] Myoungji University, Seoul, Korea
[4] Hanyang University, Ansan, Korea

Abstract. This paper addresses the wavelength assignment problem (WAP) in optical wavelength division multiplexed (WDM) telecommunication networks. We show that, even though WAP on optical ring topology belongs to NP-hard, WAP can be exactly solvable in practical size optical WDM rings for current and future traffic demand. To accomplish this, we convert WAP to the vertex coloring problem of the related graph and choose a special integer programming formulation for the vertex coloring problem. We develop a branch-and-price algorithm for the problem and carry out the performance comparison of the suggested algorithm with a well-known heuristics.

1 Introduction

The fast growth of the Internet and new applications such as electronic commerce, high-speed internet access, and video-on-demand services have created an ever-increasing demand for greater bandwidth in telecommunication networks. A cost effective way to deliver high speed services is to send multiple wavelengths through a single optical fiber using wavelength division multiplexing (WDM) technologies. Therefore WDM transmission systems and related technologies are being developed and deployed for the optical backbone networks.

A pair of WDM transmission nodes enables the establishment of all-optical WDM channels, called *lightpaths*. A lightpath connection goes through several intermediate WDM nodes but two lightpaths must not use the same wavelength on a given link[1] [1].

To deploy economical WDM transmission networks, "routing each lightpath" that needs to be established on the network (*routing problem: RP*), and "assigning wavelengths to these lightpaths" satisfying the wavelength continuity

* This work was supported by grant No.R01-2004-000-10948-0 from the Basic Research Program of the Korea Science & Engineering Foundation.

[1] This technical requirement is called the *wavelength-continuity constraint.*

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 194–203, 2005.

constraint (*wavelength assignment problem: WAP*) using WDM add-drop multiplexers (ADM's) in a optical ring topology are two important research topics [2]. In this study we assume to consider only one favorable directional (e.g. clockwise in this paper) path for lightpath routing in the WDM ADM ring networks. In this "route-fixed" WDM ring networks, the main network design problem to consider is WAP in a uni-directed optical ring graph without considering RP. On the contrary, in "route-unfixed" WDM ring networks, where lightpath for a demand pair can be established using clockwise or counterclockwise routes in a bi-directed optical ring network of two fiber cables, both RP and WAP must be optimized [3].

In this paper, we focus WAP in route-fixed ring to minimize the number of the used wavelengths in "uni-directed ring". We try to minimize the number of the used wavelengths because efficient wavelength assignment is considered as the most important network planning object to design optical WDM ring networks.

The remainder of this paper is organized as follows. In Sect. 2, we study relationship between WAP and a coloring problem on a class of graphs, circular-arc graphs. In Sect. 3, we propose an integer programming formulation for the coloring problem and an exact solution algorithm for the formulation. In Sect. 4, we study generic heuristics and some implementing strategies. In Sect. 5, we show, by computational experiments, that the suggested exact algorithm can find optimal solutions for moderate-sized WDM ring networks. We also carry out performance comparison of our algorithm with the generic heuristics and a known branch-and-bound algorithm suggested in the literature. In Sect. 6, we give some concluding remarks and discuss further study topics.

2 Wavelength Assignment in Route-Fixed Ring

Wavelength assignment on each route-fixed lightpath in a WDM network can be interpreted as the coloring problem of the "(undirected) paths" where all paths going trough "a link" of the WDM network should have different colors, where each color represents a wavelength. When the physical topology of the WDM network is the ring, lightpaths around the uni-directed optical ring can be viewed as a collection of (undirected) arc paths on a circle [4]. Hence we can convert the *wavelength assignment problem on lightpaths* for this WDM ADM ring network into a *vertex coloring problem* by constructing the following *coloring graph* [4].

For each lightpath of the original optical WDM ADM ring network, we define a vertex in the coloring graph, and define an edge between two vertices of the coloring graph if the associated two lightpaths overlap at any link of the original optical WDM ADM ring network.

This conversion technique can convert a path coloring problem of not only a WDM ring network but also of an arbitrary WDM network into a vertex coloring problem of the related coloring graph. When the original WDM network is a ring, the coloring graph is a special class of graphs so called *circular arc graph (CAG).* Hence WAP on a route-fixed WDM ring network is equivalent to the optimal vertex coloring problem on the associated CAG.

WAP in route-fixed ring network is NP-hard since the optimap vertex coloring problem in CAG belongs to NP-hard [5]. Hence many previous works in the literature [3], [6], [7], have focused on the development of not an "exact" algorithm but an "approximation" algorithm for minimizing wavelength assignment on route-fixed WDM rings.

With a difficulty of NP-hardness of the vertex coloring in CAG, we encounter another problem for using this conversion technique: Even for WDM ring networks with small number of ADM's, the coloring graph is large. For example if WDM ring has 10 ADM's and 50% of all ADM demand pairs requires a lightpath, we have 10 x 9 x 50% = 45 lightpaths in the WDM network. This needs 45 vertices and several hundred edges in the related coloring graph. Hence "usual" algorithmic techniques for vertex coloring may not guarantee to get an optimal solution for WAP of WDM ring network for even small number of ADM nodes and lightpath demands.

3 Suggested Optimization Algorithm

Let $G = (V, E)$ be an undirected coloring graph, with V, the set of vertices, and E, the set of edges. A simple and canonical integer programming (IP) formulation for the vertex coloring problem is possible by defining binary decision variable $y_{ik} = 1$ if vertex i is assigned color k and $y_{ik} = 0$ otherwise. However, this canonical IP formulation has critical disadvantages since its linear programming (LP) relaxation is extremely fractional and it has "symmetry" property of the variables. Here the symmetry means that the variables for each color k appear in exactly the same way for deciding number of colors. The symmetry property also makes difficult to enforce integrality in one variable without problems showing up in the other variables because any solution to the LP relaxation has an exponential number of representations (as a function of the number of colors). For these reasons we suggest the following IP formulation that has a very strong LP relaxation without the symmetry property.

An *independent set* of G is the set of vertices such that there is no edge in E connecting any pair of vertices of the independent set. A *maximal independent set (MIS)* is an independent set that is not included in any other independent set. Let S be the set of all MIS's of G and the binary variable $x_s = 1$ if MIS s is chosen, while $x_s = 0$ otherwise. Then the vertex coloring problem on a CAG (or WAP for route-fixed ring) can be formulated as the following IP problem so called (MIS IP) that is to find the minimum number of MIS's where each vertex of G is covered by at least one MIS.

(MIS IP)

$$Minimize \sum_s x_s \tag{1}$$

$$subject\ to \sum_{\{s:i \in s\}} x_s \ \geq\ 1,\ \forall\, i \in V, \tag{2}$$

$$x_s\ \in\ \{0,\ 1\},\ \forall\, s \in S. \tag{3}$$

The objective function (1) is to minimize the number of used MIS's, and the constraints (2) and (3) imply that each vertex of G must be included in at least one MIS. The number of used colors is the same with the number of chosen MIS's since we assign the unique color for all vertices in an MIS. The same color can not be used for the adjacent vertices since there exists no edge between any pair of nodes in a chosen MIS. Note that a feasible solution to this IP may assign multiple colors to a vertex since each constraint has the condition of *at least* 1 instead of the condition of *equal to* 1. This multiple color possibility can be corrected by using arbitrary one color of the multiple possible colors to a vertex.

(MIS IP) has only one constraint for each vertex and without symmetry property for the decision variables. The number of decision variables is, however, huge since the number of all MIS's for a graph can be an exponential function of the number of edges of the coloring graph. Therefore generating all MIS's for a coloring graph to get the explicit formulation is intractable. Hence solving even the LP relaxation of (MIS IP) may be computationally difficult if we use the explicit formulation. We resolve this difficulty by using only subset of the variables and "generating" more variables and their incidence information when they are needed. This technique, called *column generation*, is well known for LP with many variables (see [8] for details). We also develop a branch-and-price algorithm for (MIS IP).[2] The general idea of the column generation procedure for LP is based on the fact that an optimal solution to LP with many columns can be obtained without explicitly including all columns. The column generation technique has recently emerged as an effective technique for many NP-hard IP problems since in many cases a column generation formulation of an IP has a stronger LP relaxation than a canonical compact IP formulation [8].

In our problem, the column generation technique for the LP relaxation of (MIS IP) is described as follows: Begin with $\bar{S}$, a subset of S, the set of all MIS's. Solve the LP relaxation of (MIS IP) restricted to all $s \in \bar{S}$. From a feasible solution for the LP relaxation and a dual value π_i of the dual LP problem for each constraint i of the primal LP relaxation, determine if we need more columns. From the LP duality theory (see [9] for details of the LP duality theory), in order to check if we need more columns, we need to find an MIS of total weight is greater than 1 where each vertex i has weight π_i. Hence the column generation for LP relaxation of (MIS IP) can be done by solving the following subproblem.

(Column Generation Decision Subproblem)

$$maximize \sum_{i \in V} \pi_i \; z_i \tag{4}$$

$$subject\ to\ z_i \ + \ z_j \ \leq \ for\ all\ (i,j) \in E, \tag{5}$$

$$z_i \in \{0,1\}\ for\ all\ i \in V. \tag{6}$$

Note that this problem is to find the maximum weight MIS on CAG, when the non-negative weights π_i are given for every vertex i [10]. If the optimal

---

[2] A branch-and-price algorithm is defined as a column generation algorithm embedded in a branch-and-bound algorithm.

objective function value to this problem is greater than 1, then the z_i with value 1 correspond to the vertices that constitute an MIS that should be added to $\bar{S}$. In this case, the column generation process is repeated until the optimal objective function value of (5) is not greater than 1. If this case happens, then there exists no improving MIS: Solving the LP relaxation of (MIS IP) over this $\bar{S}$ is the same as solving the LP relaxation of (MIS IP) over S.

The complexity of this column generation decision subproblem may greatly affect the solution time of the LP relaxation of (MIS IP). Fortunately, we can prove that the maximum weight MIS Problem in CAG can be solved in polynomial time due to our following results: The maximum weight MIS problem in the *interval graph* can be solved in polynomial time since there exists an $O(nlogn)$ algorithm for this problem [11], where n is the total number of vertices of the graph. Using this algorithm at most n times, we can solve the maximum weight MIS problem on CAG in polynomial time by decomposing a CAG into at most n problems as follows:

Each of decomposed problem is to find a maximum weight *independent lightpath set* that is a subset of lightpaths in the original ring network without an overlapping link between any pair of chosen lightpaths of the subset. To get an independent lighpath set, choose a lightpath k, then delete all lightpaths that are overlapping with lightpath k. The remaining lightpaths can be converted to a coloring graph that is an interval graph. Find a maximum weight independent lightpath set via the corresponding maximum weight MIS in the coloring graph that is an interval graph. Augment the maximum weight independent lightpath set of the interval graph with the path k for a candidate of a maximum weight independent lightpath set of CAG. Repeat this procedure for every lightpath $k \in V$ and choose the maximum weight independent lightpath set among at most n candidates. Hence the maximum weight MIS problem for a CAG can be solved in $O(n^2logn)$ time.

```
(Algorithm for LP relaxation of (MIS IP))

1. Start.
2. Do {
  (1) Select any lightpath, say lightpath k, among previously
      unselected ones.
  (2) Delete all the lightpaths that overlaps with lightpath k.
  (3) Sort the end point of remaining lightpaths in non-increasing
      order.
  (4) Solve the maximum weight MIS problem in the corresponding
      interval graph.
3. } until (all lightpaths considered).
4. Compare candidate MIS's and obtain the maximum MIS for CAG.
5. If the optimal function value of MWIS problem is more than 1,
   then an MIS corresponding z_i with value 1 is added to (MIS IP).
   Go to Start.
6. Otherwise, stop. No more column is needed.
```

When the column generation process is finished if the resulting solution to the LP relaxation of (MIS IP) over $\bar{S}$ has an "integer" solution, then the corresponding solution is an integer optimal solution for (MIS IP) over S. When some of the variables of the LP optimal solution, however, are not integer, we need to enforce integrality for those variables. We use a branch-and-bound procedure to enforce integrality of the variables. One important thing of the column generation within a branch-and-bound algorithm is that the column generation problem should still not difficult after branching. In our problem by choosing a pair of "minimal distance non-overlapping" lightpaths for branching, the column generation is maintained as the maximum weight MIS problem in a modified graph that maintains CAG property.

4 Heuristics

Heuristic can be used for an NP-hard optimization problem. FirstFit Heuristic studied in the literature is a generic class of heuristics for assigning a wavelength to a lightpath using some fitness measure. It assumes that the wavelengths are labelled $1, \cdots, W$. then choose a lightpath using some lightpath selection rule that represents a fitness measure. We assign an available wavelength of the lowest label to the selected lightpath as an implementation rule. In terms of coloring, it sequentially chooses a vertex of the coloring graph and colors the vertex with the available color with the lowest label for each lightpath. We propose three heuristics by using some known lightpath selection rules studied in the literature as follows: Longest-lightpath-first Heuristic (LPH), Shortest-lightpath-first Heuristic (SPH), and Maximum-degree-first Heuristic (DegH). The generic FirstFit Heuristic is described as the following: The LPH, SPH and DegH heuristics can be easily described by defining the lighpath selection rule of this generic FirstFit Heuristic.

`(Procedure of FirstFit Heuristic)`

1. Do {
   (1) `Select a first fit lightpath unassigned yet according to the suggested lightpath selection rule.`
   (2) `Assign the lowest label from the set of available wavelengths.`
2. } `until (all lightpaths are assigned).`

LPH is based on the fact that the longer the lightpath is, the more it is likely to overlap with other lightpaths. Moreover the longer lightpath is hard to have colors when the number of available colors is small since the possible color should be different from the other lightpath in every link in the long paths. Note also that a vertex with the maximum degree in coloring graph corresponds to the lightpath overlaps with maximum numbers of other lightpaths in the original ring network. Note that a lightpath that overlaps with many other lightpaths is likely to have few alternatives to reuse the wavelengths already assigned. Therefore, LPH and DegH seem intuitively to perform better than SPH since the more flexible coloring is possible for LPH and DegH considering the degree of freedom for given wavelength assignment.

5 Computational Experiments

The proposed algorithm for (MIS IP) formulation of WAP has been coded in C and experimented on a SUN Sparc Ultra workstation using an IP optimization callable library, CPLEX. By the experiments, we want to prove that the suggested algorithm is computationally feasible to implement in real-sized WDM rings. We also want to compare the suggested algorithm with the generic heuristics studied in Sect. 4 for the vertex coloring problem [12].

We experiment five classes of problem instances for WDM networks that have 5, 10, 15, 20, and 25 WDM ADM ring nodes. To know the effect of demand on the ring to the performance, we divide each class of node sizes into four demand sets by setting the *demand density* of 0.3, 0.5, 0.7, and 0.9, which is defined as the probability that requires one lightpath for an ADM pair. For example, we have 25 x 24 x 0.9 = 540 lightpaths for 25 ADM ring nodes with demand density of 0.9. We experiment five instances for each of twenty sets, total 100 instances. In Table 1, input parameters of twenty sets are summarized. In Table 1, $G(n, d)$ represents that n, is the number of ADM's from 5 to 25, and d is the demand density from 0.3 to 0.9. "Verts" and "Edges" in Table 1 denotes respectively the

Table 1. Problem inputs

set	$G(n,d)$	n	Verts	Edges
1	$G(5,0.3)$	5	6	10.6
2	$G(5,0.5)$	5	10	30.6
3	$G(5,0.7)$	5	14	67.2
4	$G(5,0.9)$	5	18	113.4
5	$G(10,0.3)$	10	27	285.0
6	$G(10,0.5)$	10	45	790.8
7	$G(10,0.7)$	10	62	1490.0
8	$G(10,0.9)$	10	80	2503.0
9	$G(15,0.3)$	15	63	1567.8
10	$G(15,0.5)$	15	105	4382.4
11	$G(15,0.7)$	15	147	8634.8
12	$G(15,0.9)$	15	188	14082.0
13	$G(20,0.3)$	20	114	5200.0
14	$G(20,0.5)$	20	190	14507.8
15	$G(20,0.7)$	20	266	28619.6
16	$G(20,0.9)$	20	341	47461.8
17	$G(25,0.3)$	25	180	13154.6
18	$G(25,0.5)$	25	300	36702.0
19	$G(25,0.7)$	25	420	72262.8
20	$G(25,0.9)$	25	516	118979.2

Table 2. Average performance

set	Col	BandB	LB	LP	Opt	Heur
1	0.8	0.0	4.0	4.2	4.2	4.2
2	1.4	0.0	6.2	6.2	6.2	6.2
3	5.6	0.0	8.2	8.2	8.2	8.8
4	10.8	0.0	9.6	9.6	9.6	10.6
5	8.8	0.6	17.2	17.6	17.6	18.2
6	34.2	0.4	25.4	26.2	26.2	27.8
7	77.6	0.4	33.2	33.6	33.6	37.2
8	127.6	0.0	41.8	41.8	41.8	47.2
9	57.6	3.6	36.2	36.8	36.8	39.4
10	145.8	1.8	56.6	58.2	58.2	63.2
11	269.4	4.8	77.6	78.4	78.4	86.4
12	418.2	3.4	96.6	96.6	97.0	10.2
13	166.4	0.8	60.8	61.6	61.8	67.4
14	324.0	4.8	102.0	103.2	103.4	111.2
15	593.8	4.6	139.0	140.8	141.2	155.0
16	962.8	12.0	175.0	176.2	176.2	197.8
17	293.2	1.8	97.0	99.5	100.0	108.4
18	613.6	16.6	160.0	160.7	161.0	174.6
19	993.4	19.2	218.8	220.8	220.8	243.4
20	1798.0	15.4	274.0	276.6	276.6	310.0

number of vertices and the average number of edges in the coloring graph. Note that Verts is the same in a given set but Edges of five instances of a given set can be different since each instance can have different overlapping conditions even with the same number of lightpaths.

Average performances of five instances for each set for the experiment are displayed in Table 2. In the table "Col" denotes the average of five instances for the total number of columns generated during the column generation procedure. "BandB" denotes the average of the number of the branch-and-bound tree nodes to get the final integer solution from the optimal solution of the LP relaxation. "LB" denotes the average of the *maximum load* of links that is defined as the maximum value of the number of paths in a link of the original WDM network, which is a lower bound of the minimum number of wavelengths for the WDM network. "LP" denotes the average of the optimal objective value of the LP relaxation of (MIS IP) and "Opt" denotes the average of the minimum number of wavelengths finally obtained by the suggested branch-and-price algorithm. "Heur" denotes the average of the number of wavelengths obtained by LPH.[3]

As we see in Table 2, our column generation procedure does not require generating huge number of columns to get an optimal solution of the LP relaxation of (MIS IP). It means that after getting an LP optimal solution the

[3] Comparing the solution qualities, LPH is more effective than SPH about 5% and do not show significant difference with DegH, in our computational experiments.

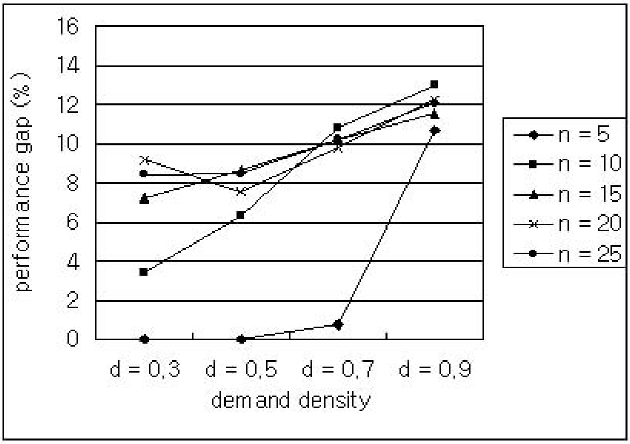


Fig. 1. Solution quality gap of LPH heuristic from the optimum value

algorithm can be terminated with an optimal solution after traversing not so many branch-and-bound nodes. This can be possible since (MIS IP) has a very strong LP relaxation as we see in Table 2. Note that the duality gap, the difference between the "LP" and "Opt" is only 0.5% average for 100 instances. In Table 2, we can also know that LB, the maximum load of links is a very good lower bound for the minimum number of wavelengths when a WDM network has a ring topology. For small size networks, this lower bound can happen to be equal to the minimum number of wavelengths while for large size networks it has a difference from the minimum number of wavelengths about 6.6% in the worst case of 100 instances.

The suggested branch-and-price algorithm of (MIS IP) shows better solution quality than the First-Fit Heuristics. The LPH heuristic has the solution quality gap between 0% to 13% from the optimum values obtained in Table 2. Fig. 1 shows the average solution quality gap between (MIS IP) and LPH for four different demand densities of n = 5, 10, 15, 20, and 25. This figure shows that as the number of lightpaths to be established increases, the solution quality gap of the heuristic gets larger.

WAP in the route-fixed ring is an NP-hard problem. We can, however, solve the problem not to an approximation solution but an optimal solution for up to the instances of 25 WDM ADM nodes, 516 lightpaths, and average 118,979 edges in a few minutes. We think our experiment covers sufficiently large networks for practical implementation of the current and near future demand that the current WDM technology can support.

6 Conclusions

In this paper, we have shown that WAP on route-fixed optical WDM ring can be exactly solvable up to 25 WDM ADM rings that can be enough size for near future demand of WDM ring networks. To accomplish this, we proposed MIS based IP formulation for the vertex coloring problem on CAG and developed a branch-and-price algorithm.

Optimization study for wavelength assignment and path routing in route-unfixed ring topology is one of the topics of further study [7]. Usual approaches for this problem have been based on decomposing the problem into RP and WAP independently and solving each problem iteratively. Optimization study for TDM over WDM is another new research topic [2], [13]. In this problem the major cost of a network is related not only to the number of wavelengths but also to the number of TDM interfaces. Our MIS covering formulation and column generation decision sub-problem can be applied or extended for these topics.

References

1. R. Ramaswami, and K. N. Sivarajan, "Optical Networks: A Practical Perspective", Morgan Kaufmann, 1998.
2. X. Yuan and A. Fulay, "Wavelength assignment to minimize the number of SONET ADMs in WDM Rings", Photonic Network Communications, 5(1), pp. 59-68, 2003.
3. G. Wilfong, and P. Winkler, "Ring routing in WDM networks", Proc. IEEE Infocom'99, 1999.
4. A. Tucker, "Coloring a family of circular arcs", SIAM J. Applied Mathematics, 29(3), pp. 493-502, 1975.
5. M. Garey, and D. S. Johnson, "The complexity of coloring circular arcs and chord", SIAM J. Algebraic Discrete Methods, 1 (2), pp. 216-227, 1980.
6. T. Erlebach, and K. Jansen, "The complexity of call-scheduling", Preprint, 1997.
7. R. Ramaswami, and K. N. Sivarajan, "Routing and wavelength assignment in all-optical networks", IEEE/ACM Trans. on Networking, pp.489-500, Oct, 1995.
8. J. Birge, and K. Murty eds. "Mathematical Programming: State of the Art 1994", University of Michigan, 1994.
9. G. Nemhauser, and L. Wolsey, "Integeger and Combinatorial Optimization", Wiley, 1988.
10. A. Mehrotra, and M. A. Trick, "A column generation approach for graph coloring", Preprint, 1995.
11. J. Y. Hsiao, C. Y. Tang, and R. S. Chang, "An efficient algorithms for finding a maximum weighted 2-independent set on interval graph", Information Processing Letters, n43, pp. 229-235, 1992.
12. D. Brélaz, "New methods to color the vertices of a graph", Communications of the ACM, 22, pp. 251-256, 1979.
13. L. Liu, X. Li, P. Wan, and O. Frieder, "Wavelength assignment in WDM rings to minimize SONET ADMs, Proc. IEEE INFOCOM'2000, vol. 2, pp. 1020-1025, 2000.

M-MIP: Extended Mobile IP to Maintain Multiple Connections to Overlapping Wireless Access Networks

Christer Åhlund[1], Robert Brännström[1], and Arkady Zaslavsky[2]

[1] Luleå University of Technology, Department of Computer Science,
SE-971 87 Luleå, Sweden
{christer.ahlund, robert.brannstrom}@ltu.se
[2] School of Computer Science & Software Engineering, Monash University,
900 Dandenong Road, Caulfield East,
Vic 3145, Melbourne, Australia
a.zaslavsky@csse.monash.edu.au

Abstract. In future wireless access networks, connectivity to wired infrastructure will be provided through multiple access points with possibly different capabilities and utilization. The demand for increased network performance requires the ability to predict the best overall performance of those access points and to switch access point when the performance changes. Then there is the demand for mobility between networks with maintained connectivity which requires the ability to switch the point of attachment. We propose Multihomed Mobile IP, enabling performance discovery at the networks layer and the capability to decide what AP to use. Mobile IP support is needed to allow mobile hosts to move between networks with maintained connectivity. Multihomed Mobile IP enables mobile hosts to register multiple care-of addresses at the home agent, to enhance the performance of wireless network connectivity. This article describes a simulator evaluation of multihomed Mobile IP.

1 Introduction

With increasing demands for wireless connectivity and mobility support, new solutions are required to maintain the wireless network connection and to optimize the performance. This is important for mobile hosts (MHs), both when moving and when stationary for a period of time. The major access technology used today in wireless local area networks (WLAN) is 802.11. The support of mobility and handover at the datalink layer enables flows to be maintained within the same network. However mobility between networks is not supported, since this would require handover at the network layer. For this, Mobile IP (MIP) [1]is proposed.

When combining wireless access (802.11) and network mobility (MIP), there are several things to consider. First, association is managed at the datalink level with no support from the network layer. An MH decides which AP to associate with based on the signal to noise ratio (SNR). The MH needs to associate to receive MIP agent advertisements used to discover available networks. If the MH discovers a foreign

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 204–213, 2005.

network (or if the MH arrives back to the home network), it requires a registration with the home agent (HA). Since the performance at the network layer may not be reflected in the SNR, the association may be with an AP having bad performance. With a high SNR metric the actual performance can still be low since an MH cannot sense collisions from other MHs using the same AP if it is out of communication range. Also, since the Network Allocation Vector (NAV) is used in 802.11, hosts will defer their communication and thereby avoid collisions. Therefore MIP cannot entirely rely on the datalink level to make the right decision about the selection of an AP. Instead network layer characteristics needs to be considered.

To enable this, performance discovery at the networks layer is required and the capability to decide what AP to use. This can be achieved with multihoming. Multihoming is enabled by using a single wireless network card switching between APs [2] or by using multiple network cards. By maintaining multiple network connections, network layer performances can be compared and the best one selected.

Handover can be classified into soft and hard handover. With soft handover the association with the old AP is sustained while associating with a new AP. In this ways two connections will be maintained for some time. With hard handover the connection to the old AP is ended before associating with a new AP.

In this paper we present an approach to multihoming with MIP, called M-MIP. With M-MIP, passive network-layer measurements are enabled by maintaining multiple registrations at the HA. In this way we can maintain connectivity and handle handovers without generating delays due to MIP registrations. M-MIP enables soft handover.

The paper is structured in the following way. Section 2 describes the architecture of M-MIP. Section 3 describes a simulation study and the results of the study. Section 4 describes related work and section 5 provides a concluding discussion.

2 M-MIP

This section briefly describes the changes made to MIP to enable multhoming functionality (M-MIP). For a more detailed description see [3]. M-MIP enhances the performance and reliability of MHs connections to WLANs. The multihoming functionality is managed by M-MIP and hidden from the IP routing process.

To register a care-of address at the HA, a registration request is sent by the MH. To enable the HA to distinguish between a non-multihomed and a multihomed registration, an N-flag is added to the registration request (see figure 1).

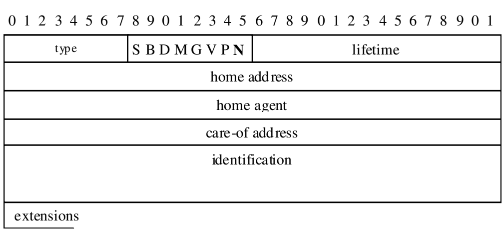


Fig. 1. The modified registration request message with the added N-flag

An HA receiving the registration request with an N-flag will keep the existing bindings for the MH. If a registration is received without the N-flag, the HA will clear the existing bindings for the MH which makes M-MIP compatible with standard MIP. One of the registered care-of addresses will be used to forward packets to the MH. To enable the selection at the HA, a metric is added as an extension in the registration request. The HA will maintain all registrations for an MH and based on the metrics it will install a tunnel into the forwarding table.

With a care-of address advertised by an FA, the MH is not allowed to use the Address Resolution Protocol (ARP). This will confuse other hosts connected to the network and may cause problems when the MH disconnects and moves to another network. To avoid this in MIP, the MH monitors the MAC address in the frame containing the agent advertisement, and installs the binding between the FA's MAC address and the IP address in the ARP table, for the FA registered with. When a packet is sent using the default gateway, an entry in the ARP table will already be available and no ARP request is needed. In M-MIP, the MH will maintain multiple registrations with different FAs as well as keep control of available FAs not registered with. All IP addresses for the FAs are installed in the forwarding table, and the bindings between the IP and the MAC addresses are installed in the ARP table.

To enable an MH to select the "best" AP to use, we evaluate the performance of an AP at the network layer. In M-MIP the MH keeps a list of all networks it receives valid advertisements from and registers the care-of address of the network(s) supporting the best connectivity, with respect to the throughput, at the HA. To evaluate the connectivity, the MH monitors the deviation in arrival times between MIP agent advertisements and makes a running variance metric (RVM) calculation based on this information (see formula 1).

$$\Delta t_{mean} = \frac{1}{n}\Delta t_n + \frac{n-1}{n}\Delta t_{prev\_mean} \qquad RVM_{new} = \frac{1}{n}(\Delta t_n - \Delta t_{mean})^2 + \frac{n-1}{n} * RVM_{prev} \tag{1}$$

The RVM is used to evaluate MHs wireless connectivity to foreign networks. A small RVM indicates that agent advertisements are received at discrete time intervals arrive without collisions and without being delayed by the FA. This indicates available bandwidth as well as the FA's capability to relay traffic for the MH.

The RVM is then added to the round trip time (RTT) between the MH and it's HA using formula 2.

$$\Delta RTT_{mean} = \frac{1}{n}\Delta RTT_n + \frac{n-1}{n}\Delta RTT_{prev\_mean} \qquad RNL = \Delta RTT_{mean} + RVM_{new} \tag{2}$$

This formula is defined as the Relative Network Load (RNL). The calculation is carried out at the MH and the metric is attached to the next registration request sent to the HA. The RTT measure is based on the registration messages sent between the MH and the HA.

In IP routing, with protocols like RIP [4] and OSPF [5], a wireless last hop link is not considered in the route calculation. A hop count of one is used in the RIP protocol, and a static link cost is used in OSPF. In M-MIP, IP routing is used towards the selected care-of address, but the selection of what care-of address to use is managed by M-MIP considering the wireless links.

The measurements and metric calculations are made prior to registration and maintained while being registered at foreign networks. Since the MH may register multiple foreign networks, the HA can have multiple bindings for an MH. Among the registered care-of addresses, the FA with the smallest RNL metric will be installed as the default gateway in the MH and as the selected care-of address at the HA.

With route optimization it is possible to choose a different FA (to communicate with the correspondent host) than the FA used to communicate through the HA. An MH (as in MIPv6) sends binding updates to the CH with available care-of addresses. By requesting the CH to respond to binding updates with an acknowledgement, RTT can be measured in the MH. We then have the same functionality between CHs and the MH with route optimization as the registrations between the MH and it's HA.

3 M-MIP Analysis Using RVM Based Simulation

In this section we present our work simulating M-MIP with the network simulator GlomoSim, version 2.4 [6]. The topology used is shown in figure 2.

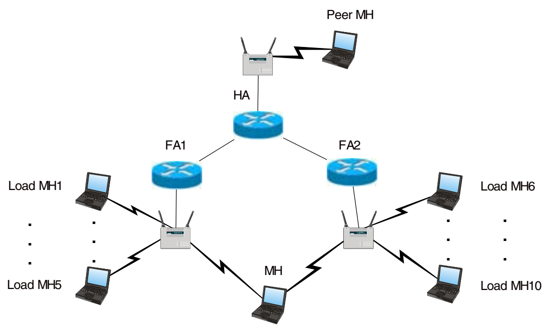


Fig. 2. The simulation topology

The simulation evaluates how well M-MIP discovers the utilization of APs and, based on this, selects the AP with the best network layer performance, considering the throughput.

Agent advertisements are sent every second and the MH registers every third advertisement with the HA. This is based on the MIP specification, where the timeout for a binding is three times the agent advertisement time. At each received advertisement the MH calculates the RNL metric and based on this decides which FA to use. The MH then attaches the RNL metric to the next registration request message.

The MH registers with two foreign agents (FA1 and FA2) using different channels and maintain multiple bindings with the HA. Hereby the HA as well as the MH maintain the RNL metric for each connection.

To add load to the wireless links we use the hosts LoadMH1 to LoadMH10 communicating with FA1 and FA2. We will use the phrase *load traffic* in the text below to name this traffic between the LoadMHs and the FAs. Based on the load traffic, we investigate how M-MIP responds to this load. The throughput presented in the graphs is the traffic sent by the peerMH and received at the MH, with and without using M-MIP. We name this traffic the *monitored traffic*.

Load traffic between peers is sent in both directions: the hosts LoadMH1 to LoadMH5 communicate with FA1 and LoadMH6 to LoadMH10 with FA2. The monitored traffic is also sent in both direction between the MH and the peerMH. Since the throughput presented looks similar in both the MH and the peerMH, we only present the monitored traffic for the MH.

Without using M-MIP, we evaluate the monitored traffic when the MH associates with an FA based on the SNR, without considering the performance at the network layer.

We use different combinations of traffic types (TCP and UDP) for the evaluation. For UDP traffic we use Constant Bit Rate (CBR) traffic and for TCP we use the generic File Transfer Protocol (FTP) provided by GlomoSim.

In our scenarios, the combination of traffic types for the load traffic and the monitored traffic is as follows:

- FTP is used as the load traffic and CBR as the monitored traffic
- CBR is used as the load traffic and FTP as the monitored traffic
- All hosts use FTP traffic.
- All hosts use CBR traffic.

We run each scenario with the two major packet sizes used in the Internet: 1500 bytes and 576 bytes [7,8]. Although another frequently used packet size is 40bytes (ACK packets in TCP), we do not look into this size.

In the graphs the solid line plots the throughput with M-MIP and the dashed line with a SNR-selected AP. In figures 3 to 6 the x-axis shows the number of LoadMHs generating load traffic. The y-axis shows the throughput of the monitored traffic received at the MH. The load traffic pattern is as follows: the first 10 seconds up to five LoadMHs add traffic to FA1; then 10 seconds to FA2. This is then repeated with a 20 second interval as well as a 30 second interval. The time to discover a loaded FA using the RNL calculation is about 2 seconds in all simulations.

The results are presented as mean values of multiple simulations (different seeds) and the error-bars express a 95% confidence-interval.

Figure 3 plots the result from the scenario where FTP is used as load traffic. Here traffic between the MH and the peerMH uses CBR traffic. The plotted solid green line is the throughput with a packet size of 1500 bytes using M-MIP. Behind the green line is a dotted blue line plotted showing the throughput with the SNR selected AP. The red lines show the throughput with a packet size of 576 bytes. Both the MH and the peerMH send 2.5Mpbs CBR traffic.

With an MTU of 576 bytes: less data in sent in each packet resulting in queuing at the sender with buffer overflow as a result. This occurs since there is a settling time for the interface, creating queuing with this packet size.

As expected, there is no difference between M-MIP and choosing the AP based on the SNR. The reason for this is that FTP (the TCP mechanism) degrades throughput caused by collisions, while CBR (UDP) continues sending at the same rate, forcing FTP to continue degrading its throughput.

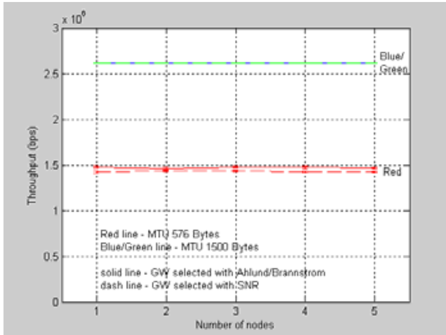
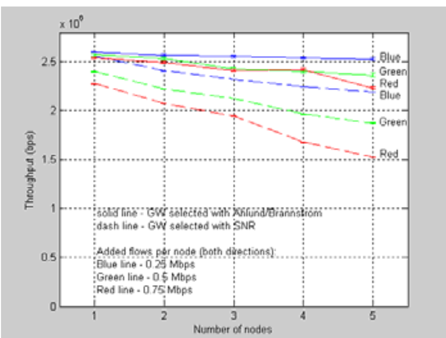
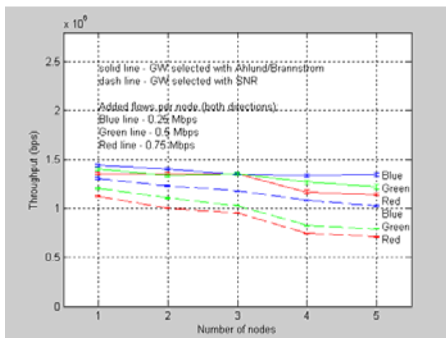


Fig. 3. CBR traffic received at MH with FTP traffic as load

In figure 4a we show the results where all hosts use CBR traffic with an MTU of 1500 bytes. The blue lines plot the monitored traffic when up to five LoadMHs generate load traffic of 0.25 Mbps. The green curves plot the same for load traffic of 0.5 Mbps and the red line for 0.75 Mbps. In figure 4b this is repeated for an MTU of 576 bytes.



(a)



(b)

Fig. 4. CBR traffic received at MH with CBR traffic as load with an MTU of 1500 bytes and 576 bytes

The results from the scenario where all hosts uses FTP traffic is plotted in figure 5. The throughput with a MTU of 1500 bytes and a MTU of 576 bytes shows the same results. FTP using an MTU of 1500 bytes is plotted by the blue line and the green line plots throughput with the MTU of 576 bytes.

The results from the last scenario are shown in figure 6, where CBR is used as the load traffic, and where monitored traffic uses FTP communication. In figure 6a, load traffic with a MTU of 1500 bytes are shown. The blue line plots the FTP traffic received at the MH with each LoadMH sending and receiving 0.25 Mbps. The green line plots the same with load traffic of 0.5 Mbps and the red line with load of 0.75 Mbps. In figure 6b this is repeated for an MTU of 576 bytes.

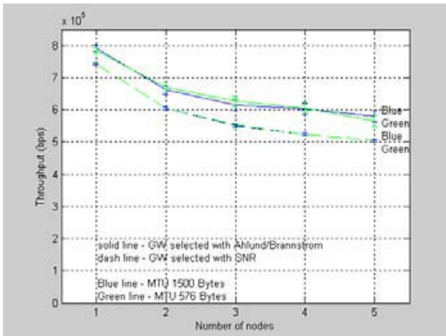
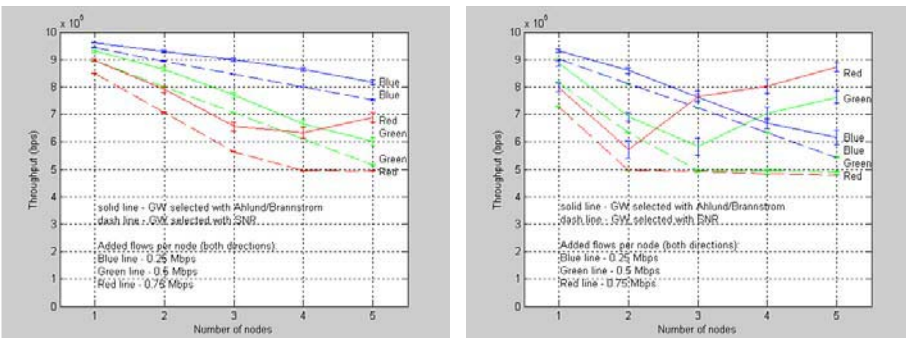


Fig. 5. FTP traffic received at MH with FTP traffic as load

In all scenarios M-MIP (plotted by solid lines) perform better than when only the SNR (dashed lines) is considered. An interesting observation from the last scenario (plotted in figure 6) is that the throughput increases with increased load as plotted in some of the curves.



(a) (b)

Fig. 6. FTP traffic received at MH with CBR load traffic using a MTU of 1500 bytes and 576 bytes

The reason for this is that we do not consider how traffic communicated by the MH affects the RNL. Before communication takes place the MH monitors the RVM and RTT and calculates the RNL metric. The RNL metric is sent to the HA in a registration request. Based on the metric a FA is selected. When communication takes place we continue to monitor the RVM and RTT and calculate the RNL metric. Since MHs own traffic affects the metric a new selection of FA may take place, selecting the FA being more loaded (not considering the own traffic). This will happen for both CBR and FTP traffic. With CBR traffic this happens if the MHs traffic increases beyond the difference between the least loaded FA and the next least loaded FA. With FTP, since TCP is used, the MH will take as much of the available link as possible, rendering a handover. This is most visible in the red curve in figure 6a and 6b. With a small difference in RNL, handover to the more loaded FA happens more often, keeping the sending window smaller. The same happens in all scenarios, but it is most visible in the last simulation. It also means that the performance of M-MIP will increase if we can avoid “false” handovers.

One solution to handle “false” handovers is for the MH to predict how much the own added flow increases the metric. However this is difficult. We are not able to say that X kbps effects the RNL metric with a value of Y. This depends on the utilization of the link, e.g. whether it is near congestion or not. Another option for the MH is to calculate the difference between the RNL metric after starting to send the own flow with the RNL metric before doing so. However the resulting metric may be in error. Let us say that another host begin communicating at the same time, the calculated difference will be too big. Or that a host that communicated stops, the calculated difference will be too small. A more straight forward solution is to make a decision when selecting the FA and starting to communicate. After that the FA cannot change for that flow. As soon as communication stops, new selections become possible. If all MHs behave in the same way we will have a distribution of MHs between APs.

In the case where route optimization is not used all traffic will use the selected FA. With route optimization multiple FAs may be used. This is possible since a unique binding update is sent to each CH.

4 Related Work

In MIPv4 [9] a proposal to multihoming is presented, sending one copy of a packet to each AP an MH is associated to. This means sending duplicated packets in the wireless media wasting scarce resources. In MIPv6 [10] there is no proposal for multiple bindings enabling multihoming with MIP.

MIP similar methods for handovers using IP multicasting are discussed in [11-13]. A multicast address is used to reach nearby APs in WLANs where the MH is located. An MH instructs one of the APs listening at the multicast address to forward packets to it, and the other APs to buffer packets. When doing handover the MH first tells the previous AP to stop forwarding packets and the new AP to start doing so. In [11,12]the MH decides which AP to use based on the SNR. The AP having the best SNR is ranked as the best one to use. However, this may not be true in the topology shown in figure 2 when the LoadMHs is out of radio range from the MH.

In [13] the bandwidth usage is monitored by APs. This calculated bandwidth utilization is announced in beacons sent by the AP. Our approach decides which AP to use based on network layer characteristics and does not require any modification of existing WLAN infrastructure compared to [13]. [14] suggests a proposal using MIP to decrease the time for handover and to reduce the number of dropped packets. An MH doing handover at the datalink layer tells the old FA to buffer packets for it. After the MH associates with a new FA, the HA tells the old FA to forward buffered packets to the new FA. In the proposal, an FA-sent agent advertisement includes a neighbour list in the message. The neighbour list includes the IP address, link-layer type and channel information. The information is used to enable the MH to select which FA to handover to. To avoid having to wait for three times the advertisement time (as specified in the MIP specification) to discover loss of connection to a FA, a signal from the datalink level is used to inform the network layer. Here all agents need to know the position of all neighbour FAs. This is not required in our proposal.

In [15] support for fast handover is managed at the datalink level. This proposal is based on the usage of a MAC bridge assisting in bridging packets to a roaming MH's new location, while MIP registration is in process. This avoids loosing packets during network layer handover. The delay for handover where packets can be lost only includes the datalink layer handover time. This method only works as long as all MHs do handover to APs connected to the MAC bridge. In a real system this is hardly the case, but for micro mobility it can be used.

More related work is presented in [16,17]. Compared to our proposal a high message complexity is required.

5 Discussion

This paper addresses performance measurements in WLANs. We have proposed and shown how to discover the relative load at MHs in the network layer when connecting wirelessly to APs. Our methodology uses passive measurements based on advertisements like MIP agent advertisements and router advertisements. With increased traffic on a wireless link, collisions will increase and packets will be delayed in buffers. The simulation study reported in this paper demonstrates that RVM is a complement metric that can be used in combination with SNR to improve efficiency and throughput of wireless communications between MHs and APs. This simulation study also supports the theoretical contribution presented in [18].

We have presented a proposed and validated solution to Multihoming in MIP named M-MIP. M-MIP enables an MH to discover multiple networks and to register them at the HA. We have also presented a solution for discovering the RNL in wireless access networks based on 802.11. A simulation study describing the performance of our approach is presented and discussed.

The work presented in this paper has focused on improving performance of MHs using MIP and connecting to 802.11 access networks by enabling MHs to associate with multiple FAs and to evaluate the performance at the network layer. M-MIP gives a higher throughput than if the selection is based only on the SNR. With multiple FAs, one FA will be used for traffic sent through the HA and other FAs can be used for CHs using route optimization. With M-MIP soft handover is enabled, allowing an MH to use multiple FAs. A roaming MH will receive unique packets through both FAs. When the MH decides to handover, it will register with the new FA at the same time as it uses the old FA. With registration completed; packets will be sent using the new FA. With this approach loss of packets because of handover can be avoided. M-MIP does not require any new types of MIP-messages.

Compared to other proposals to enable soft handover with MIP, we present a solution that do not require extended message complexity or modified APs. We use the messages proposed by MIP and analyses the network performance based on this messages.

A prototype based on our proposal is currently being implemented using the Linux platform. We will compare our results from the simulation study presented in this paper with measurements from the prototype.

References

[1] C. Perkins, Mobile IP *IEEE Communications Magazine*, vol. 40, no. 5, pp. 66-82, May, 2002.

[2] R. Chandra, P. Bahl, and P. Bahl, "MultiNet: Connecting to Multiple IEEE 802.11 Networks Using a Single Wireless Card," *Proceedings of IEEE Infocom,* 2004.

[3] C. Ahlund and A. Zaslavsky, "Multihoming with Mobile IP," *6th IEEE International Conference on High Speed Networks and Multimedia Communications,* pp. 235-243, July 2003.

[4] Cisco Systems. RIP, http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/rip.html. 2004.

[5] Cisco Systems. OSPF, http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/ospf.htm. 2004.

[6] UCLA Parallel Computing Laboratory. Glomosim, http://pcl.cs.ucla.edu/projects/glomosim/. 2004.

[7] A. Klemm, C. Lindemann, and M. Lohmann, "Traffic Modeling of IP Networks Using the Batch Markovian Arrival Process, Lecture Notes In Computer Science archive," *Proceedings of the 12th International Conference on Computer Performance Evaluation, Modelling Techniques and Tools,* pp. 92-110.

[8] C. Williamson, Internet Traffic Measurement *IEEE Internet Computing*, vol. Vol 5, pp. 70-74, Nov, 2001.

[9] C. Perkins. IP Mobility Support for IPv4, http://www.ietf.org/internet-drafts/draft-ietf-mip4-rfc3344bis-00.txt. 2004.

[10] D. Johnson , C. Perkins, and J. Arkko. Mobility Support in IPv6, http://www.ietf.org/rfc/rfc3775.txt. 2004.

[11] M. Stemm and R. H. Katz, Vertical Handoffs in Wireless Overlay Networks *Mobile Networks and Applications*, vol. 3, pp. 335-350, 1998.

[12] S. Seshan, H. Balakrishnan, and R. Katz, Handoffs in Cellular Wireless Networks: The Daedalus Implementation and Experience *Wireless Personal Computing*, vol. 4, pp. 141-162, 1997.

[13] H. J. Wang, R. H. Katz, and J. Giese, "Policy-enabled Handoffs Across Heterogeneous Wireless Networks," *Proceedings of the Second IEEE Workshop on Mobile Computer Systems and Applications,* pp. 51-60, Feb. 1999.

[14] J. C.-S. Wu , C.-W. Cheng, N. -F. Huang, and G. -K. Ma, Intelligent Handoff for Mobile Wireless Internet *Mobile Networks and Applications*, vol. 6, pp. 67-79, Jan, 2001.

[15] H. Yokota, A. Idoue, T. Hasegawa, and T. Katao, "Link Layer Assisted Mobile IP Fast Handoff Method over Wireless LAN Networks," *8th International Conference on Mobile Computing and Networking* , pp. 131-139, Sept. 2002.

[16] G. Dommety, K.-E. Malki, M. Khalil, C. Pergins, H. Soliman, G. Tsirtsis, and A.-E. Yegin . Fast Handover for Mobile IPv6. 2004. Internet Draft, IETF.

[17] R. Hseih, Z.-G. Zhou, and A. Seneviratne, "S-MIP: A Seamless Handoff Architecture for Mobile IP," pp. 1774-1784, Apr. 2003.

[18] C. Ahlund and A. Zaslavsky, Extending Global IP Connectivity for Ad Hoc Networks *Telecommunication Systems, Modeling, Analysis, Design and Management*, vol. 24, pp. 221-250, Oct, 2003.

Light-Weight WLAN Extension for Predictive Handover in Mobile IPv6

Soohong Park[1] and Pyung Soo Kim[2*]

[1] Mobile Platform Lab, Digital Media R&D Center,
Samsung Electronics Co., Ltd, Suwon City, 442-742, Korea
soohong.park@samsung.com
[2] School of Electrical Engineering,
Korea Polytechnic University, Korea
peterkim@vsix.net

Abstract. This paper describes a current mechanism for mobile IPv6 fast handover in the wireless LAN and examines its drawbacks, and suggests a new mechanism of predictive fast handover which provides a reduced latency for obtaining address configuration parameter from a router using of light-weight extended WLAN function when performing address configuration through access point. Analytic performance evaluation and comparison have shown that the proposed mechanism is faster in terms of delay than existing mechanism including reduced packet loss when in motion.

1 Introduction

The recent trend towards Internet usage encourages mobility to expand coverage of Internet connectivity and increase resource utilization. Especially, because of increasing mobile node, IPv6 [1], [2] as the next generation of Internet protocol has evolved considerably since it was first defined in the early 1990's. The main thrust of IPv6 is the greatly increased addressing space, which is expected to provide ample address space for the foreseeable future. IPv6 also improved mobility support as Mobile IPv6 [3]-[5] which was already proposed and defined in IETF as a standard. So far, several mechanisms are being studied in IETF to support fast handover [6], [8] when moving to a new network. Wireless and mobile node are subject to changing their point of attachment from one access network to another. The network attachment occurs when a link-layer connection is established and mobile node can send and receive some IP packet in attached network. For network-layer connection, mobile node has to gather required address configuration parameter by receiving router advertisement (RA) message of IPv6 from a router. The fast handover is able to be used for a mobile node to sustain current connection without packet loss and latency.

* Corresponding author: Pyung Soo Kim (peterkim@vsix.net)

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 214–220, 2005.

On the other hand, over the past several years, wireless LAN as IEEE 802.11 standard [7] are being widely deployed around the world. Current wireless access point (AP) performs functions that require IP lever service, and so they are not strictly layer 2 devices, conventional wisdom to the contrary notwithstanding. However, unlike wired network elements, AP requires an additional set of management and control functions related to their primary function of bridging between the wireless and wired medium.

Therefore, light weight access point protocol (LWAPP) [9] is proposed to allow a router or switch to interoperably control and manage a collection of wireless AP in IETF. This paper proposes a new mechanism of predictive fast handover using LWAPP to reduce latency when mobile node moves to a new network. In particular, LWAPP is used by AP to request fast RA message using 802.11 frame. Through this light-weight WLAN extension, AP can trigger fast RA messages including address configuration parameters to the router, and mobile node configure its a new address as care-of address faster than existing schemes.

This paper is organized as follows. In Section 2, related works are simply introduced. In Section 3, the new mechanism is described in detail. In Section 4, analytic performance evaluation and comparison are made. Finally, the conclusions are made in Section 5.

2 Related Works

Related mechanisms surrounding fast handover of mobile IPv6 are simply illustrated in this section.

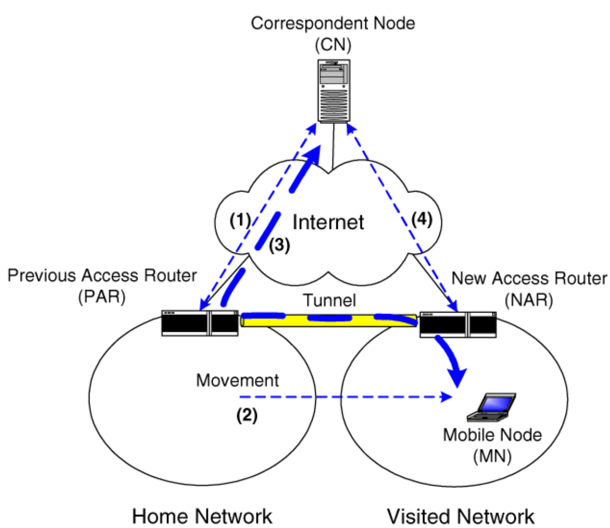


Fig. 1. Reference Model for Fast Handover in Mobile IPv6

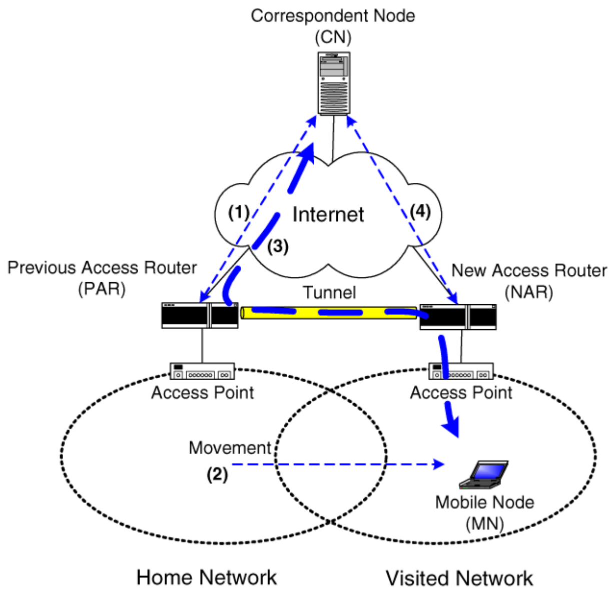


Fig. 2. Reference Model for Fast Handover in 802.11 Wireless LAN

2.1 Mobile IPv6 Fast Handover

It [6] is a dominant mechanism as fast handover in mobile IPv6 today in order to reduce the handover latency. New router solicitation (RS) and router advertisement (RA) messages of IPv6 are defined in this specification for supporting message exchange between previous access router and current access router, so that they set up tunneling path between them and maintain mobile node ongoing connectivity without packet losses. This mechanism aims to allow a mobile node to send packets as soon as it detects a new subnet link, and deliver packets to a mobile node as soon as its attachment is detected by the new access router. This mechanism works without depending on specific link-layer features while allowing link-specific customizations. There are no special requirements for a mobile node to behave differently with respect to its standard mobile IP operations. Reference model of this mechanism is simply shown in Fig. 1.

2.2 Fast Handover for 802.11 Wireless LAN

The main goal of Mobile IPv6 Fast Handover is for shortening the period of service interruption during a change in link-layer point of attachment. On the other hand, this mechanism [8] aims to provide how each may be applied in the 802.11 wireless LAN environment. Both anticipated mode and tunnel mode are proposed in this specification. Reference model of this mechanism is simply shown in Fig. 2.

3 New Mechanism for Predictive Fast Handover Using Light-Weight WLAN Extension

When a mobile node moves to a new network, two kinds of handover happen on the mobile node as link-layer handover and network-layer handover. The network-layer handover occurs when link-layer handover is established in a new link and a mobile node can send and receive some IP packets from neighbor nodes. In particular, a mobile node has to discovery access router (AR) in order to obtain address configuration parameters such as valid prefix, lifetime and etc. RA message in response to RS is used to provide above parameters. In the wireless LAN, RS message requesting RA message has to be reached AR through AP which is operated as a bridge. So, AR must wait for AP operation to be completed because there is no synchronization.

The purpose of the new proposed mechanism is to synchronize among mobile node, AP and AR. The RA Requests message frame newly defined in LWAPP as Fig. 3 is used for synchronization between AP and AR while AP receives a (Re)association.request frame of WLAN from a mobile node. This mechanism does not require any modification of current AP except LWAPP supporting, particularly, this mechanism does not require additional network-layer operations though. This method also can be efficiently used for fast RA without layer 2 trigger protocol [10].

LWAPP must be supported in both AP and AR in order to provide fast RA. (Re)association.request frame is sent by a mobile node when a mobile node wants to change its layer 2 association from its current AP to a new AP and (Re)association.reply frame is sent by new AP either allowing or denying the request. After establishing layer 2 association, a mobile node sends RS message for soliciting RA to the new AR. The new proposed method combines existing methods and make use of LWAPP.

Fig. 3 depicts RA Request message frame format. All LWAPP control messages are sent encapsulated within the LWAPP header. If a router receives RA Request message frame as a trigger from AP, then the router has to send RA message including configuration parameters immediately although link-layer handover is not completed. It can be able to be performed both link-layer handover and network-layer handover at the same time. No random delay is applied to solicit RA message and unsolicited RA message will be sent by router up to 3 times as defined in based IPv6 specification [11]. This value can be reconfigured by operator policy. If a mobile node does not complete its link-layer attachment when receiving unsolicited RA message, it will be silently discarded. If a mobile node does not receive an available RA message sent from a router up to 3 times after completing link-layer attachment, it will generate RS message to solicit RA message as solicited RA.

During AR discovery phase, AP must wait for an interval not less than DiscoveryInterval parameter which is a minimum time, in seconds, that AP must wait after receiving a discovery reply, before sending a join request (Default: 5) for receipt of additional discovery replies according to [9]. In addition, AP

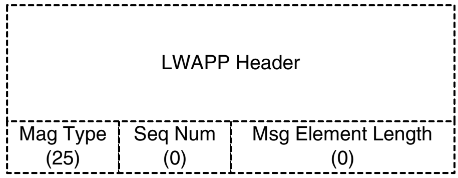


Fig. 3. Option Format of RA Request Message in LWAPP

must store necessary information of the additional discovery replies for candidates. For example, when AR which AP joined does not function as a default router (does not advertise RA message with Prefix information), or its advertised router lifetime = 0, AP can request unsolicited RA to one of the candidate ARs which are stored during AR discovery phase. Preference indicated by new flag or alternatives can be used for AP to decide its default router among them.

When AP receives (Re)association.request frame from a mobile node, if AP decides to allow the request, AP sends (Re)association.reply frame to the mobile node as well as RA Request (Type = 25) to AR using LWAPP simultaneously. After then, AR sends unsolicited RA up to 3 times including address configuration parameters such as valid prefix, lifetime and etc.

4 Implementation and Performance Evaluation

In this section, we present our implementation and distinct performance against the existing mechanism especially reduced delay time and packet loss. We use the Linux operating system (Kernel 2.4.22 version) for all implementation as Access Point and mobile node and encode the RA Request message frame in the Light Weight Access Point Protocol. We measured the date 50 times and the results are shown in Table 1 and Fig. 5. The data was measured at the system S1 using *ethereal*

As we can see in Fig. 5, the delay time which means the roundtrip time between RS and RA messages to configure its new address when attaching a new network is definitely enhanced against the existing mechanism and particular the operation of address configuration is more stable than the existing mechanism. Duplicated Address Detection and Binding Update procedures of IPv6 address configuration are omitted from the evaluation.

We evaluate performance of our proposed mechanism with Fig. 4 network topology when a mobile node is operating RTP/UDP streaming service from a server as corresponding node between different wireless domains. Through this evaluation, we can see the reduced delay, packet loss and stable connection against the existing mechanism. The measurement of mean value and standard deviation are shown in Table 1.

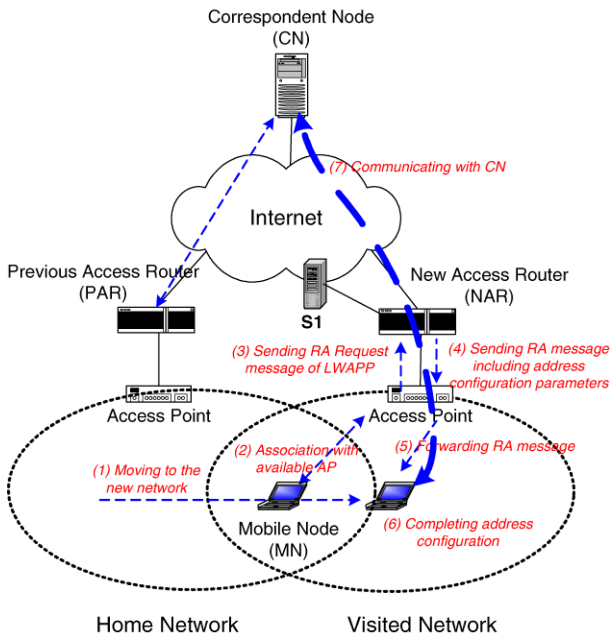


Fig. 4. Network Architecture for performance evaluation

Table 1. Comparison of delay time

Item	Mean Value (msec.)	Standard Deviation
Current mechanism	854.56	91.72020675
New mechanism	134.66	184.3364448

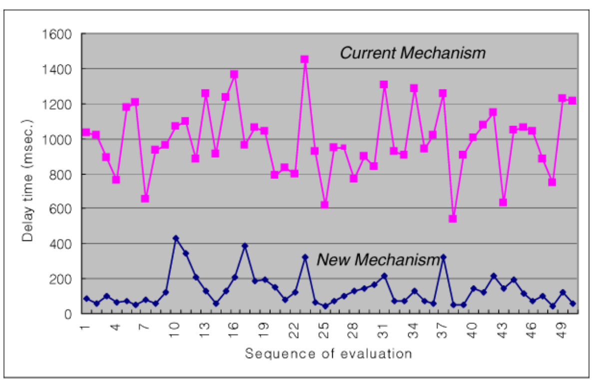


Fig. 5. Comparison and result of performance evaluation

5 Concluding Remarks

This paper describes a current mechanism for mobile IPv6 fast handover in the wireless LAN and examines its drawbacks, and suggests a new mechanism of predictive fast handover in conjunction with Light Weight Access Point Protocol which provides a reduced latency for obtaining address configuration parameter from a router using of light-weight extended WLAN function when performing address configuration through Access Point. The main advantage of the new mechanism is to synchronize among mobile node, AP and AR when a mobile node attaches to a new network. RA Request message frame which is newly defined in LWAPP is used for synchronization between AP and AR when AP receives (Re)association.request frame as layer 2 indication from a mobile node.

Analytic performance evaluation and comparison have shown that the proposed mechanism is faster in terms of delay than existing mechanism including reduced packet loss when in motion.

References

1. S. Deering and R. Hinden: Internet Protocol Version 6 Specification. IETF RFC 2460 (1998)
2. Tatipamula, M., Grossetete, P., Esaki, H.: IPv6 integration and coexistence strategies for next-generation networks. IEEE Communications Magazine, Vol.42. (2004) 88-96
3. D. Johnson, C. Perkins and J. Arkko: IP Mobility Support for IPv6. IETF RFC 3775 (2004)
4. Costa, X. P., Hartenstein, H.: A simulation study on the performance of Mobile IPv6 in a WLAN-based cellular network. Computer Networks, Vol.40. (2002) 191–204
5. Chao, H. C., Chu, Y. M., Lin, M. T.: The implication of the next-generation wireless network design: cellular mobile IPv6. IEEE Transactions on Consumer Electronics, Vol.46. (2000) 656-663
6. R. Koodli: Fast Handovers for MIPv6. IETF Internet Draft (2004)
7. Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. ANSI/IEEE Std 802.11, 1999 Edition.
8. P. McCann: MIPv6 Fast Handovers for 802.11 Networks. IETF Internet Draft (2004)
9. P. Calhoun: Light Weight Access Point Protocol. IETF Internet Draft (2003)
10. A. Yegin: Link Layer Triggers Protocol. IETF Internet Draft (2003)
11. T. Narten: IPv6 Stateless Address Autoconfiguration. IETF RFC 2462 (1999)

Algorithms for Energy-Efficient Broad- and Multi-casting in Wireless Networks

Hiroshi Masuyama[1], Kazuya Murakami[2], and Toshihiko Sasama[1]

[1] Information and Knowledge Engineering,
Tottori University, Tottori, 680-8552, Japan
{masuyama, sasama}@ike.tottori-u.ac.jp
[2] Graduate School, Tottori University, Tottori, 680-8552, Japan
kmurakam@ike.tottori-u.ac.jp

Abstract. The wireless networking environment presents some interesting challenges to the study of broadcasting and multicasting problems, because networks have to be different as occasion demands. Therefore, several types of broadcasting or multicasting protocols have been studied. This paper addresses the problem of broadcasting (and multicasting) focusing on the two points of energy efficient networking and of time efficient computing, where all base stations are fixed and each base station operates as an omni-directional antenna or transceiver. We developed one broadcasting algorithm based on the Stingy method and based on the two performance indices given above. We evaluate this and the other two algorithms based on the Greedy and Dijkstra methods. The purpose of this paper is to make clear the best performing domain of each algorithm. In this paper, the performances of these three algorithms are evaluated in many types of networks. The evaluation gave the result that the Stingy method provides the best performance for energy efficient networking in a type of network where basic stations are distributed wholly, not partially. In this type of network, the Stingy and Dijkstra methods have a trade-off relationship in the two performance indices.

1 Introduction

In this paper, we study the problems of broadcasting and multicasting in all-wireless networks which have fixed base stations. Many researchers have proposed various communication algorithms for various kinds of networks, such as multi-computer (hypercube, mesh, torus or Chordal ring) networks [1], MINs (multi-stage interconnection networks) [2], cellular networks [3], or wireless networks [4]. Most of their reports are concentrated on routing and one-to-all broadcasting, in either the presence or the absence of faulty components of the networks, because of the universality and importance of primitives. In such one-to-all broadcasting schemes, one node of the network, called the "source", has to transmit a message to all other nodes (and also to many other nodes) which are called base stations. In this paper these one-to-all broadcasting schemes will be discussed first. Second, one-to-many multicasting schemes will be commented on.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 221–230, 2005.

The importance of wireless communications is rapidly growing due to their inherent convenient services. The wireless networks studied until now are a little different from each other, but in the network studied here, all base stations are fixed and each base station operates as an omni-directional antenna or transceiver. Therefore, a basic station can broadcast to all the basic stations that lie within its communication range. This means that there exists a trade-off between an immediate broadcast communication from an original source to all other base stations and the other type of broadcast communication, that is, broadcasting is realized by a set of "multiple hopped multicast communications". Since the propagation loss varies nonlinearly with distance (at some where between the second and fourth power), in unicast communications it is best (from the perspective of transmission energy consumption) to transmit at the lowest possible power level, even though doing so requires multiple hops to reach the destination. However, in multicast communications such solutions are not always best, because the use of higher power may permit simultaneous communication to a sufficiently large number of base stations that lie within its communication range, so that the total energy required to reach all members of the broadcast or multicast group may actually be reduced [5].

Since, in our wireless networks, each base station is less prone to failures and it is not necessary to consider any link faults among base stations, the traditional fault tolerance based on the redundant scheme is much less important. Therefore, the only broadcasting or multicasting problems in our networks are to establish a necessary tracking and sufficient stepping base stations, that is, our problems are condensed into the two problems of energy-efficience and calculation-time-efficience.

There have been several papers which treat the problem of broadcasting based on the above two performance indices in the following networks; Veronoi cellular networks [3], wireless networks [4], and mobile Ad Hoc networks [6] and [7]. As it is known that finding a spanning tree of minimum routing cost in a general weighted undirected graph is NP-hard, and our energy-efficient broadcasting algorithm is also NP-hard, we must find an approximate solution. We can consider several basic algorithms as a broadcast algorithm for the wireless networks: these are the Greedy method, the Stingy method, the Minimum-Cost Spanning Tree, and the Dijkstra method. J. E. Wieselthier et al. [4] evaluated three algorithms; the Greedy method, the Minimum-Cost Spanning Tree, and the Dijkstra method in wireless networks. They concluded that their presented algorithm based on the Greedy method provided better performance than the others that have been developed originally in wired environments. However, in broadcast or multicast applications it is not prudent to draw such conclusions because the networks may not always have base stations randomly located within a region. Each algorithm may have an advantage in each particular circumstance. In this paper, we would like to make clear the solution to this problem. Since it is shown in [4] that Minimum-Cost Spanning Tree provided the worst performance in their comparison, we would like to remove this algorithm in our comparison. We will first present an algorithm based on the Stingy method, as the one remaining algorithm, and evaluate this and the other methods as the optimal broadcast (and multicast) algorithm fitted for a wireless environment. We will evaluate them based on performance in many different networks.

2 Wireless Networks and Wireless Communication Model

We will consider aspects of wireless networks, such that they consist of N nodes, which are distributed over a specified region. Each node operates as an omni-directional antenna or transceiver, so it can transmit the message to all nodes within the communication range or receive the message from a transmitting node if the node is within the communication range of the transmitting node, or it can support several multicast sessions simultaneously where each multicast group consists of the source node plus at least one destination node. The connectivity of a wireless network depends on the total transmission power which is produced by all transmitting nodes.

We will assume that the signal power received varies as $r^{-\alpha}$, where r is the range and α is a parameter that typically takes on a value between 2 and 4, depending on the characteristics of the communication medium (in this paper, one case of 2 will be discussed). This means that the transmitted power required to support a link between two nodes separated by range r is proportional to r^{α}.

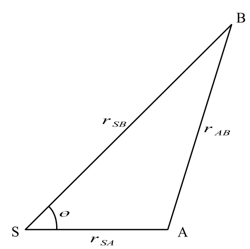


Fig. 1. Broadcasting to two destinations

We will consider a case in which a source (or a transmitting) node S must broadcast to two destination nodes A and B as shown in Fig.1. Paper [4] has shown the following energy efficient conditions for broadcasting from S: S should broadcast with power r_{SB}^2 when the following equation (1) holds, and with power r_{SA}^2 otherwise.

$$x^{\alpha} - 1 < (1 + x^2 - 2xcos\theta)^{\frac{\alpha}{2}}, \tag{1}$$

where $x = r_{SB}/r_{SA}$.

The latter case means that, in order to perform the required broadcast, node A must also broadcast with power r_{AB}^2. Our performance indices are the total power of the broadcast tree and the calculation time required to obtain the broadcast tree. In the latter case, since the broadcasting tree is a set of arcs (SA

and AB) and nodes (S, A, and B), then the total power of the broadcast tree is $r_{SA}^2 + r_{AB}^2$.

3 The Broadcasting Algorithm

In the Stingy method the first task is an immediate broadcast communication from an original source to the furthest base station, and the next is to find the furthest intermediate station based on energy-efficience and "one hopped multicast communications". In the following, we will present one Stingy algorithm we have built up. Let the power required to communicate from node A to node B, and the path consisting of hopping nodes to communicate, be E_{AB} and P_{AB}(or $P_{A12...NB}$ when $12...N$ are known as hopping chain nodes), respectively.

[Broadcast Algorithm]

S1: Let the total power E be E_{SD} where D is the furthest node from the original source node S.
$E_{det} = 0$. $\mathtt{R}$ = A set of nodes except S.
S2: Check whether there is at least one node in the circle with the diameter SD or not.
If there is at least one node, then go to S3, or else go to S6.
S3: By using the Dijkstra algorithm, check in the circle whether there is at least one set of multiple hopped multicast communications brought on an energy-efficient or not, that is check whether E is greater than $E' = E_{S1} + E_{12} + \cdots + E_{ND}$ where $12...N$ means the path $P_{S12...ND}$ consisting of hopping chain nodes to communicate from S to D.
If there is at least one energy-efficient path, go to S4, otherwise go to S6.
S4: Check in $\mathtt{R}$ whether there is at least one node which cannot communicate from S with the energy E' or not.
If there is at least one node, then let the furthest node from S among them be D' and go to S5 (in order to set a new E'), otherwise go to S6.
S5: Find the furthest node I from S on path $P_{S12...ND}$ located in the circle with radius SD'.
Let the energy E' be $E_{SD'} + E_{ID} + E_{det}$.
If E is greater than E', then $E \leftarrow E'$, $E_{det} \leftarrow E_{det} + E_{ID}$, and remove all succeeding nodes to I on path $P_{S12...ND}$ from $\mathtt{R}$.
$D \leftarrow D'$, and go to S2.
S6: End.

Let us see the above algorithm in an example shown in Fig.2.1.

S1: $E = E_{S9}$.
$E_{det} = 0$.
S2: See Fig.2.2.
S3: See Fig.2.3.
$E' = E_{S2} + E_{29}$.

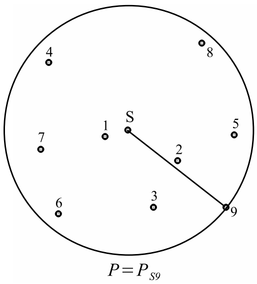


Fig. 2.1. Step 1

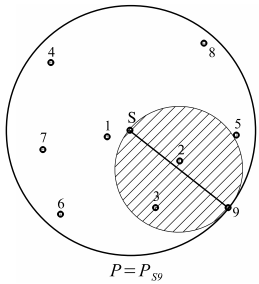


Fig. 2.2 Step 2

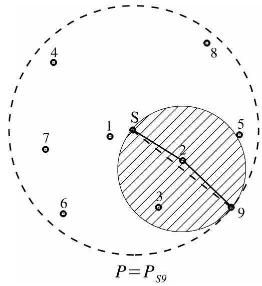


Fig. 2.3 Step 3

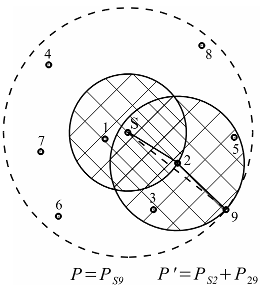


Fig. 2.4 Step 4

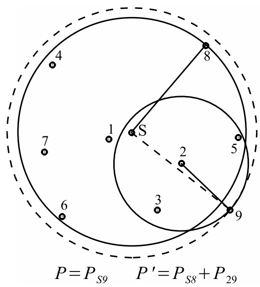


Fig. 2.5 Step 5

Fig. 2. Algorithm for broadcasting

S4: See Fig.2.4.
Nodes 4, 6, 7, 8 are nodes which cannot communicate from S with $E' = E_{S2} + E_{29}$, and $D' = 8$.

S5: See Fig.2.5.
Since $E = E_{S9}$ is not greater than $E' = E_{S8} + E_{29}$, hold E.

S6: The final E is E_{S9}.

4 Performance Results

We have evaluated the performance of the three algorithms for 9 different patterns of networks where each pattern has many network examples. Networks with a specified number of nodes (typically 10 or 100) are generated within a square region where the 5×5 region is characterized in the following nine specified patterns, as shown in Fig.3:

Pattern1; nodes are distributed in normal distribution only in square region 1.
Pattern2; nodes are distributed in normal distribution only in square regions 1 and 2.
Pattern3; nodes are distributed in normal distribution only in square regions 1 and 3.
Pattern4; nodes are distributed in normal distribution only in square regions 1,2, and 4.

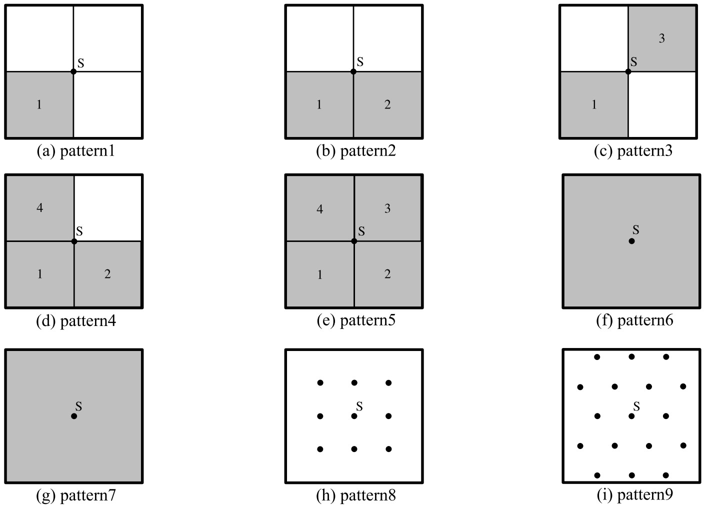


Fig. 3. Nine basic patterns to generate many network examples

Table 1. Mean of normalized broadcasting power

(a)10 node networks

	Stingy method	Greedy method	Dijkstra method
pattern1	1.135466	**1.010530**	1.050697
pattern2	1.099219	**1.015794**	1.028591
pattern3	1.100997	**1.011479**	1.034172
pattern4	1.031288	**1.020940**	1.026823
pattern5	**1.012002**	1.044857	1.071954
pattern6	**1.034987**	1.044857	1.071954
pattern7	**1.029927**	1.143659	1.215390
pattern8	**1.000000**	2.000000	2.000000
pattern9	**1.000000**	1.250879	1.250879

(b)100 node networks

	Stingy method	Greedy method	Dijkstra method
pattern1	2.251798	**1.000466**	1.161417
pattern2	1.614182	**1.000000**	1.145018
pattern3	1.608082	**1.000000**	1.132338
pattern4	1.336774	**1.000437**	1.144432
pattern5	1.153664	**1.005818**	1.159290
pattern6	**1.004041**	1.182403	1.425783
pattern7	**1.000621**	1.201188	1.560292
pattern8	**1.000000**	1.960000	2.280000
pattern9	**1.000000**	1.524216	2.038621

Pattern5; nodes are distributed in normal distribution in all square regions 1, 2, 3, and 4.
Pattern6; nodes are distributed in normal distribution in the whole region.
Pattern7; nodes are randomly distributed throughout the whole region.
Pattern8; nodes are distributed in a lattice-patterned distribution in the whole region. In the case of Fig.3 the total number of nodes is 9.
Pattern9; nodes are distributed in triangular-pattern distribution in the whole region. In the case of Fig.3 the total number of nodes is 17.

A central node is chosen to be the Source. In all patterns, the results are based on the performance of 100 randomly generated networks which typically have 10 or 100 nodes (9 or 121 nodes in Pattern 8, and 13 or 93 nodes in Pattern 9). As mentioned above, one of our performance indices is the total power of the broadcast tree. To facilitate the comparison of our algorithms over a wide range of network examples, we used the notion of the normalized power for each network example, as the same as mentioned in [4]. Let $Q_{best}(m)$ be the lowest power, in all algorithms in our comparison, required to broadcast in the network m. Based on the total power $Q_i(m)$ of the broadcast tree associated with algorithm i for network m, we then define the normalized power to be

$$N_i(m) = Q_i(m)/Q_{best}(m).$$

Table 2. Mean time to calculate the broadcast tree

(a)10 node networks.($10^{-9}sec$)

	Stingy method	Greedy method	Dijkstra method
pattern1	0.035	0.034	**0.012**
pattern2	0.032	0.032	**0.012**
pattern3	0.031	0.032	**0.012**
pattern4	0.027	0.031	**0.011**
pattern5	0.022	0.032	**0.011**
pattern6	0.022	0.035	**0.011**
pattern7	0.021	0.032	**0.012**
pattern8	0.022	0.030	**0.011**
pattern9	0.021	0.058	**0.011**

(b)100 node networks.($10^{-6}sec$)

	Stingy method	Greedy method	Dijkstra method
pattern1	0.006742	0.011859	**0.000359**
pattern2	0.003879	0.013150	**0.000346**
pattern3	0.003759	0.013070	**0.000343**
pattern4	0.002089	0.011991	**0.000359**
pattern5	0.001403	0.013535	**0.000327**
pattern6	0.001133	0.012247	**0.000334**
pattern7	0.001302	0.012453	**0.000362**
pattern8	0.000513	0.020885	**0.000498**
pattern9	0.000286	0.009441	**0.000262**

This index provides a measure of how close each algorithm comes to providing the lowest-power tree.

4.1 Total Power of the Broadcast Tree

Tables 1 (a) and (b) summarize performance results associated with total power of the broadcast tree required for each algorithm in networks with 10 and 100 nodes, respectively. The Stingy and Greedy methods share the best performing regions, that is, the Stingy method provides the best average performance in patterns over 5 or 6, while the Greedy method provides the best performance except in the specialized region of the Stingy method.

4.2 Calculation Time Required to Obtain the Broadcast Tree

Tables 2 (a) and (b) summarize the performance results associated with the other performance index. For the calculation time required to obtain the broadcast tree, the Dijkstra method provides the best average performance. The Stingy method gave the second best performance.

5 Algorithms for Multicasting

In multicasting, we assume that we may use some non-multicast nodes (although they are not necessary to transmit massages) as intermediate nodes to transmit a message to multicast nodes. As was explained in [4], to obtain the multicast tree based on the Greedy method or Dijkstra method, the broadcast tree is pruned by eliminating all transmissions that are not needed to reach the members of the multicast group. More specifically, nodes with no downstream destinations will not transmit, and some nodes will be able to reduce their output [4].

[Multicast Algorithm]

S1: Let the total power E be E_{SD} where D is *the furthest multicast node* from the original source node S.
$E_{det} = 0$. $\mathtt{R}$ = A set of *multicast nodes* except S.

S2: Check whether there is at least one node in the circle with the diameter SD or not.
If there is at least one node, then go to S3, otherwise go to S6.

S3: By using the Dijkstra algorithm, check in the circle whether there is at least one set of multiple hopped multicast communications brought on an energy-efficient or not, that is check whether E is greater than $E' = E_{S1} + E_{12} + \cdots + E_{ND}$ where $12 \ldots N$ means the path $P_{S12 \ldots ND}$ consisting of hopping chain nodes to communicate from S to D.
If there is at least one energy-efficient path, then go to S4, otherwise go to S6.

S4: *In* $\mathtt{R}$, check whether there is at least one node which cannot communicate from S with the energy E' obtained now or not.

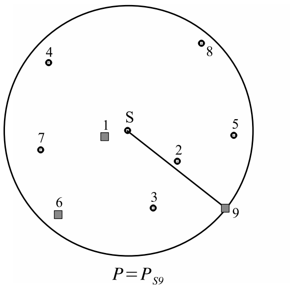


Fig. 4.1 Step 1

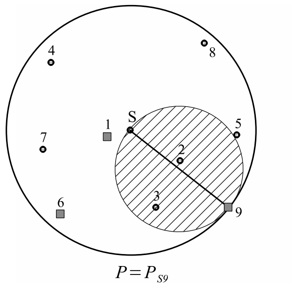


Fig. 4.2 Step 2

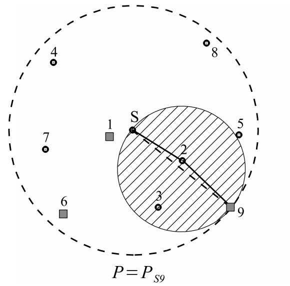


Fig. 4.3 Step 3

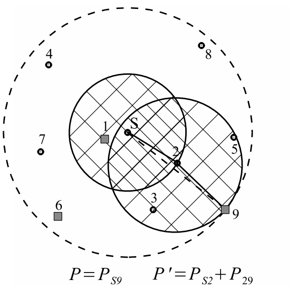


Fig. 4.4 Step 4

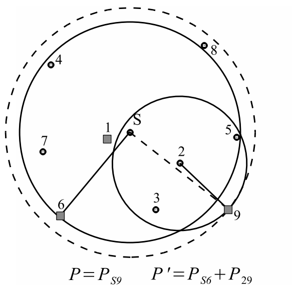


Fig. 4.5 Step 5

Fig. 4. Algorithm for multicasting

If there is at least *one multicast node*, then let *the furthest multicast node* from S among them be D' and go to S5 (in order to set new E'). Otherwise go to S6.

S5: Find the furthest node I from S on path $P_{S12...ND}$ located in the circle with radius SD'.
Let the energy E' be $E_{SD'} + E_{ID} + E_{det}$.
If E is greater than E', then $E \leftarrow E'$, $E_{det} \leftarrow E_{det} + E_{ID}$, and remove *all succeeding multicast nodes* to I on path $P_{S12...ND}$ from R.
$D \leftarrow D'$, and go to S2.

S6: End.

Let us see the above algorithm in an example shown in Fig.4.1 where multicast nodes are nodes 1, 6, and 9.

S1: $E = E_{S9}$.
$E_{det} = 0$.

S2: See Fig.4.2.

S3: See Fig.4.3.
$E' = E_{S2} + E_{29}$.

S4: See Fig.4.4.
Node 6 is a node which cannot communicate from S with $E' = E_{S2} + E_{29}$, and $D' = 6$.

S5: See Fig.4.5.
Since $E = E_{S9}$ is not greater than $E' = E_{S6} + E_{29}$, hold E.
S5: The final E is E_{S9}.

6 Conclusion

In this paper, in a type of wireless network where all base stations are fixed and each base station operates as an omni-directional antenna or transceiver, we have addressed some of the issues associated with two performance indices; energy-efficience and calculation-time-efficient broadcasting (and multicasting). We have presented one preliminary algorithm based on the Stingy method to address this problem, and made clear the best performance domain for the representative and three traditional algorithms. The evaluation gave the result that the Stingy method provides the best performance for energy efficient networking in the domain where basic stations are distributed in the whole network, in detail, irrespective of distribution patterns as long as basic stations are distributed in the whole network. The Dijkstra method is superior to the others in calculation time. The Stingy and Dijkstra methods have a trade-off relationship within our performance induces.

For the future work on our developed Stingy-method-based algorithm, further research is needed to develop an algorithm variable in scale that can achieve nearly optimal performance.

References

1. S.Park and B.Bose, "All-to-All Broadcasting in Faulty Hypercubes," IEEE Trans. Computers, Vol.46, No.7, pp.749-755, July 1997.
2. M.Yaku and H.Masuyama, "A Method of Fault-Tolerant All-to-All Personalized Communication in Banyan Networks," IPSJ Journal, Vol.42, No.10, pp.2476-2484, Oct.2001.
3. B.A.Elisabeth and S. David, "An Optimal Fault-Tolerant Broadcasting Algorithm in Voronoi Cellular Networks," Proceedings of 15th IASTED International Conference Parallel and Distributed Computing and System (PDCS'03), pp.69-74, 2003.
4. J.E.Wieselthier, G.D.Nguyen, and A.Ephremides, "On the Construction of Energy Efficient Broadcast and Multicast Trees in Wireless Networks," Proceedings of IEEE INFOCOM '00, pp.585-594, 2000.
5. J.E.Wieselthier and G.D.Nguyen, "Algorithms for Energy-Efficient Multicasting in Static Ad Hoc Wireless Networks," Journal of Mobile Networks and Applications, No.6, pp.251-263, 2001.
6. C.M.Chao, J.P.Sheu and C.T. Hu, "Energy-Conserving Grid Routing Protocol in Mobile Ad Hoc Networks," Proceedings of the 2003 International Conference on Parallel Processing (ICPP'03), pp.265-272, 2003.
7. I.Chatzigiannakis, S.Nikoletseas and P.Sirakis, "An Efficient Routing Protocol for Hierarchical Ad-hoc Mobile Networks," Proceedings of 15th International Parallel and Distributed Processing Symposium (IPDPS'01), p.185, April 2001.

Converting SIRCIM Indoor Channel Model into SNR-Based Channel Model

Xiaolei Shi[1], Mario Hernan Castaneda Garcia[2], and Guido Stromberg[1]

[1] Infineon Technologies AG, 81730 Munich, Germany
{xiaolei.shi, guido.stromberg}@infineon.com
[2] Technische Universität München, 80333 Munich, Germany

Abstract. The event-driven network simulation platforms, e.g. NS2 and GloMoSim, adopt some general wireless channel propagation models providing the pathloss of the channel as well as the SNR of each received packet to estimate the BER. However, these SNR-based general channel models give more accurate results for outdoor environments than indoor. On the other hand, the SIRCIM is a well-established channel impulse response based indoor channel model taking different indoor situations into consideration. This paper presents a method that integrates the SIRCIM model into the simulation platforms using SNR-based model by converting the channel impulse response of the SIRCIM into the pathloss. This conversion provides not only a more precise SNR-based indoor channel model but also the inter-symbol-interference information to achieve a more accurate BER estimation for simulating the indoor networks such as wireless sensor networks and wireless personal area networks.

1 Introduction

Wireless Sensor Networks (WSNs) and Wireless Personal Area Networks (WPANs), as the enabling technologies of ubiquitous computing, are booming networking research topics attracting more and more attention. Upon developing a WSN or WPAN, network simulation is a necessary task, in which a proper channel propagation model directly affects the simulation results. Since WSNs and WPANs are mostly deployed in indoor environments, finding a precise indoor channel model becomes an important simulation issue. However, the most popular network simulation platforms, e.g. NS2 [1] and GloMoSim [2], provide only general channel propagation models, such as free space, shadowing pathloss models and Rice and Rayleigh fading models, which generate a total channel power pathloss to get the signal-to-noise ratio (SNR) for each packet. Then the bit-error-rate (BER) is estimated according to the SNR by a certain algorithm. These general models are typically used for simulating the outdoor environments.

The indoor channel is much more difficult to model than the outdoor channel, because it is susceptible to the changes in the geometry of environment, e.g. a door being shut and a walking person around one of the antennas. Among many available indoor channel models, the SIRCIM (Simulation of Indoor Radio Channel IMpulse response) model is a well-established model based on the

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 231–238, 2005.

statistics of large amount of measurements from many different kinds of buildings. Therefore, it is a good choice to implement into simulation platforms using an SNR-based channel model to perform more precise simulation.

However, there is still a big gap to be bridged. The SIRCIM generates a channel impulse response that should be convolved with a transmitted signal to get a received signal, while the SNR-based model needs the total power pathloss of the signal for each packet. Therefore, how to convert the channel impulse response into a total power pathloss becomes the key issue. This paper will give an answer to it.

2 SIRCIM Model

2.1 General Channel Model Concepts

A multipath fading wireless channel can be modelled as a linear time-varying filter [3] and represented by an impulse response

$$h(t) = \sum_K A_K e^{j\theta_K} \delta(t - T_K), \tag{1}$$

where A_K represents the amplitude, $e^{j\theta_K}$ denotes the phase shift caused by reflection, diffraction and scattering, and T_K is the time delay of the Kth path in the channel with respect to the arrival time of the first arriving component, called excess delay. The received signal can be calculated by applying the convolution of the transmitted signal with this impulse response. The fluctuations of the amplitudes, phases and multipath delays of a signal can be referred to as fading.

Indoor and outdoor channel models share some basic characteristics as described above, but indoor channel models cannot be viewed as a scaled down version of the outdoor channel model, because it has its own special features: severe pathloss, non-stationary, low doppler spread and small access delay.

So far, many researches have been done on modelling indoor wireless channel In which the SIRCIM model is a well-established model considering different types of indoor environments.

2.2 SIRCIM Principles

There exist two types of general topographies found in the indoor environment: line of sight (LOS) and obstructed one (OBS) [4] [5]. For each of the two we can further classify them into three smaller groups: *open plan* (OPEN), *hard partitioned*(HARD) and *soft partitioned* (SOFT) as indicated in [6]. OPEN buildings are those that have large open spaces where exists only few large obstructions or scatterers, e.g. factories. HARD buildings are typical multiple floor buildings partitioned by concrete or drywall, e.g. offices. SOFT buildings are also multiple floor buildings with large open spaces but partitioned into small offices using dividers that do not extend from the floor to the ceiling.

In [3], Seidel and Rappaport present a model that determines a channel impulse response for different types of environment, whether it is LOS or OBS in an OPEN, HARD or SOFT environment. To this end, they describe the following: the distribution of the number of multipath components; the probability of receiving a multipath component at a particular excess delay, the distributions of the mean amplitude and phase of each multipath component, and the distribution of the large and small scale fading of each multipath component. The spatial and temporal correlation among multipath components are also modelled. These parameters gather the information that is necessary to statistically describe a wireless channel.

Next, the procedure how the SIRCIM model generates a channel impulse response will be introduced. The open plan with LOS case is used as an example. For other cases, the steps are the same except for some differences in the equations and parameters, see [6] for details.

1. *Distribution of the Number of Multipath Components.*
   The number of multipath components N_p is taken to be Gaussian distributed with a mean of $\overline{N_p}$ and a standard deviation σ_p. The mean of this distribution, $\overline{N_p}$, is also a random variable being uniformly distributed between 9 to 35. And σ_p is modelled by

$$\sigma_p = 0.492 \times (\overline{N_p} - 4.77). \tag{2}$$

2. *Probability of the Arrivals of Multipath Components.*
   The probability that a multipath component will arrive at a receiver at a particular excess delay is modelled by piecewise functions of the excess delay:

$$P_r(T_K) = \begin{cases} 1 - \frac{T_K}{367}, & T_K < 110ns \\ 0.65 - \frac{(T_K - 110)}{360}, & 110ns < T_K < 200ns \\ 0.22 - \frac{(T_K - 200)}{1360}, & 200ns < T_K < 500ns, \end{cases} \tag{3}$$

   where T_K is the delay and takes values that are integer multiples of 7.8 ns.
3. *Distribution of the Phases of the Multipath Components.*
   The phases for each multipath component θ_K is uniformly distribution within $[0, 2\pi)$ [7].
4. *Large Scale Fading of Multipath Components.*
   The large scale amplitude of each component K is log-normally distributed around a mean amplitude $\overline{A_K}$ with a standard deviation $\sigma_{large-scale}$ of 4 dB. The mean amplitude $\overline{A_K}$ in dB obeys the exponential law

$$\overline{A_K}(T_K) = 10 \times n(T_K) \times \log\left(\frac{d}{10\lambda}\right), \tag{4}$$

   where d is the distance between the two antennas, and λ denotes the wavelength of the signal. The distribution of n is given by

$$n(T_K) = \begin{cases} 2.5 + \frac{T_K}{39}, & T_K < 15ns \\ 3.0 + \frac{(T_K - 15.6)}{380}, & 15ns < T_K < 250ns \\ 3.6, & 250ns < T_K < 500ns. \end{cases} \tag{5}$$

5. *Small Scale Fading of Multipath Components.*
   The cumulative distribution function for $\sigma_{small-scale}$ is given by

$$F(\sigma_{small-scale}) = 1 - exp\left(\frac{-(\sigma_{small-scale} - a)^2}{2}\right), \tag{6}$$

   where the offset parameter a is 0.25 dB.

Thus, taking all previous results the amplitude of an individual multipath component can be modelled by the distribution

$$A_K(T_K) = N\left[N\left[\overline{A_K}, \sigma^2_{large-scale}\right], \sigma^2_{small-scale}\right], \tag{7}$$

where $N[x, \sigma_x^2]$ denotes the log-normal distribution with mean x (dB) and standard deviation σ_x (dB). As a result, with the equations (2) to (7) a statistical discrete channel impulse response can be generated for a particular channel in an open plan building with LOS.

3 Convert SIRCIM into SNR-Based Model

3.1 Traditional SNR-Based Channel Models

In the propagation models of simulation platforms, e.g. NS2 and GloMoSim, a total propagation pathloss is calculated according to certain channel models for each transmitted packet. The channel models used are as following.

Path Loss Model: The large scale effect is inversely proportional to the antenna separation distance, where this distance is raised to an exponent 2, as given by the Friis Free Space model:

$$P_r(d) = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d^2 L}, \tag{8}$$

The pathloss $L_p(d)$ in dB can thus be represented by

$$L_p(d) = L_p(d_0) + 10\ \log\left(\frac{d_0}{d}\right)^2, \tag{9}$$

where d_0 is some reference distance and $L_p(d_0)$ is the pathloss at this distance.

To add fading effects into the free space model, a so called log normal shadowing model is given by

$$L_p(d) = L_p(d_0) + 10\ \log\left(\frac{d_0}{d}\right)^n + X_{dB}, \tag{10}$$

where the exponent becomes n instead of 2, X_{dB} is a Gaussian random variable with zero mean and standard deviation σ [dB]. This log normal shadowing model is used in NS2 to simulate the fading effects and some values for n and σ [dB] are listed for outdoor and indoor environments with LOS or OBS cases [1].

Fading Model: Instead of using the log normal shadowing model to simulate fading, a pathloss taken from a random variable that is Rayleigh distributed for the OBS case or Rice distributed for the LOS case can be added to the pathloss generated by the free space model to simulate the fading effect, which is adopted by the GloMoSim. In the Rice distribution, a so-called K factor is defined, which has a range of 6 to 12 dB for indoor environment [8].

3.2 SIRCIM Conversion

To use SIRCIM model in an SNR-based simulation platform, a total power pathloss must be derived from the information given by the SIRCIM's channel impulse response. First, the total amplitude gain at the receiver can be obtained by summing up all the multipath components generated by the SIRCIM as phasors, which is given by

$$A = \left| \sum_{K=1}^{N_p} A_K e^{j\phi_K} \right|, \tag{11}$$

where N_p is the number of multipath component in the channel impulse response, A_K denotes the linear amplitudes of the discrete impulse response, ϕ_K represents the phase shift. Note that ϕ_K is the sum of two phase shifts – the phase shift caused by excess delay $\theta'_K(T_K)$, and the phase shift caused by reflection, diffraction and scattering θ_K. Since N_p, A_K, T_K and θ_K are known parameters generated by the SIRCIM, the only parameter left to be determined is $\theta'_K(T_K)$, which is given by

$$\theta'_K(T_K) = \frac{(T_K \times c_0)\, mod\, \lambda}{\lambda} \times 2\pi, \tag{12}$$

where c_0 is the velocity of light and λ is the wavelength of the carrier frequency. Another simple but reasonable way to get ϕ_K is taking this overall phase shift as a uniformly distributed random number within $[0, 2\pi)$.

The phasorial sum can be made by breaking each amplitude $A_K e^{j\phi_K}$ into an in phase component A_{K-I} and a quadrature component A_{K-Q} given by

$$A_{K-I} = A_K \cos\phi_K \tag{13}$$

$$A_{K-Q} = A_K \sin\phi_K. \tag{14}$$

Then summing up all the in phase components and the quadrature components separately produces a net in phase component A_I and a net quadrature component A_Q. Hence, the resulting amplitude gain A is calculated as

$$A = \sqrt{A_I^2 + A_Q^2} \tag{15}$$

$$= \sqrt{\left(\sum_{K=1}^{N_p} A_{K-I}\right)^2 + \left(\sum_{K=1}^{N_p} A_{K-Q}\right)^2} \tag{16}$$

$$= \sqrt{\left(\sum_{K=1}^{N_p} A_K \cos\phi_K\right)^2 + \left(\sum_{K=1}^{N_p} A_K \sin\phi_K\right)^2}. \tag{17}$$

This A in dB (A_{dB}) is the final total channel *amplitude gain*, thus the total channel *power pathloss* needed by the SNR-based model is $-2A_{dB}$, including both the propagation and fading effects.

3.3 Advantage of Converting SIRCIM to SNR-Based Model

Although the traditional SNR-based channel models can somehow simulate the indoor channel by properly defining their parameters, e.g. the n and X_{dB} in the log normal shadowing model and the K factor in the Rice fading, they still cannot achieve precise results because they are quite simple and cannot present the dynamical features of the complex indoor environments.

However, the SIRCIM [3] model considers different information regarding the type of environments, such as whether there is a LOS path or not and the type of building: OPEN, HARD or SOFT. This statistical model is more robust and considers different factors based on large amount of measurements. Therefore, converting the SIRCIM model into SNR-based model generates much more realistic channel pathloss supporting a more accurate simulation (see simulation results in section 4).

Furthermore, converting SIRCIM to SNR-based Model also provides the information of the inter-symbol-interference (ISI), which is a very important channel characteristic missed in the normal SNR-based models. However, our converting gives the possibility to do more accurate BER estimation based on both the pathloss and the ISI.

4 Simulation Results

To evaluate our proposal, we program our algorithm and other channel models using C language. Only the simulation results from the OBS case are presented as examples because of the limited space of the paper.

SIRICM: Fig. 1 shows the pathloss probability density functions (PDFs) of the SIRCIM model in the OBS case. In the OPEN environment, the PDFs of different distances (10, 20, 30m) have almost the same shape with relative small deviation except for the increasing mean due to the distance. In the HARD/SOFT case, both the mean and the deviation rise with the increasing of the distance, showing that the HARD/SOFT is much more dynamic than the OPEN.

Mean Comparison: In Fig. 2, the mean pathlosses of the sum of the Friis free space pathloss and Rayleigh fading, shadowing model, and SIRCIM model are shown. The mean pathlosses of the SIRCIM model increase faster than the others, which just indicates that the indoor channel suffers a severe pathloss when the distance increases.

PDF Comparison: Next, the PDFs of different models will be compared. Fig. 3 shows the OBS case with a distance of 10m. Rayleigh fading provides a reasonable curve, but it cannot distinguish the OPEN, HARD and SOFT cases. In the shadowing model, $n = 5$ is for OBS and $\sigma = 6.8, 7.0, 9.6$ are for the OPEN, HARD and SOFT respectively [1]. Because of the same exponent n, shadowing model has the same mean for the OPEN, HARD and SOFT, which is unrealistic. The other problem of the shadowing model is that, although the deviations of the HARD and SOFT are larger than that of the OPEN, an unrealistic very

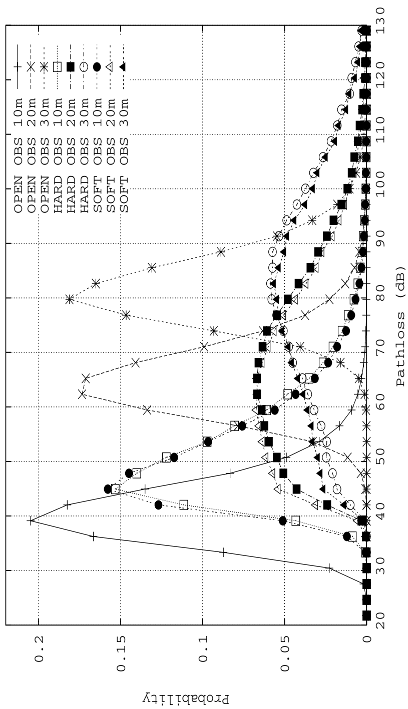


Fig. 1. PDFs of SIRCIM (OBS)

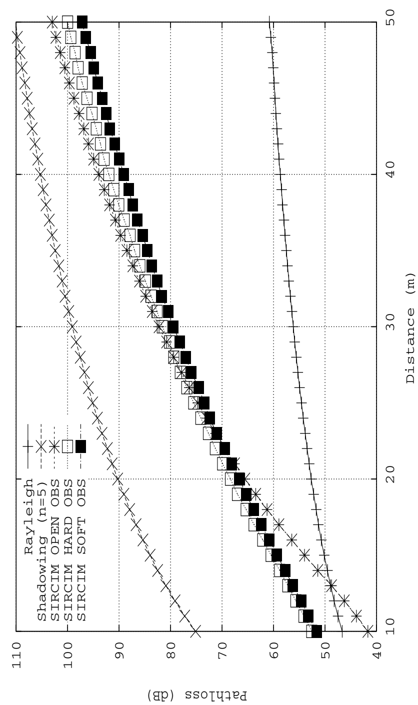


Fig. 2. Pathloss vs. Distance (OBS)

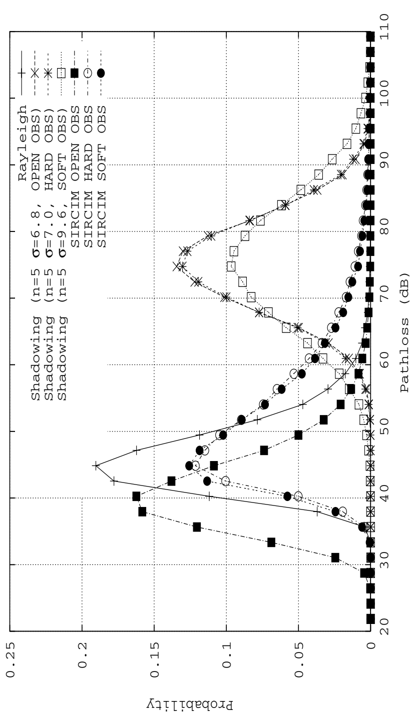


Fig. 3. PDFs Comparison (10m, OBS)

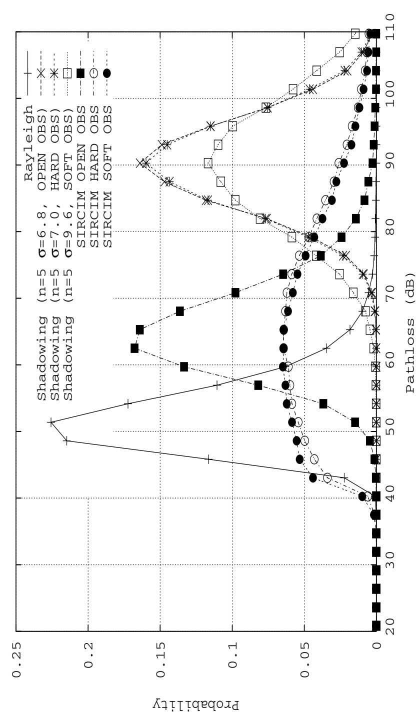


Fig. 4. PDFs Comparison (20m, OBS)

small pathloss could be generated with the same probability as a very big one, which is also not true for an indoor channel. However, the SIRCIM model gives a much more reasonable distribution that the OPEN has relative small mean and deviation than the HARD and SOFT. Furthermore, the probability of generating an unrealistic small pathloss is also much smaller than in the shadowing model.

Fig. 4 shows the same comparison with a distance of 20m, from which we can see that the very big deviations of the HARD and SOFT from the SIRCIM model indicate the dynamic characteristics of the indoor channel when the distance is large.

5 Conclusion

This paper addresses the indoor wireless channel propagation model for network simulation purpose. Because the general SNR-based channel models in the most widely used simulation platforms so far are not adequate for accurate indoor channel simulation, we propose to convert the well-established indoor channel model, SIRCIM, into an SNR-based model by deriving the total power pathloss from the channel impulse response generated by the SIRCIM. Simulation results show that our proposal generates more realistic results for simulating the total indoor channel power pathloss. Besides, our approach also provide inter-symbol-interference information for possibly more accurate bit error rate estimation, which was not available before. The drawback of our proposal is a longer simulation time because of the increased complexity of the model.

References

1. The VINT Project http://www.isi.edu/nsnam/ns/ns-documentation.html: The ns Manual. (2003)
2. UCLA PCL http://pcl.cs.ucla.edu/projects/glomosim/GloMoSimManual.html: GloMoSim Manual. Version 1.2 edn. (2003)
3. Rappaport, T.S., Seidel, S.Y.: Statistical channel impulse response model for factory and open plan building radio communication system design. IEEE Transactions on Communications (1991)
4. Rappaport, T.S.: Wireless Communications,Principles and Practice. Prentice Hall, Braunschweig/Wiesbaden (2002)
5. Mineo Takai, Rajive Bagrodia, A.L., Gerla, M.: Impact of channel models on simulation of large scale wireless networks. In: SigMobile, UCLA Computer Science Department (1999)
6. Nuckols, J.E.: Implementation of Geometrically Based Single-Bounce Models for Simulation of Angle-of-Arrival of Multipath Delay Components in the Wireless Channel Simulation Tools, SMRCIM and SIRCIM. PhD thesis, Virginia Polytechnic Institute and State University, Blacksburg, Virginia (1999)
7. Hashemi, H.: The indoor radio propagation channel. In: Proceedings of the IEEE. Volume 81. (1993)
8. Roberts, J.A., Bargallo, J.M.: Dpsk performance for indoor wireless rician fading channels. IEEE Transactions on Communications **42** (1994)

CAWAnalyser: Enhancing Wireless Intrusion Response with Runtime Context-Awareness

Choon Hean Gan, Arkady Zaslavsky, and Stephen Giles

School of Computer Science and Software Engineering, Monash University,
900 Dandenong Rd, Caulfield East, Victoria, Australia
{chgan, arkady.zaslavsky, stephen.giles}@csse.monash.edu.au

Abstract. Most existing wireless IDSs do not provide timely active responses to wireless intrusions as the execution of the responses is done manually by the administrator. Some wireless IDSs address this issue by providing automated responses. On one hand, they reduce the chances of successful wireless attacks by responding immediately to intrusions. On the other hand, they execute responses without considering environmental factors and hence, results in execution of unsuitable responses causing negative effects to legitimate systems. This paper addresses this issue by proposing a wireless IDS with adaptive automated response mechanism named Context Aware Wireless Analyser (CAWAnalyser). CAWAnalyser selects an appropriate response based on a number of contextual factors, and invokes the selected response if the total impact of such response is lower than the total impact of the corresponding attack.

1 Introduction

In recent years, increasing numbers of organisations have deployed wireless networks based on the IEEE 802.11 standard. Although the standard provides a number of intrusion prevention measures, it overlooks certain security flaws and fails to provide adequate protection to wireless networks. In particular, the infamous weakness in the Wired Equivalent Privacy (WEP) encryption algorithm [1, 2] for example, has proved to be a failure in protecting cipher frames from unauthorised decryption. This has inspired the use of wireless intrusion detection systems (IDSs) as a second line of defence. While techniques for detecting wireless attacks have been an active area of research, little effort has been put into the study of wireless intrusion response and therefore further research is required in its own right.

Most existing wireless IDSs [3, 4] operate as notification systems or manual response systems. The former passively react to intrusions (e.g. email alert) while the latter requires the administrator intervention to counter intrusions. These systems have a delay between detecting and responding to an intrusion, allowing a time window of opportunity for intruders to succeed its attacks [5]. There are some wireless IDSs provide automated responses to intrusions. They respond immediately to attacks without human intervention and thus reducing the chances of successful attacks. However, most of these systems execute active responses without considering environmental factors and hence, results in invocation of unsuitable responses causing negative impact to legitimate systems.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 239–246, 2005.

To address this issue, we propose a wireless IDS with adaptive automated response mechanism named Context Aware Wireless Analyser (CAWAnalyser). The purpose of CAWAnalyser is to reduce the possibility of negative effects caused by automated active responses. It achieves this by first selecting an appropriate response based on a number of contextual factors, for example, victim sensitivity and IDS confidence, and then invoking the selected response if the total impact of such response is lower than the total impact of the corresponding attack.

This paper is organised as follows. In Section 2, we discuss related work. Section 3 discusses contextual factors that are used in CAWAnalyser. Section 4 presents the conceptual architecture of CAWAnalyser. In Section 5, we describe the process of response adaptation. The prototype console application of CAWAnalyser is described in Section 6. Section 7 concludes the paper.

2 Related Work

Within the past five years, there have been some research activities in the development of adaptive automated response systems. Carver, Hill and Pooch [6] have proposed a response framework named Adaptive, Agent-Based Intrusion Response System (AAIRS). AAIRS focuses on response decision mechanism in which multiple agents collaborate between each other to provide adaptive automated responses. In AAIRS, the Analysis Agent is responsible for making response decisions. It selects a suitable response based on the following contextual factors [7]: timing of an attack, attack type, attacker type (e.g. novice attacker / military rganization), suspicion strength, attack implications, environment constraints and success of previous responses. The AAIRS proposal provides a good framework to the development of an automated response system. It identifies several required components for automated responses, and a number of factors that are needed for making response decisions.

Another adaptive automated response framework is the Intrusion Monitoring System (IMS) [8]. In IMS, the Responder collaborates with the Detection Engine, the Collector, the Profiles and the Intrusion Specifications to provide adaptive responses. The selection of appropriate response is based on two categories of contextual factors [9]: incident related factors and IDS related factors. Incident related factors include attacked target, user account privilege, incident severity, incident threat, perceived perpetrator and time available to respond while IDS related factors include IDS confidence, alert status, response efficiency, information source, response impact and success of previous responses. In comparison to AAIRS, the IMS proposal provides a more comprehensive focus in the field of response decision making, identifying a wider range of contextual factors that need to be considered when making response decisions. However, some of the contextual factors are too generic and are less applicable to the development of adaptive automated wireless intrusion response systems.

3 Contextual Factors for Wireless Intrusion Response

There are numerous response methods that could be launched to counter wireless intrusions. Since each of them has different effects on attackers and legitimate users, some decision making ability is required in order to select a response method that has the highest possibility of stopping an attack, while causing the least negative effects on legitimate users. Section 2 has provided a number of research studies in this direction, and has identified a number of contextual factors that influence the response decision. We include some of those factors in CAWAnalyser's adaptive automated response mechanism. In addition, we introduce several new contextual factors. The following outlines the contextual factors used in CAWAnalyser for making response decisions. They are divided into two categories, namely factors that influence the impact of an attack and factors that influence the impact of a response:

Factors that influence the impact of an attack

- **Default attack impact**: The default severity of a particular type of wireless attack without considering damages caused by attack related factors.
- **Victim sensitivity**: How sensitive is the attacked system? Is it a critical business system, or a public wireless kiosk? In CAWAnalyser, the victim sensitivity is categorised into high, medium or low sensitivity.
- **Location sensitivity**: How sensitive is the wireless cell location in which the attack has occurred? Is the cell a public access cell, or a management cell? This is because a wireless specific Denial of Service (DoS) attack launched in a management cell may cause a more severe damage compare to the same attack launched in a public access cell. In CAWAnalyser, the location sensitivity is categorised into high, medium or low sensitivity.
- **Attacking state**: What is the current state of the attack? Is it in the attempted state or in the successful state? For example, an unauthorised connection attack is in the attempted state if a rogue wireless station is detected as sending association request frames, while it is in the successful state if the rogue station is detected as receiving a successful association response frame from legitimate access points. In CAWAnalyser, the attacking state is categorised into attempted or successful state.

Factors that influence the impact of a response

- **Default response impact**: The default negative impact of a particular response method without considering undesirable effects caused by response related factors.
- **Response efficiency**: How efficient was the response method in stopping previous attacks that are similar to this attack?
- **Attacker identity**: Is the attacker an outsider, or an insider? If the attacker is an insider, to what degree would the response disrupt the insider if the suspicious attack was, in fact a false alarm? In CAWAnalyser, the attacker identity is categorised into insider or outsider.
- **IDS confidence**: Can the alarm be trusted? How many false positives did the IDS generate in the past? In CAWAnalyser, the IDS confidence is categorised into high, medium or low confidence.
- **Administrator location**: Where is the administrator during the intrusion? Is the administrator inside or outside the premises? Although this contextual factor does not provide much influence to the impact of a response, it provides valuable information in switching between automated and manual execution of selected active responses. For example, a detected intrusion may warrant an immediate automated active response if the administrator is detected as outside the premises.

4 CAWAnalyser Architecture

The conceptual architecture of CAWAnalyser is shown in Fig. 1. It consists of a wireless IDS manager and several wireless IDS sensors distributed throughout a wireless network. The manager's responsibility is to centrally manage all the IDS sensors, detect wireless intrusions and make response decisions. The sensors are responsible for capturing wireless traffic and responding to intrusions as instructed by the manager.

In CAWAnalyser, a number of components collaborate between each other to detect and respond to wireless intrusions. The Decoder first decodes raw wireless frames into IEEE 802.11 frames and forwards them to the Context Detector. Upon receiving the decoded frames, the Context Detector inspects them to detect intrusions. When an intrusion is detected, the Context Detector reports the intrusion to the Context Responder. The Context Responder analyses the intrusion using various context information provided by the Context Handlers. At this stage, the Context Handlers interact with the Context Database, the Context Sensors and the Attack-Response Database to acquire information about those contextual factors described in section 3 and provide it to the Context Responder. After analysing the intrusion, the Context Responder makes appropriate response decisions based on the Context Policy. The Context Responder then responds passively or actively to the intrusion. If the intrusion requires active responses, the Context Responder instructs the Responder Agent to launch selected countermeasures. The following sections further elaborate on each of the components that are involved in CAWAnalyser intrusion response process.

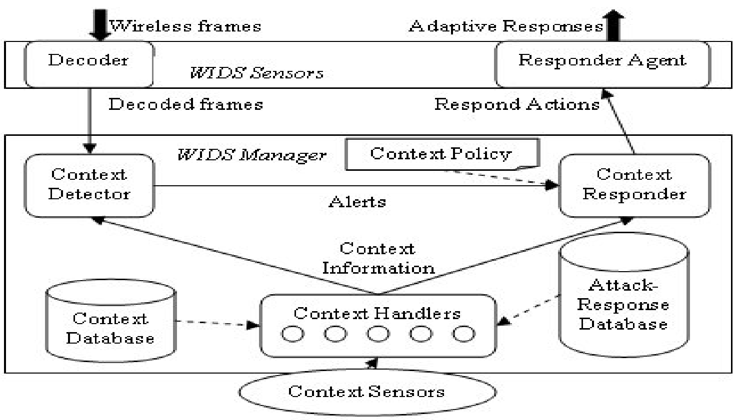


Fig. 1. CAWAnalyser Architecture

4.1 Context Detector, Context Database and Attack-Response Database

The Context Detector uses implicit information of wireless devices and users, along with known wireless intrusion patterns to perform misuse and anomaly detections [11]. When it detects an intrusion, it sends the intrusion details along with its confidence about the genuineness of the intrusion to the Context Responder. In addition, it informs the Context Responder about the state (attempted state or successful state) of the intrusion. For the Context Database, it stores static contextual records of authorised wireless devices, authorised users and cells. It is responsible for providing information about several contextual factors, namely location sensitivity, victim sensitivity and attacker identity. The Attack-Response Database contains mappings of known wireless attacks and the corresponding active response methods. Each of these mappings is associated with values of the following contextual factors, namely default attack impact, response efficiency and default response impact.

4.2 Context Sensors, Context Handlers and Context Policy

The Context Sensors is a collection of devices capable of providing up-to-date context information about a wireless device or a wireless user. It includes several location tracking systems such as Active Badge System [10]. These systems are responsible for providing information about the administrator location. For the Context Handlers, they are a collection of software components that are responsible for interacting with the Context Database, the Attack-Response Database and the Context Sensors. They acquire information about various contextual factors from these components and provide it to the Context Responder. The Context Policy contains numeric values of categories of the following contextual factors: victim sensitivity, location sensitivity, attacking state, attacker identity and IDS confidence. These values are assigned by the administrator and are used by the Context Responder to make response decisions. Fig. 2 (left) shows the console interface that is used to configure the Context Policy.

4.3 Context Responder and Responder Agent

The Context Responder is responsible for performing adaptive automated wireless intrusion responses. It contains a decision mechanism that selects appropriate responses based on a number of contextual factors (see Section 5 for more details). It also includes several passive response methods, including console display, file logging and email alert. In cases where active responses are required for an intrusion, it instructs the Responder Agent to respond actively to the intruder. Active response methods may include denial of service (DoS) response to rogue wireless station and DoS response to rogue access point.

5 Adaptive Intrusion Response

To provide automated wireless intrusion responses with minimum negative impact, the Context Responder makes adaptive response decisions based on a number of contextual factors. Its decision process consists of the following sequences:

1. Decide the mode of execution
2. Select an appropriate response
3. Determine the total impact of the selected response
4. Determine the total impact of the attack
5. Decide the execution of the selected response
6. Access the results of the executed response

1. **Decide the mode of execution.** In this sequence, the Context Responder obtains information about the location of the administrator from the Context Sensors, and determined whether or not to perform automated active responses. If the administrator is detected as inside the premises, it notifies and leaves the execution of active responses to the administrator, and will not go through the rest of the decision sequences. On the other hand, if the administrator is detected as outside the premises, it will perform automated active responses.

2. **Select an appropriate response.** In this sequence, the Context Responder first queries the Attack-Response Database (via the Context Handlers) to obtain available methods that could be used to counter the reported intrusion. After the search, it selects the response method that has the lowest default response impact value. If there are two available response methods having the same default response impact value, it selects the response method that has the highest response efficiency value.

3. **Determine the total impact of the selected response.** In this sequence, the Context Responder determines the total impact of the selected response with the considerations of response related factors. It first acquires information about the attacker identity and the IDS confidence from the Context Database and Context Detector. It then obtains numeric values from the Context Policy according to the categories of the acquired context information. Finally, it calculates the total impact of the selected response by summing up the selected response's default response impact value, the attack identity value and the IDS confidence value.

4. **Determine the total impact of the attack.** In this sequence, the Context Responder determines the total impact of the reported intrusion with the considerations of attack related factors. It first acquires information about the victim sensitivity, location sensitivity and attacking state from the Context Database and Context Detector. It then obtains numeric values from the Context Policy according to the categories of the acquired context information. Finally, it calculates the total impact of the reported intrusion by summing up the reported intrusion's default attack impact value (obtained from the Attack-Response Database), the victim sensitivity value, the location sensitivity value and the attacking state value.

5. **Decide the execution of the selected response.** In this sequence, the Context Responder decides whether or not to execute the selected response by comparing the total impact of the selected response with the total impact of the reported intrusion. If the value of the former is lower than the value of the latter, it executes the selected response.
6. **Assess the result of the executed response.** After invoking the selected response, the Context Responder checks whether or not the invoked response has successfully stopped the reported intrusion. This is done by checking whether there is any recurrence of the reported intrusion for a period of time. If it is still receiving alerts of the intrusion from the Context Detector, it decreases the response efficiency value of the invoked response, and repeats its decision sequences (from sequence 2 to 6) to select the next appropriate response.

6 Prototype Console Application

To demonstrate the adaptive automated response feature of CAWAnalyser, a prototype console application named Response Manager has been developed. As shown in Fig. 2, the Response Manager contains two elements: the Policy Editor and the Response Simulator. The Policy Editor is a console interface that allows the administrator to configure CAWAnalyser adaptive automated response mechanism. Using the interface, the administrator can modify numeric values of categories of contextual factors mentioned in Section 3 (e.g. IDS confidence, victim sensitivity).

The Response Simulator is a console interface that allows the administrator to simulate adaptive responses. Using the Response Simulator, the administrator can check whether or not an active response method for an attack type will be automatically executed under a certain condition. This is done by first selecting an attack type and the corresponding response method, changing the status of various contextual factors (e.g. changing victim sensitivity to high, changing attacker identity to insider, etc), and then clicking the simulate result button.

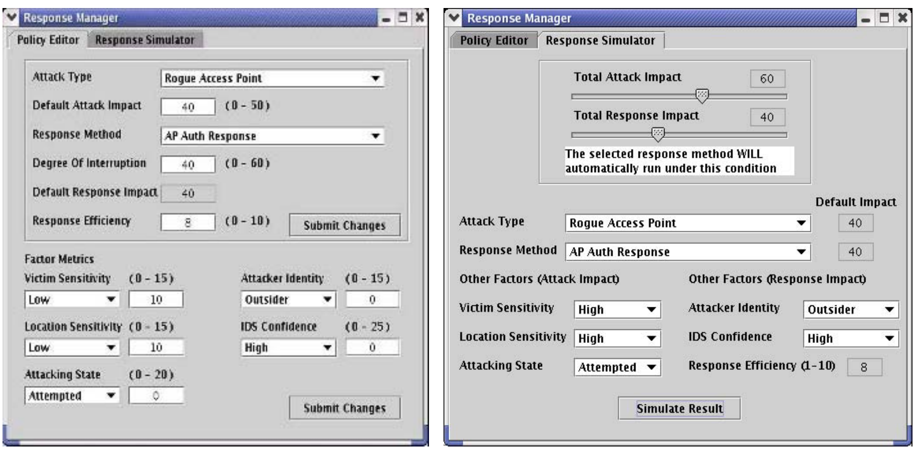


Fig. 2. Response Manager: Policy Editor (left) and Response Simulator (right)

7 Conclusion

We have presented CAWAnalyser, a wireless IDS with adaptive automated response mechanism. We have described several contextual factors that influence response decision making and have proposed an architecture that utilises such factors. We will focus on assignment of weightings to such factors in our future work. With the use of CAWAnalyser in wireless networks, response adaptation could be performed and thus reducing the chances of negative effects caused by automated active responses.

References

1. Fluhrer, S., Mantin, I. and Shamir, A.: Weaknesses in the key scheduling algorithm of RC4. LNCS Revised Papers from the 8th Annual International Workshop on Selected Areas in Cryptography, Springer-Verlag, London, UK (2001) 1-24
2. Stubblefield, A., Ioannidis, J. and Rubin, A.D.: Using the Fluhrer, Mantin, and Shamir Attack to Break WEP. Proceedings of Network and Distributed System Security Symposium, 6-8 February 2002, San Diego, California (2002)
3. Lackey, J., Roths A. and Goddard J.: Wireless Intrusion Detection. IBM Executive Strategy Report, http://www-1.ibm.com/services/strategy/files2/wireless_intrusion_detection.pdf (2003)
4. AirDefense Inc.: AirDefense Guard. http://www.airdefense.net/products/airdefense_ids.shtm (2004)
5. Cohen, F.: Simulating Cyber Attacks, Defenses, and Consequences. The Inforsec Technical Baseline studies, http://all.net/journal/ntb/simulate/simulate.html (1999)
6. Carver, C.A. and Pooch, U.W.: An Intrusion Response Taxonomy and its Role in Automatic Intrusion Response. Proceedings of the 2000 IEEE Workshop on Information Assurance and Security, 6-7 June 2000, United States Military Academy, West Point, New York (2000) 129-135
7. Carver, C.A., Hill, J.M.D. and Pooch, U.W.: Limiting Uncertainty in Intrusion Response. Proceedings of the 2001 IEEE Workshop on Information Assurance and Security, 5-6 June 2001, United States Military Academy, West Point, New York (2001) 142-147
8. Papadaki, M., Furnell, S., Lines, B. and Reynolds, P.: Operational Characteristics of an Automated Intrusion Response System. Proceedings of the 7th IFIP TC-11 TC-6 International Conference (CMS 2003), 2-3 October 2003, Torino, Italy, LNCS 2828 (2003) 65-75
9. Papadaki, M., Furnell, S.M., Lee, S.J., Lines, B.M. and Reynolds, P.L.: Enhancing response in intrusion detection systems. Journal of Information Warfare, Vol. 2, No. 1 (2002) 90-102
10. Want, R., Hopper, A., Falcao, V. and Gibbons, J.: The Active Badge Location System. ACM Transactions on Information Systems, Vol 10 (1992) 91-102
11. Gan, C.H., Zaslavsky, A. and Giles, S.: CAWAnalyser: Enhancing Wireless Intrusion Detection with Runtime Context-Awareness. Proceedings of the 2004 Australian Telecommunications Networks & Application Conference, 8 – 10 December 2004, Sydney, Australia (2004) In Press

Evaluation of Transport Layer Loss Notification in Wireless Environments

Johan Garcia and Anna Brunstrom

Karlstad University,
Dept. of Comp. Sci., 651 88 Karlstad, Sweden
Tel: +46-54-7001789 Fax: +46-54-7001446
{johan.garcia, anna.brunstrom}@kau.se

Abstract. Residual bit-errors in wireless environments are well known to cause difficulties for congestion controlled protocols like TCP. In this study we focus on a receiver-based loss differentiation approach to mitigating the problems, and more specifically on two different loss notification schemes. The fully receiver-based 3-dupack scheme uses additional dupacks to implicitly influence the retransmission behavior of the sender. The second TCP option scheme uses a TCP option to explicitly convey a corruption notification. Although these schemes look relatively simple at first glance, when examining the details several issues exist which are highlighted and discussed. A performance evaluation based on a FreeBSD kernel implementation show that the TCP option scheme works well in all tested cases and provides a considerable throughput improvement. The 3-dupack scheme also provide performance gains in most cases, but the improvements varies more between different test cases, with some cases showing no improvement over regular TCP.

1 Introduction

The increased use of wireless links to transport TCP/IP traffic, not only for access links but also in ad-hoc and sensor networks, introduces additional difficulties for transport protocol design. The physical characteristics of a radio channel, such as interference and fading, lead to far more bit-errors being generated compared to a wired connection. Often physical and link-layer solutions such as forward error correction (FEC) and link level retransmissions are used to handle the high bit error rates generated by the radio channel. However, there are scenarios where it is not practical or possible for the lower layers to eliminate all bit-errors, typically due to power, complexity or delay constraints. Consequently, transport layer behaviors that consider bit-errors at the transport layer have started to appear. One example is UDP-Lite [1] that uses partial checksums and allows the delivery of data with bit-errors to the application. Another example is the new Datagram Congestion Control Protocol (DCCP) [2] that can use a partial checksum option to also differentiate between congestion and corruption losses. In the case of TCP, the existence of residual bit-errors has been shown to severely decrease the performance [3]. Many TCP-specific enhancements have been proposed to improve the performance over wireless links that corrupt packets by introducing bit-errors. Examples

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 247–257, 2005.

include using split connections [4], TCP-aware local retransmissions (snoop) [5] or loss differentiation at the transport layer [6]. A number of TCP variants that consider the problem of wireless over TCP exists, including TCP Westwood [7] and TCP Real [8]. For many wireless environments the problems surrounding mobility induced losses and intermittent disconnections are also relevant and challenging, but these are outside the scope of this paper.

In this paper we focus on the loss differentiation approach to handle corruption errors on a wireless link, and more specifically on the performance of two loss notification schemes that are used together with receiver-based loss differentiation. In Sect. 2 we provide a background on loss differentiation as related to the loss notification work in this paper. Section 3 provides a description of design and protocol issues for the two examined loss notification mechanisms, 3-dupack and TCP option. The 3-dupack mechanism requires changes only to the receiver side, and when used together with checksum-based loss differentiation it provides a solution for environments where only the receiver side can be modified. If modifications can be done also at the sender side the more effective TCP option notification mechanism can be used. In Sect. 4 we present experimental results on the performance of the two mechanisms. The results show considerable improvements in most cases, but also that there are cases where the 3-dupack notification provides no benefit. The paper ends with conclusions and future work in Sect. 5.

2 Loss Differentiation

We use the term *loss differentiation* to mean the process of deciding the cause of a lost packet. Many diverse loss differentiation schemes exist but a high-level classification can be made into schemes that require support from the infrastructure such as base stations, and those that only need end-host changes. End-host based loss differentiation can be performed either at the sender or at the receiver. A requirement for *loss notification* arises if the loss differentiation is not performed where the congestion control is performed (i.e. at the sender). Loss notification is the process of communicating the results of loss differentiation to a sender. Loss notification is thus mainly interesting for receiver based loss differentiation schemes, although it could also be used in conjunction with schemes that require infrastructural changes such as [9, 10]. Although sender based loss differentiation schemes [11, 12, 13] do not need loss notification, their precision in correctly classifying losses is lower than for receiver based schemes.

Considering receiver based loss differentiation schemes, several proposals exist. Biaz et al [14] propose a scheme which examines the inter-arrival times and is based on an assumption that the wireless hop is the constraining link. Biaz' scheme was later improved by Cen et al [15] who evaluated an improved version along with two other schemes, Spike and ZigZag as well as hybrid schemes that use heuristics to select the most appropriate base scheme for the current environment. Zhang [16] also use inter-arrival times to infer loss causes, but uses wave patterns to improve the differentiation mechanism.

Checksum-based schemes are based on the observation that wireless losses often do not actually lose the whole packet on the link, but rather corrupt a number of bits in it. This corruption will then lead to checksum failure. When a checksum control function detects an erroneous checksum the packet is discarded. The fact that data was discarded due to a checksum error is, however, not known outside the checksum control function. By informing the transport layer that a checksum error has occurred, a lost packet can be classified as a wireless loss and not a congestion loss. In [3], Balakrishnan et al examine several mechanisms to improve TCP performance. With regards to loss differentiation, the possibility of using checksums to differentiate between losses is discussed, and a simplified implementation is evaluated. Balan et al [17] describe TCP HACK, a similar scheme except that it uses a TCP option containing a checksum for the TCP header. When a corrupted packet arrives, a correct header checksum guarantees the integrity of the header information. For the experiments in this paper we use a checksum based loss differentiation mechanism that uses only the checksums already present. For further details and a discussion on the advantages and limitations of this specific checksum-based loss differentiation, see [6].

3 Loss Notification Mechanisms

Since it is the sender that performs the congestion control, the outcome of receiver-based loss differentiation must in some way be communicated to the sender. In this section we propose two loss notification mechanisms that differ in how they convey the differentiation information and how they ultimately influence the sender's congestion control. The design of a loss notification mechanism is to a great extent decoupled from the loss differentiation method used. Although the actual implementation presented in this paper is done together with checksum based loss differentiation, the loss notification mechanisms described in this section can also be used with other receiver based loss differentiation mechanisms.

3.1 3-Dupack Notification

The 3-dupack loss notification mechanism requires changes only to the receiver side. When used in conjunction with a receiver based loss differentiation scheme, such as the checksum based one used for the experiments, it is possible to get a solution that is purely receiver-based and does not require any changes at the sender side. This is a clear deployment advantage. The working principle of the 3-dupacks mechanism is to immediately generate three dupacks when a corruption loss is detected, not waiting for the reception of additional packets in order to send out the dupacks as regular TCP would. While the basic idea is simple, several issues emerge when looking into the details as discussed in the rest of this subsection.

The detection of a corruption loss helps to resolve the ambiguous cause for a hole in the sequence numbers of received packets. A hole can be caused either by a packet loss or by packet reordering in the network. Since a loss in fact has been detected, and attributed to corruption, a hole must be present in the sequence number stream, thus disambiguating the cause for the hole. Consequently, there is no need to wait for additional packets before the receiver can be sure that the cause indeed is packet loss. The 3-dupack mechanism thus immediately generates three dupacks in order to implicitly inform the sender that there has been a loss. Since this scheme is receiver based, no modifications are done at the sender to adapt the congestion behavior to different loss causes. The sender will do a regular retransmission with congestion window halving when it receives the three dupacks. However, since the three dupacks are generated instantaneously and not when packets in flight are received, the time until retransmission will be lower when using the 3-dupack scheme. Additionally, the extra dupacks will add to the window inflation done during recovery, allowing earlier data flow. The usefulness of 3-dupacks is further amplified for short connections. Shorter connections spend a relatively larger amount of the connection lifetime in *fast-retransmit inhibition.* When the sender cannot send enough packets to generate the necessary dupacks, it is in fast-retransmit inhibition. The first inhibition occurs during the start of the connection, when the sender's congestion window may not be large enough to allow for four outstanding packets which makes regular fast retransmit based on three dupacks impossible. The second inhibition occurs at the end of the connection, when there are less than three packets left to send after a loss. Regular TCP must handle this with a time-consuming timeout. The 3-dupack scheme does not have these inhibitions for losses that are caused by corruption. For congestion losses these inhibitions exist both for regular TCP and the 3-dupack scheme, although we note that limited transmit [18] is a solution for the first inhibition for congestion and corruption losses alike.

One weakness of the 3-dupack scheme is that there exists a case where false retransmissions can cause a small amount of unnecessary network load. These *reordering-induced false retransmissions* can only occur in networks which have network reordering together with link corruption. Assume that there has been a reordering in the network so that packets are received at the last link in the order P1 P2 P4 P3 P5 P6 P7, and that packet P4 is corrupted when traversing the last link.[1] Generating 3-dupacks based only on the state of the TCP connection would in this case cause the dupacks to be for packet P3 instead of packet P4. This would in turn cause an unnecessary retransmission of P3 that consumes network resources in vain. The retransmission of P4 would occur after receiving the P5-generated dupack, assuming that the sender uses the NewReno behavior for multiple losses in one window. For loss differentiation mechanisms that provide the sequence number of the corrupted packet there is a possibility to address the reordering-induced false retransmissions. By checking for a mismatch between the sequence number of the corrupted packet and the next expected sequence number, sending of 3-dupacks for a reordered packet can be avoided. The behavior would then become the same as for regular TCP.

[1] Note that in this example the sequence number is per packet, whereas in actual TCP it is expressed in bytes.

The 3-dupack modification provides improved performance, but the discussion above also indicates that there is a weakness and that the exact effect of using the 3-dupack scheme to some extent is dependent of the specific TCP implementation used at the sender.

3.2 TCP Option Notification

The TCP option loss notification scheme uses a new TCP option to explicitly inform the sender that a packet has been lost due to corruption. This scheme requires modifications to both the sender and the receiver side in order to be able to negotiate and use the option. Additionally, modifications are made to the sender so that it only retransmits the packet reported as corrupt, without performing window halving congestion avoidance. This is an advantage over the 3-dupack scheme where all losses cause the same sender congestion avoidance behavior.

The format of the option is shown in Fig. 1. A loss diff counter is used to identify loss differentiation events. The loss diff sequence number, i.e the sequence number of the corrupted packet is also included in the option.

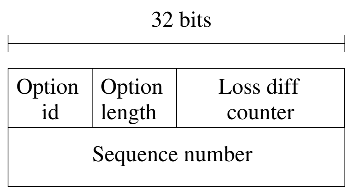


Fig. 1. TCP loss notification option

The use of this new option is negotiated between the communicating parties at connection setup. After a successful negotiation, the option is included only in duplicate acks, minimizing the extra header overhead. When a corruption loss is detected by the client a dupack is immediately sent, with the option included. In addition the option will be present in the later dupacks triggered by the receipt of out-of-order packets. Upon receipt of the option, the server side retransmits data with the sequence number of the option, without performing any window halving. The server will perform this the first time it sees an option with a new loss diff counter value. This ensures correct behavior even if the immediately generated dupack is lost.

However, when considering the details, this mode of operations leads to a *retransmission ambiguity problem*; that is when the 3rd dupack is received, retransmission should not occur for a packet that has been previously retransmitted due to the reception of a

```
| P1  |  LL  |  LL  |  P4  |  P5  |  P6  |  P7  |   :Rcvd by receiver
| A2  | D2L2 | D2L3 | D2L3 | D2L3 | D2L3 | D2L3 |   :Sent by receiver
| P2  |  P2  |  P3  | ???? |      |      |      |   :Sent by sender
```

Legend: Px=Packet with seq nr x, LL=link error packet, Ax=Ack with next expected packet x, DxLy=Duplicate ack for packet x including loss notification option for packet y

Fig. 2. Retransmission ambiguity problem

TCP option. A simple check against the loss diff sequence number is not sufficient as shown in Fig. 2.

If there is no state information at the sender indicating that it has already retransmitted P2, then the three dupacks received should cause it to do a regular fast retransmit of P2 at the ???? mark, which would include an inappropriate window halving. To solve the retransmission ambiguity problem an additional data-structure that holds information on unacked packets that have been previously retransmitted due to a TCP loss diff option is necessary. When the 3rd dupack is received the loss diff data-structure is checked to see if the dup-acked value is present, and if so nothing is done. If the dup-acked value is not present, a regular fast retransmit with window halving is performed.

Another potential issue, *congestion loss masking*, may arise if packets smaller then the maximum segment size (MSS) are used and Nagle's algorithm is disabled. If less-than-MSS sized segments are sent, there is a risk that a congestion loss will stay undetected when a corruption loss of a segment of size $x|x < MSS$ is directly followed by a congestion loss of a segment of size $y|x + y \leq MSS$. The receiver is aware of the corruption loss because it is reported by the loss differentiation mechanism, and a dupack with an option for the corrupted packet is immediately sent by the receiver side. When the sender side receives this dupack it immediately performs a retransmission. If this retransmission has a size of at least $x + y$, the congestion loss becomes "masked" by the retransmission caused by the preceding corruption loss, and no congestion avoidance behavior is applied for the congestion loss. For TCP applications this normally does not happen since the segmentation function in the sender always fills up the segments to the MSS if there is data in the send queue. For applications like Telnet, where the application does not generate enough data to fill a MSS, congestion loss masking is not a problem anyway since the sending rate will be limited not by the congestion window but by the availability of data to send from the application. The masking issue can also be solved by retaining the IP-layer packet size information for corrupted packets.

For loss differentiation mechanisms that cannot guarantee the integrity of the sequence number of the corrupted packet, *sequence number validation* at the sender side can be useful. The validation ensures that the loss diff sequence number reported in the option is between the highest unacked byte (snd_una) and the highest sent byte (snd_max). Some TCP implementations already keep the sequence numbers of the outstanding packets in a circular buffer. This data-structure can be reused to check that the sequence number received in the option matches to a sequence number corresponding to the start of a packet. If sender side validation is not used, bit-errors in the sequence number can potentially cause retransmission of data that are not needed. All in all the TCP option scheme provides better and more consistent performance than the 3-dupack scheme, but requires modifications at both the receiver and sender sides.

3.3 Behavior Example

The effect of the two presented loss notification schemes are illustrated in Figs. 3 to 5. The figures show the situation at the server side with the x-axis being wall time and the y-axis sequence numbers. The thin continuous lower line shows the latest acknowledged sequence number. A step in this line signifies the receipt of an acknowledgment packet, raising the acknowledged sequence number and allowing packets to be sent. Additionally, dupacks are shown as ticks in the continuous line. Outgoing packets are shown as vertical bars between arrows. When a loss has occurred, the thin lower line will be horizontal until the missing packet has been retransmitted and the ack of the retransmission is received. This ack creates a large step since it acknowledges not only the retransmitted packet, but also data sent after the initial missing packet. As can be seen from the figures, the 3-dupack scheme allows the retransmission to take place earlier, and allows more packets to be sent during the retransmission ack wait. The TCP-option scheme keeps data flowing nicely all the time while waiting for the retransmission ack, this is a consequence of the absence of congestion avoidance behavior for corruption losses.

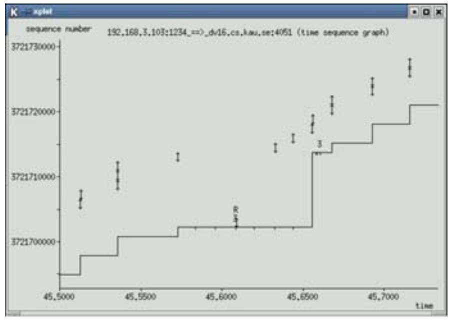


Fig. 3. Regular FreeBSD 4.5 TCP behavior

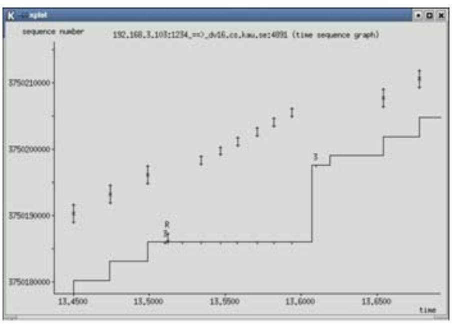


Fig. 4. With 3-dupacks loss notification

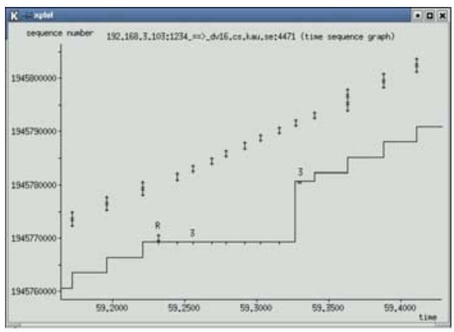


Fig. 5. With TCP-option loss notification

4 Experimental Evaluation

In order to get an understanding of the performance of the two loss notification schemes, we implemented both checksum based loss differentiation and the two notification schemes in the FreeBSD 4.5 kernel. Experiments were performed with a client and a server connected via a router/emulator. The router/emulator used Dummynet [19] to generate bit-errors and apply bandwidth restrictions and delay as appropriate for the test case. Three different bandwidths, two different delays and uniformly distributed bit

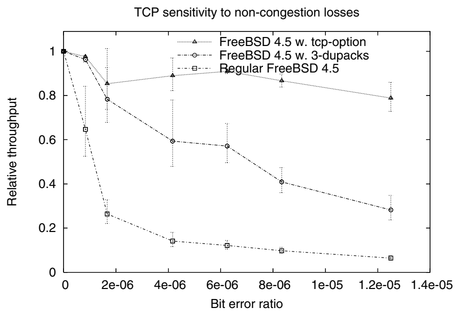


Fig. 6. Relative performance (1 Mbps, 10 ms)

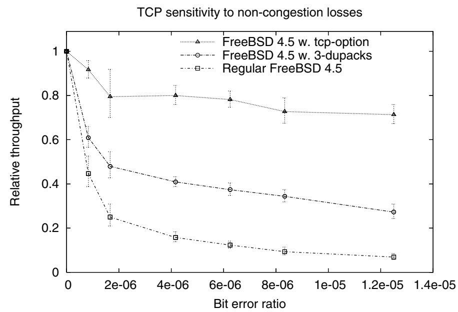


Fig. 7. Relative performance (1 Mbps, 50 ms)

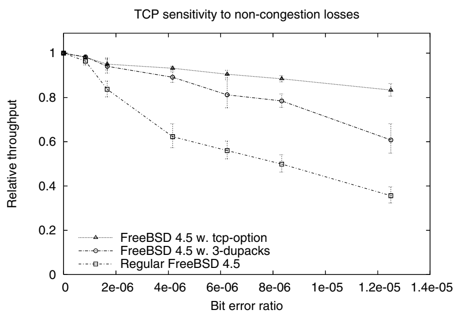


Fig. 8. Relative performance (160 kbps, 10 ms)

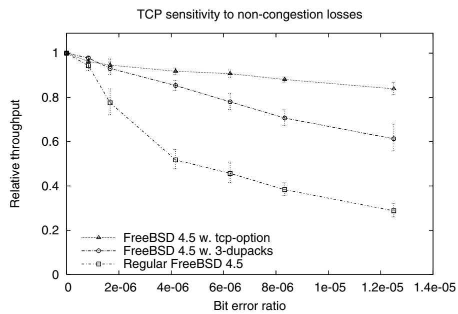


Fig. 9. Relative performance (160 kbps, 50 ms)

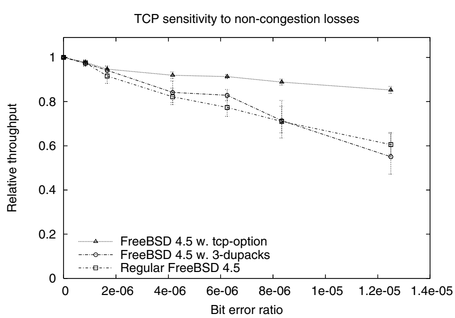


Fig. 10. Relative performance (40 kbps, 10 ms)

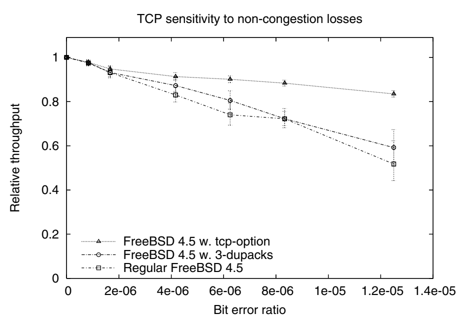


Fig. 11. Relative performance (40 kbps, 50 ms)

errors with seven bit error rates (BER) varying between 0 and $1.9 * 10^{-5}$ were used. The BER values were chosen so that they correspond to packet loss ratios (PLR) of approximately 1, 2, 5, 7.5, 10 and 15 percent for the packet size used (1500 bytes). To be noted is that we used a modified version of the Dummynet FreeBSD kernel code that enabled us to generate bit errors using bit error pattern files. The same pattern file was reused when testing different schemes, thus comparing them for exactly the same channel conditions. The time to transfer 250 kbyte was measured, and to avoid the risk of a specific bit error pattern skewing the results, 30 different randomly generated bit error pattern files were used for each parameter combination. The receiver window was set to its default value of 64 kbyte and the buffering before the constrained link was set to be large enough not to overflow. Since the focus of the experiment was on the behavior over a link with bit-errors rather than congestion losses, no competing traffic was generated thus leaving wireless link errors as the only cause for lost packets.

The results are shown in Figs 6-11. The figures show the throughput of the flow relative to the throughput of the flow when no bit errors occur. The point estimates show the mean over the 30 different bit error patterns and the 95% confidence intervals. From the figures it can be seen that loss differentiation/notification is beneficiary, and that the relative benefit increases as the loss rate increases. The general trend is that regular TCP becomes more sensitive to losses at high bandwidths (BW) and, to a lesser extent, high link delays (D). Both increased bandwidth and increased delay lead to an increase in the bandwidth-delay product (BW*D). A larger BW*D requires a larger average congestion window size to keep the pipe filled. A larger average window can only be obtained if the distances between losses are large enough, i.e. the sensitivity to losses increases. The minimum retransmission timeout (RTO) is also a factor that influences the results, and it has the greatest impact on the high bandwidth case. When a retransmission of a previously lost packet is itself lost, this loss can only be detected by a timeout. There is a fixed minimum time for the RTO. Since the same amount of data was transferred for all the bandwidths, the minimum RTO value was proportionally larger for the high bandwidth configurations than for the low.

When comparing the 3-dupack and the TCP option schemes it is clear that the 3-dupack scheme performance is lower and varies much more than for the TCP-option. The benefit of the 3-dupack scheme seems to be the largest for the 1Mbps and 160 kbps cases. Even though the 3-dupack scheme can improve the TCP throughput by well over 100% in many instances for the 1Mbps cases, the 3-dupack scheme only utilizes a smaller fraction of the available bandwidth. For the low bandwidth 40kbps cases, the 3-dupack scheme utilizes a larger fraction of the bandwidth but so does TCP, and thus 3-dupack provides no performance gain in these cases.

5 Conclusions

We have described two approaches to loss notification, the implicit 3-dupacks scheme and the more explicit TCP-option approach. The 3-dupack scheme has the advantage that it is simple and requires no modifications to the sender side. Its weaknesses include worse performance than the TCP option and that the results to some extent are dependent on the specific sender TCP implementation. The TCP-option scheme, on the other hand, provides very good performance but requires sender side modifications. Several issues were identified and discussed such as the retransmission ambiguity and the less-than-MSS problem. Experiments using FreeBSD kernel implementations were performed to obtain performance results. The results show a considerable performance increase in most cases. However, for low bandwidth and low delay links, which are not uncommon in ad-hoc and sensor networks, the performance gain from using loss differentiation was not as large as in the high bandwidth cases. The 3-dupack scheme even showed no performance increase for the low bandwidth cases. Future work include further evaluation using more complex topologies and with competing traffic.

References

1. Larzon, L.A., Degermark, M., Pink, S., Jonsson, L.E., Fairhurst, G.: RFC 3828: The lightweight user datagram protocol (udp-lite) (2004)
2. Kohler, E., Handley, M., Floyd, S., Padhye, J.: Datagram congestion control protocol (DCCP). draft-ietf-dccp-spec-07.txt, Work in progress (2004)
3. Balakrishnan, H., Padmanabhan, V.N., Seshan, S., Katz, R.H.: A comparison of mechanisms for improving TCP performance over wireless links. IEEE/ACM Transactions on Networking **5** (1997) 756–769
4. Bakre, A., Badrinath, B.R.: I-TCP: Indirect TCP for mobile hosts. 15th International Conference on Distributed Computing Systems (1995)
5. Balakrishnan, H., Seshan, S., Amir, E., Katz, R.H.: Improving TCP/IP performance over wireless networks. In proc. 1st ACM Int'l Conf. on Mobile Computing and Networking (Mobicom) (1995)
6. Garcia, J., Brunstrom, A.: Checksum-based loss differentiation. Proceedings 4th IEEE Conference on Mobile and Wireless Communications Networks (MWCN 2002), Stockholm, Sweden (2002)
7. Casetti, C., Gerla, M., Mascolo, S., Sansadidi, M., Wang, R.: TCP Westwood: End-to-end congestion control for wired/wireless networks. Wireless Networks **8** (2002) 467–479
8. Zhang, C., Tsaoussidis, V.: TCP Real: Improving real-time capabilities of TCP over heterogeneous networks. Proceedings of the 11th IEEE/ACM NOSSDAV (2001)
9. Balakrishnan, H., Katz, R.: Explicit loss notification and wireless web performance. In: Proceedings Globecom Internet Mini-Conference, Sydney, Australia. (1998)
10. Chen, W.P., Hsiao, Y.C., Hou, J.C., Ge, Y., Fitz, M.P.: Syndrome: a light-weight approach to improving TCP performance in mobile wireless networks. Wireless Communications and Mobile Computing (2002) 37–57
11. Chengpeng, F.: TCP Veno: End-to-end congestion control over heterogeneous networks. Ph. D thesis, Chinese University of Hong Kong (2001)
12. Kim, T., Lu, S., Bharghavan, V.: Improving congestion control performance through loss differentiation. Proceedings International Conference on Computers and Communications Networks (ICCCN99), Boston, USA (1999)
13. Liu, J., Matta, I., Crovella, M.: End-to-end inference of loss nature in a hybrid wired/wireless environment. CS Dept Technical report 2002-008, Boston University (2002)
14. Biaz, S., Vaidya, N.: Discriminating congestion losses from wireless losses using inter-arrival times at the receiver. IEEE Symposium ASSET'99, Richardson, TX, USA (1999)

15. Cen, S., Cosman, P., Voelker, G.: End-to-end differentiation of congestion and wireless losses. Proc. Multimedia Computing and Networking (MMCN2002), San Jose, CA (2002) 1–15
16. Zhang, C., Tsaoussidis, V.: Error differentiation with measurements based on wave patterns. Computer Communications **27** (2004) 989–1000
17. Balan, R.K., Lee, B.P., Kumar, K.R.R., Jacob, L., Seah, W.K.G., Ananda, A.L.: TCP HACK: A mechanism to improve performance over lossy links. Computer Networks **39** (2002) 347–361
18. Allman, M., Balakrishnan, H., Floyd, S.: RFC 3042: Enhancing TCPs loss recovery using limited transmit (2001)
19. Rizzo, L.: Dummynet: A simple approach to the evaluation of network protocols. ACM Computer Communication Review **27** (1997) 31–41

End-to-End Wireless Performance Simulator: Modelling Methodology and Performance

Sung-Min Oh, Hyun-Jin Lee, and Jae-Hyun Kim*

School of Electrical and Computer Engineering,
Ajou University, Suwon, Korea
{smallb01, l33hyun, jkim}@ajou.ac.kr

Abstract. To evaluate the application-level performance in wireless networks, we build a wireless performance simulator which include a application traffic characteristic, network architecture, network element details and protocol features. We also develop the simulation modelling methodology using Lindley's recursion method to reduce the number of simulation events. Using the simulator, we assess the user-perceived application-level performance of the voice and web browsing service in the cdma2000 network for the wireless technology migration from 2.5G to 3G+. The main conclusion of this paper is that end-to-end application-level performance is affected by various elements and layers of the network. Thus, it must be considered in all phases of the development process.

1 Introduction

cdma2000 3G-1X RTT (Radio Transmission Technology) was on the market from 2001. Many wireless service providers have been considering to migrate from the circuit to the packet switched service in the 3G wireless technology. Therefore, user performance studies for cdma2000 were published in many papers [1, 2, 3, 4, 5].

In [1], the data service performance was evaluated for 3G-1X RTT system but an alternative architecture or voice service was not addressed. In [2], the TCP performance was presented in a wireless interface but an end-to-end performance was not included. Most of the papers addressed the wireless channel throughput or sector throughput and some of the papers studied QoS strategies in cdma2000 [3, 4]. However, a very few studies considered the whole network architecture. The user-perceived application-level performance should be considered in an end-to-end reference architecture, which includes a Radio Access Network (RAN), a Core Network (CN) and a data center. Otherwise we can get only partial information on the application-level performance. To assess the characteristics of the user-perceived application-level performance of different QoS service classes for alterative transport technologies and wireless technology evolution scenarios, we propose an end-to-end performance simulator for 2.5G or 3G+ networks.

* The work was supported in part by IITA project.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 258–267, 2005.

In this paper, we describe the end-to-end performance simulation model and methodology that we used to build the cdma2000 network. We model all the protocol layers from the physical through the application layer. We also model details of the packet handling characteristics of each network element along the path. Foreground and background traffic loads are generated to represent a specific application environment. We also address application-level performance issues in terms of wireless technology evolution from 3G-1X RTT to 3G-1X EV and the transport technology evolution from ATM to IP. Some highlights of the wireless performance simulator are:

- Models end-to-end reference connections
- Models cdma2000 applications and services
- Models user-plane traffic
- Impacts of mobility will be approximated
- Models each network element
- Models cdma2000 packet flow and detailed protocol stacks

2 Network Simulation Models

2.1 Reference Architecture and Connection Models

We study the performance modelling for the 2.5G and 3G+ networks. The 3G-1X system supports data rates from 9.6 kbps to 2.4 Mbps[6]. Fig. 1 shows a reference network architecture model for the cdma2000. The reference network architecture can be considered as four different networks; RAN , CN, internet and data center. RAN may include Mobile Terminal (MT), Base Station Transmission System (BTS), Base Station Controller (BSC), Mobile Switching Center (MSC) and ATM or IP concentrators. CN includes ATM or IP routers and Packet Data Serving Nodes (PDSN). The data center network can be composed of three zones to protect servers from hacking or virus attack; a public, a DMZ (Demilitarized Zone), and a secure zone. Each zone can be protected by firewalls as shown in Fig. 1.

2.2 Protocol Architectures and Models

In this paper, we consider two transport technologies such as ATM and IP. For ATM transport scenarios, a BTS chops a reverse link traffic packet into ATM cells and transmits them to MSC or Radio Network Controller (RNC) (for the ALL IP scenario). The voice traffic uses AAL2 and the data traffic uses AAL5 layers respectively in RAN. For the ALL IP transport scenarios, BTS transmits a IP packet on the top of T1 and IP router converts it to Ethernet packet and sends it to MSC. The detailed protocol stack for the IP protocol architectures are shown in Fig. 2. To assess the application-level performance, we implement all the protocol stacks in Fig. 2 except wireless channel model. To simulate the wireless channel error, we used the following link-level simulation results. The channel model used in this paper is based on the models specified in 3G 1X-RTT.

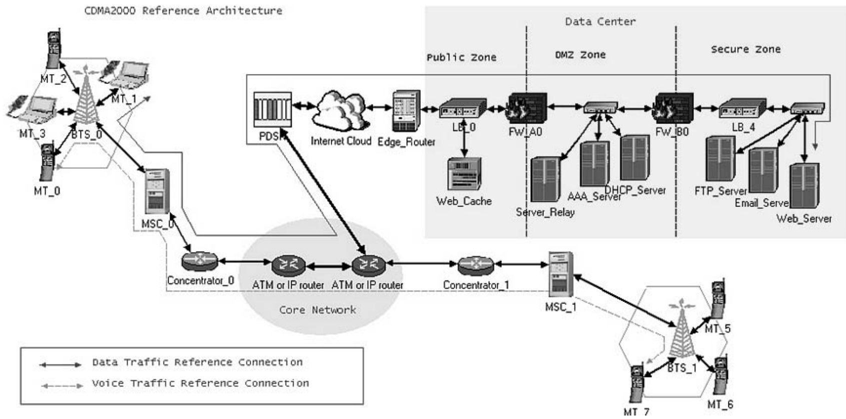


Fig. 1. Reference network model for cdma2000

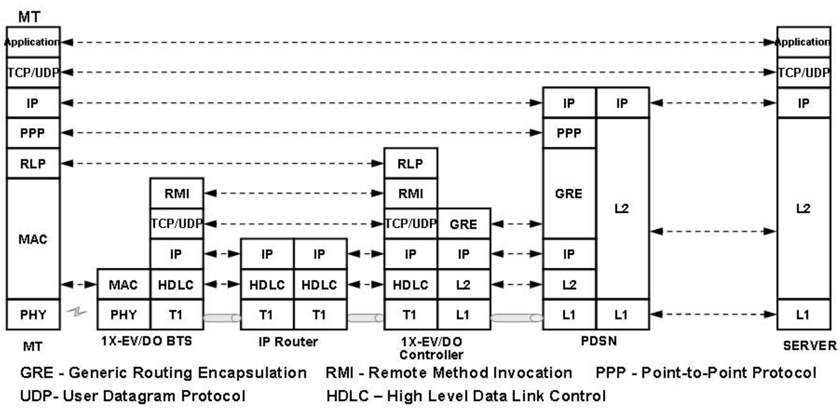


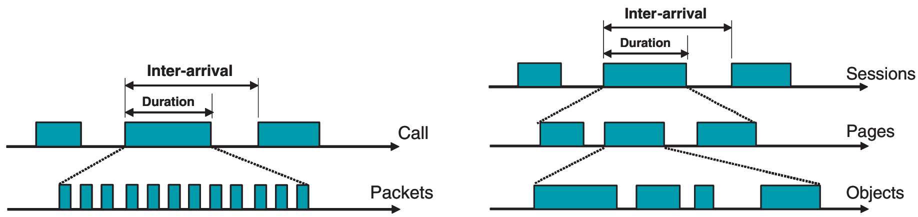
Fig. 2. Protocol stack model for All IP CDMA 2000

For the link-level simulation, we use a traced file which contains frame errors when the target frame error rate is fixed at 1%, 4% or 10%. These error data are time co-related for each frame upon channel model.

3 Voice and Web Service Traffic Models

3.1 Foreground Traffic Load Models

We use a voice traffic model and a web browsing traffic model for the applications in the paper. The voice traffic is generated by two hierarchical structures; call and packet level. The call level model consist of a a sequence of ON and OFF periods as shown in Fig. 3(a). Each durations of ON and OFF periods are exponentially distributed with the mean of 3 sec. (activity factor is 0.5). During the ON period MT generates Enhanced Variable Rate Codec (EVRC) 8 kbps



(a) Voice traffic model

(b) Web browsing traffic model

Fig. 3. The hierarchical structure for voice and web services

voice traffic packets[7]. We use the 3GPP2 standard traffic model for the web browsing service[8]. An example of the design model for the web browsing service is illustrated in Fig. 3(b). We characterize the arrival of page requests within a session, and the number of objects and their sizes for each page. The detailed statistics can be found in Table 1. Other applications of interest can be modelled similarly. The simulation will measure the performance of these foreground applications in detail.

3.2 Background Traffic Load Models

The background traffic load must represent the applications that run in the network so that the impact of the background traffic load, which uses on the foreground traffic, is accurately accounted for. However, the impact on the simulation run time precludes detailed application-level models. After reviewing an enormous number of different methods, such as statistical models for data traffic (long range dependant type)[9] and traffic analysis and synthesis[10], we decide to use the method of using traced traffic to get the effect of the background traffic load. The first step is to collect a detailed packet trace for 1000 simultaneous application sessions using the simulator. This traced file is then scaled to match the desired mean rate for a given application. This approach improved the simulation run-time performance, but it was still too slow to run large scale network simulations. Thus we use the traced file to simulate a virtual packet load by calculating the delay effect in the buffer instead of generating the background traffic packet by packet one by one. To calculate the packet delay effect, we used Lindley's recursion method and extended it to account for the impact of multiple queues and queue scheduling disciplines. Lindley's recursion equation is given by

$$W_q^{(n)} = \begin{cases} W_q^{(n-1)} + S^{(n-1)} - T^{(n-1)} & , \ if \quad \left(W_q^{(n)} + S^{(n)} - T^{(n)} > 0\right) \\ 0 & , \ otherwise \end{cases} \quad (1)$$

where, $W_q^{(n)}$ and $W_q^{(n-1)}$ mean waiting times of the n^{th} packet and $(n-1)^{th}$ packet respectively. $S^{(n-1)}$ denotes the service time of the $(n-1)^{th}$ packet and

$T^{(n-1)}$ means the inter-arrival time between the $(n-1)^{th}$ and n^{th} packets. The packet delay calculation algorithm for multiple priority queue is as following.

- Definition
  - $F_0, F_1, \cdots, F_i, \cdots, F_p$: Background traced file with the priority 0, 1, $\cdots$,i, $\cdots$, p. 0 is highest priority and the F_i is the traced file for the current reference packet with priority i.
  - *t_last*: the time that the $(n-1)^{th}$ reference packet is arrived
  - *t_arv*: the time that the n^{th} reference packet arrived
  - *t_ia*: inter-arrival time between the $(n-1)^{th}$ packet and the n^{th} reference packet
  - *t_wait*: waiting time for the n^{th} reference packet which calculated by Lindley equation
  - *t_serv*: the service time for the $(n-1)^{th}$ packet for the n^{th} reference packet
  - *t_dep*: the departure time for the n^{th} reference packet. *t_dep* = *t_arv* + *t_wait*

- Algorithm

step 1. Calculate the waiting time for the reference packet (priority i) for the F_i.

```
while  (t_last + t_ia <= t_arv) {
    t_wait = t_wait + t_serv - t_ia ;
    if (t_wait < 0) {
        t_wait = 0;}
    t_last = t_last + t_ia;}
```

step 2. If there is any packets between *t_arv* and *t_dep* in any of the higher priority background traced file, then repeat step 1 until no other higher priority traced packet is between *t_arv* and *t_dep*.

step 3. If there is any higher packet(s) arrived between *t_arv* and *t_dep*, defer the *t_dep* by the service time of the higher priority packet(s) and recalculate *t_dep*

step 4. Repeat step 2 and 3 until there is no any other reference of background packet between *t_arv* and *t_dep*

4 Simulation Scenario and Network Element Model Description

4.1 Simulation Scenario

In this study, we consider voice and data service scenarios. To evaluate the application-level performance with different network configurations, we consider the following five scenarios:

– Scenario 1 (2.5G Tandem switch): A voice packet is initiated in MT and transferred to BTS, the BTS then sends the voice packet on the ATM/AAL2 to MSC. In the MSC, InterWorking Function (IWF) converts an EVRC packet to PCM 64 kbps packet format and sends it to tandem switch.

– Scenario 2 (3G ATM, G.711): From MT to MSC is the same as Scenario 1 but IWF in MSC converts an EVRC packet to a PCM packet format and sends it to a media gateway. The media gateway transfers PCM packet to ATM CN using ATM/AAL1
– Scenario 3 (3G ATM, G.726): From MT to media gateway is the same as Scenario 2. The media gateway converts a PCM 64 kbps packet to a G.726 ADPCM 32 kbps packet and sends it to ATM core network using AAL2 multiplexing.
– Scenario 4 (3G+ IP, VoIP): MSC converts an EVRC packet to a G.726 32 kbps packet and sends it to IP based media gateway using 100BT Ethernet. Then the IP media gateway sends it to IP CN.
– Scenario 5 (3G+ All IP, VoIP, vocoder bypass): This is All IP scenario. IP based BTS sends EVRC packet to IP RNC using 100BT Ethernet. The RNC then transfers the EVRC voice payload over an IP packet to IP CN.

4.2 Network Elements Model Description

Two firewalls, two load balancers and 3 routers are modeled in the data center. For CN elements, media gateway, ATM switch and IP router models are implemented. We also implemented MSC(for 2.5G and 3G), RNC(for 3G+), BTS and MT models for RAN elements. The packet processing time for each network element follows the 3GPP standard specification [11]. We fully implemented each protocol in the network elements shown in Fig. 2. The air channel and physical layer is modeled based on the average channel quality and mobility. The user mobility model assumed that mobile users are uniformly distributed in a cell. Mobile users ware assumed to move at a pedestrian speed of 3 km/h with worst

Table 1. Simulation Parameter

Category	**Parameter**	**Reference**
Voice traffic	EVRC 8 kbps	[7]
Web browsing traffic	Main object size: lognormal(10.8,250)kbyte Embedded object size: lognormal (7.8,126) kbyte Number of objects per page: Pareto shape :1.1, location: 55	[8]
TCP parameters	Windows 2000 based parameters	[8]
Radio Link Data Rate (kbps)	9.6, 153.6, 2000, 2400	[6]
RLP scheme	(2,3) RLP scheme	[2]
Processing time (m sec)	MT - forward : 36.55, reverse : 63.05 BTS - forward : 15, reverse : 9 MSC/RNC - forward : 7, reverse : 7 ATM/IP router: 0.1, Internet : 1. IP router processing time:100 μsec	[11]

case fading of single path Rayleigh. Based on the location of the mobile terminal, we use the results of the link-level simulations to estimate the power requirement for the user requested data rate. we have implemented a 3G 1X-RTT packet scheduler and a proportional fair scheduling algorithm for 3G 1X-EV scenario based on [12]. Some of the simulation parameters are summarized in Table 1.

5 Simulation Results and Discussions

5.1 Voice Quality Simulation Results

To compare the end-to-end voice packet delay performance of 2.5G to that of 3G+, we perform the simulation for the five different scenarios. The results are presented in Fig. 4. The background traffic load for each network element is 40% in the simulation. In scenario 2 and 3 (ATM CN), the voice packet delay is a little

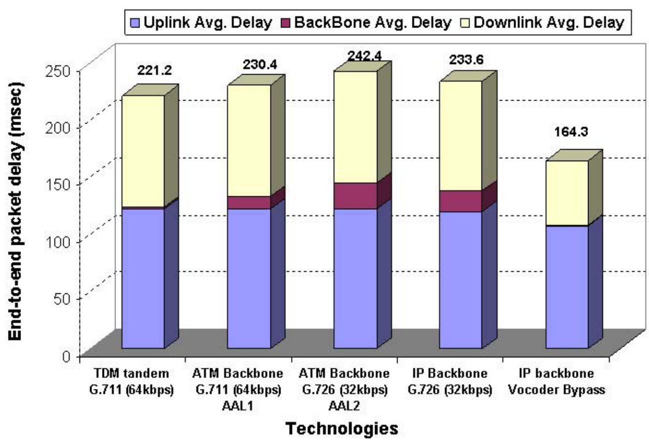


Fig. 4. end-to-end voice packet delays for technology evolution

Table 2. Voice quality scores(R value)

E2E One Way Delay(msec)	64kbps (G.711)	32kbps(G.726) 16kbps(G.728)
0	94	87
50	93	86
100	92	85
150	90	83
200	87	80
250	80	73
300	74	67

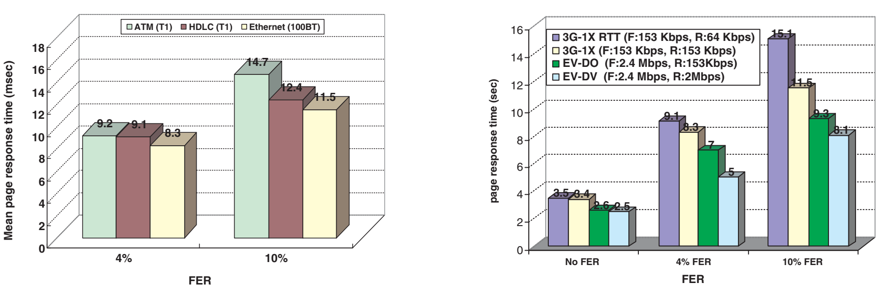
100-90 : Best Quality , 90-80 : High Quality
80-70 : Medium Quality, 70-60: Low Quality

bit larger than that of the scenario 1 (tandem switch). Because in both scenarios ATM processing delay is included in the CN, and the AAL2 multiplexing delay is (processing delay and Timer_CU : 2msec) also included in CN for the scenario 3. (IP CN, G.726 ADPCM), the CN packet delay increase more in scenario 4 than in the scenario 2 (ATM AAL1) since the IP packet overhead for EVRC voice traffic is larger than the packet format overhead of ATM. The scenario 5 is for the vocoder bypass which means that an EVRC voice packet is transferred over to the IP packet without any transcoding to other coding scheme. Vocoder bypass reduces RAN and CN coding processing delay by results in 30% compared to scenario 3. If we map these one-way end-to-end delay to the R value in Table 2 in [13], the voice quality for the scenario 1 and 2 provide the high quality voice. The scenario 3 and 4 provide the medium quality voice, while the vocoder bypass scenario meets the high quality voice.

5.2 Data Service Performance Simulation Results

ATM vs. IP in Radio Access Network. The ATM transport technology in current 3G network will eventually migrate to the IP technologies. Fig. 5(a) presents the web page response time for three different RAN transport technologies: ATM, HDLC over T1, and 100BT Ethernet. At 10% FER, we observe 15.6% and 21.7% page response time reduction when RAN transport technology migrates from ATM to HDLC and ATM to 100 BT Ethernet, respectively. The 21.7% performance improvement is due to the higher transmission speed and lower packet overhead in IP layer. In this case, IP transport technology is a better solution for the higher FER environment since the IP packet overhead is smaller than the ATM packet overhead for web browsing data traffic. However, it shows the opposite effect to small size voice packet as shown in Fig. 4.

3G-1X RTT vs. 3G-1X EV. 3G-1X EV service started from 2001 in Korea. 3G-1X EV enhances the data rate to 2.4 Mbps. To compare the data service per-



(a) different RAN transport technologies (b) 3G technology

Fig. 5. Web page response time for different RAN transport and 3G technology

formance for 3G wireless technology with 2.5G wireless technology, we measure the web browsing response time for different data rates in 3G-1X RTT and EV networks. We assume that 100 BT Ethernet RAN and IP CN transport technology are used in this scenario. There is no significant performance difference when the channel is error-free. However, at 10% FER, EV provides 46% reduced response time compared to 1X RTT. The higher data rate in RAN of EV results in faster frame retransmission compared to 1X RTT. As FER increase, the performance difference between the two technologies becomes more significant.

5.3 Simulation Runtime Performance

The run-time performance of the simulation can be defined in terms of number of events and processing time per event. The simulation run-time performance is always an important issue, but is especially so for the network simulations where the number of events can be extremely large. As mentioned previously, we have divided the traffic into the foreground and the background traffic and developed the specialized techniques for handling each to improve the simulation efficiency. To build the wireless performance simulator, we modelled FTP applications with varying numbers of users: The FTP application was a 1 Mbyte file download over the 64 kbps data rate. Table 3 shows some of the measured simulation run times. The simulation takes 120 sec for a single user file download and the simulation time increased linearly according to the number of concurrent FTP sessions. It clearly shows that the simulation performance is not feasible when the number of concurrent application sessions is large. However, the last two rows of Table 2 show that the simulation performance is improved when additional FTP sessions are modelled as background traffic. For this scenario, 124 Mbps and 147 Mbps traffic on the average, which is about 80% and 95% of STM-1, ware generated in all of the nodes along the reference connection (excluding the application server) and one foreground FTP session was created.

Table 3. Simulation Run Time with and without Background traffic model

Number of Foreground Users	Number of background Users	Download File size	Simulation Time (sec)
1	0	1 Mbytes	120 sec
2	0	1 Mbytes	237 sec
3	0	1 Mbytes	355 sec
4	0	1 Mbytes	478 sec
1	1400	1 Mbytes	190 sec
1	1670	1 Mbytes	205 sec

6 Conclusions

We described an end-to-end performance simulation model and methodology that we build for cdma2000 network. The simulator modelled all protocol layers from the physical through the application layer and modelled details of the packet handling characteristics of each network element along the path. We addressed application level performance issues in terms of wireless technologies evolution from 2.5G to 3G+. We found the end-to-end QoS mechanism should be provided in every network elements where the packet passes by. The main contributions of this paper are threefold:

1. Develop the new simulation methodology using the traced file and Lindley's recursion method to improve the simulation runtime performance
2. Build the end-to-end network simulation model for cdma2000
3. Access user-perceived application performance for the voice and the data services

The wireless performance simulator presented in this paper has been used to predict and quantify the performance of cdma2000 applications, services, and network architectures.

References

1. Z. Dziong, F. Khan, K. Medepalli, S. Nanda, "Wireless Internet Access Using IS-2000 Third Generation System: A Performance and Capacity Study," *Wireless Networks*, vol. 8, pp. 325–336, 2002
2. E. Chaponniere, S. Kandukuri and W. Hamdy, "Effect of physical layer bandwidth variation on TCP performance in CDMA2000," *IEEE Vehicular Technology Conference spring 2003*, Jeju, Korea.
3. F. Li, M. Nguyen and W. Seah , "QoS Support in IP/MPLE-based Radio Access Networks," *IEEE Vehicular Technology Conference spring 2003*, Jeju, Korea
4. O. Sallent, J. Romero, R. Agusti and F. Casadevall, "Provisioning Multimedia Wireless Networks for Better QoS: RRM Strategies for 3G W-CDMA," *IEEE Commun. Mag.* pp. 100-106, Feb, 2003.
5. J. H. Kim and C. W. Lee, "End-to-end User Perceived Application Performance in 3G+ Networks,"*in Proc. IEEE ICC'04*, Vol. 4, Paris, France, Jun., 20-24, 2004, pp. 2337 - 2341.
6. 3GPP2 S.R 0023 V.20, "High-Speed Data Enhancements for cdma2000 1x Data Only" 2000-11
7. 3GPP2 C.S0014-0 V.1.0, "Enhanced Variable Rate Codec," 1999-12
8. 3GPP2 TSG-C.R1002, "1xEV-DV Evaluation Methodology (V13)," 2003
9. A. Reyes-Lecuona, et al. "A page-oriented WWW traffic model for wireless system simulations,"Proceedings of ITC-16, pp 1271-1280, 1999
10. M. Lucas, B. Dempsey, D. Wrege and A. Weaver," An Efficient Self-Similar Traffic Model for Wide-Area Network Simulation," IEEE GLOBECOM '97, Phoenix, AZ, November 1997
11. 3GPP TR 25.853 v4.0.0, "Delay Budget within the Access Stratum," 2001-03
12. A.Jalali, R.Padovani and R.Pankaj, "Data throughput of CDMA-HDR a high efficiency-high data rate personal communication wireless system," *IEEE Vehicular Technology Conference*, Tokyo, May 2000
13. M. Perkins, C. Dvorak, B. Lerich, J. Zebarth, "Speech Transmission Performance Planning in Hybrid IP/SCN Networks," *IEEE Commun. Mag.* pp.126-131, July 1999.

Client-Controlled QoS Management in Networked Virtual Environments

Patrick Monsieurs, Maarten Wijnants, and Wim Lamotte

Expertise Centre for Digital Media,
Limburgs Universitair Centrum,
Universitaire Campus, B-3590 Diepenbeek, Belgium
{patrick.monsieurs, maarten.wijnants, wim.lamotte}@luc.ac.be

Abstract. In this paper, we propose an architecture to regulate the bandwidth usage of multimedia streams in networked virtual environments. In this architecture, intelligent proxies are placed in the network. These proxies can transcode incoming streams to lower quality versions of those streams, thereby decreasing network traffic. A network intelligence layer at the receiver controls these transcoders based on the bandwidth the streams consume, and the importance that the receiving application assigns to each stream. To access this latter information, the network intelligence layer provides an interface between the QoS management of the network and the application's interest manager. This interest manager assigns a relative importance to each individual network stream. As a result, the network intelligence layer separates application logic from network QoS management, thereby maximizing its reusability. These concepts were implemented in an existing networked virtual environment framework, and experiments were performed to validate the ideas. The experiments demonstrate that bandwidth allocation can be changed dynamically, based on user interest, thereby maximizing network throughput and quality of experience.

Keywords: QoS, Network intelligence, Multimedia.

1 Introduction and Related Work

Transmission of multimedia content over the Internet tends to consume large amounts of bandwidth. Therefore, applications where a large number of users simultaneously send video and audio need some way to reduce the amount of transmitted data, while still maintaining an acceptable quality. This paper focuses on techniques to reduce the downstream bandwidth usage in the final segments of the network connection. By incorporating application knowledge about the importance of individual streams, we attempt to maximize the user's quality of experience.

Several techniques exist to reduce the downstream bandwidth requirements of networked multimedia applications. A first technique is to subdivide the environment in several regions, associated with multicast groups. Receivers join only the multicast groups of those regions they are interested in, and as a result will only receive the streams sent by users located in these regions [1]. Receivers with less available bandwidth can then subscribe to fewer regions to limit the amount of data they need

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 268–276, 2005.

to receive. This allows receivers to regulate the total amount of received data. However, when using this technique, the bandwidth can only be adjusted by joining or leaving entire regions. Sometimes it is desirable to have more fine-grained control over the amount of bandwidth consumed.

Another technique consists of reducing the quality of a network stream by its sender, based on the receiver's interest in that stream [3]. While this reduces the network load of congested links, receivers with uncongested links needlessly receive lower quality streams. Also, this approach does not tend to scale well with large numbers of receivers.

The highest level of control over the bandwidth consumption is achieved when every individual stream can be either accepted, rejected or transcoded inside the network before it reaches the client. We propose an architecture where intelligent proxies are placed in the network nearby the clients. Clients notify the proxies of the relative importance of each network stream, from the client's point of view. The proxies use knowledge of the maximum available bandwidth, the bandwidth of each individual stream, the relative importance values of each stream, and the capabilities of its transcoders to decide to accept, reject or transcode those stream.

The importance of each individual stream is determined by the application's interest manager. To maintain application-independence of the network intelligence layer, a simple programming interface is provided that is linked to the interest manager of the application. Some examples of user interest detection already exist in the literature. In [3], more importance is assigned to video conferencing participants whose window is currently selected. In [4], video playback and decoding is suspended in obscured windows to save CPU resources.

Content-based transcoding of data streams by proxies in the network is not a new idea. In [5] and [6], images of HTTP streams are transcoded to different resolutions based on the capabilities of the receivers. For example, images can be rescaled or cropped when sent to devices with small displays. In [7], transcoding-enabled caching proxies are placed in the network. These proxies transcode video streams, based on the capabilities of both the client and the network. The highest quality video stream is cached at the proxy, and is transcoded for clients that require a lower quality version.

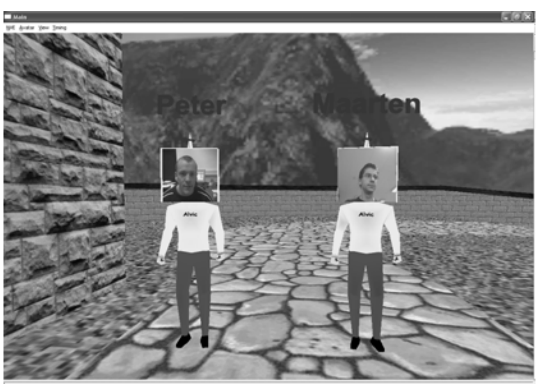


Fig. 1. Screenshot of video avatars in our networked virtual environment

Distribution of a network's link available bandwidth is discussed in [8]. All network traffic arriving at a router is divided into a number of classes, which are placed in a hierarchy. Non-leaf nodes in this hierarchy specify a distribution of minimum available bandwidth among its child nodes. Each class in the hierarchy is allocated the specified minimum bandwidth. When classes do not consume all available bandwidth, the remaining bandwidth is distributed among the other classes in the hierarchy. The difference between this approach and our technique is that this approach manages bandwidth by assigning individual network packets to appropriate queues. Our approach does not consider individual network packets, but deals with streams as a whole using the average bandwidth consumed by those streams.

2 Example NVE Application

To illustrate and test the concepts of this paper, we used our in-house developed networked virtual environment where users are represented by video avatars. A detailed description of this environment is given in [2][9][10], and a screenshot is shown in Fig. 1. In this system, web cams capture the faces of the users. These video streams are subsequently encoded in three different frame rates and/or bit rates.

The virtual world is subdivided in a number of regions. Each region has a multicast address associated with it for each video quality. Each client is interested in receiving video and positional data only from a number of regions in its vicinity. Based on the available bandwidth and the distance to those regions, an appropriate video stream quality for that region is selected, and the associated multicast address is joined. While receiving clients move around in the environment, multicast groups are dynamically joined and left.

3 Network Intelligence Architecture

When a network link becomes saturated, throughput decreases and delays become higher. As a result, in order to maintain the quality of experience in networked applications it is necessary to avoid requesting more bandwidth than is available. This can be achieved by reducing the quality and bandwidth usage of certain streams, or by completely blocking them, before they reach the client. To maintain a high quality of experience, it is essential that streams that are important to the user be allocated more resources than less important streams.

Determining the importance of individual streams is an application-specific issue. On the other hand, calculating and distributing available bandwidth is an application-independent networking task. To keep a strict separation between application logic and network management operations, a network intelligence layer is introduced between the transport and application layers, as shown in Fig. 2. This layer communicates with an intelligent proxy located inside the network. The proxy is able to transcode or block incoming network streams. The network intelligence layer queries the interest manager of the application to determine the relative importance of each stream. The importance of each stream is then communicated to the proxy. If a threshold of the available network capacity at the proxy is exceeded, the proxy

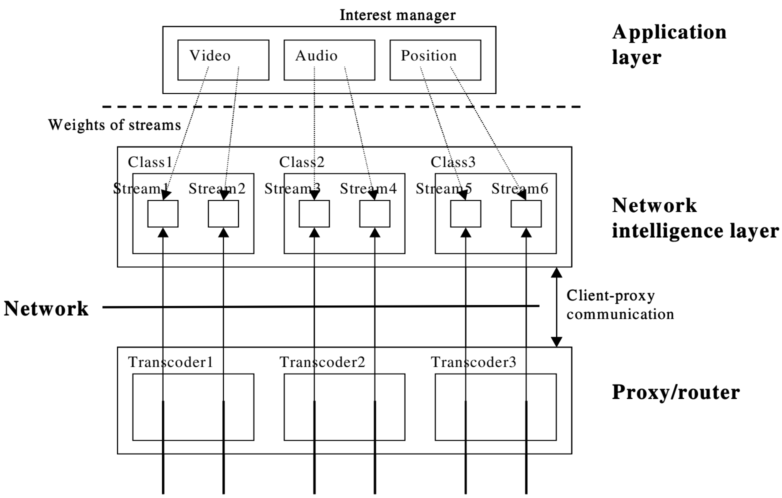


Fig. 2. Overview of the proposed network intelligence architecture. Incoming network traffic can be limited by an intelligent proxy located inside the network, by blocking or transcoding network streams

transcodes or blocks the least important streams until the desired bandwidth usage is reached. The individual components of the architecture are discussed in the following sections.

3.1 Intelligent Proxy

In this section, we will present a short description of the intelligent proxy. A more detailed description of the proxy is given in a different paper, which is submitted to another conference [11].

The intelligent proxy, located in the network near the client, contains a number of content-based transcoders. These are able to transcode incoming network streams into less detailed streams that consume less bandwidth. The concept of a transcoder in this context is very general. It can range from a simple filter that blocks specific streams, to a process that decodes, re-encodes and retransmits an entire video stream. Because the transcoding of video streams is very processor intensive, it does not tend to scale well with large numbers of streams. It is therefore better for clients to transmit a layered version of the stream using a codec such as MPEG-4 FGS [12]. This way, transcoding a stream can be limited to filtering those packets that contain the desired quality and discarding the other packets.

The proxy measures the consumed bandwidth of the network streams used by all connected clients. When the downstream capacity of the client application is exceeded, the proxy transcodes specific streams to a lower quality until the available bandwidth is no longer exceeded. As a result, the available downstream capacity can be fully utilized. The algorithm used to distribute the available bandwidth over the different streams is described in section 3.3. The communication between client and proxy is described in section 3.4.

By placing the intelligent proxy near the receivers, the complexity of managing network traffic is located at the edge of the network. Consequently, the proxy will have to handle only a limited number of streams. We propose that these proxies are placed at DSLAM-level of xDSL networks. This approach manages the bandwidth of the final links between senders and receivers, which is often the most problematic part of the connection. Furthermore, the intelligent proxy does not have to be located at the nearest router to the receiver, which has the advantage that not every router in the network must support the architecture.

3.2 Network Intelligence Layer

The network intelligence layer provides the application with a set of networking operations that replace the standard Winsock or Unix networking operations. Upon reception of a network stream, the application provides information about the content of the network stream to the network intelligence layer. This allows the network intelligence layer to associate the appropriate transcoder with this network stream at the proxy.

This layer also provides a generic interface to the interest manager of the client application. Using this interface, the network intelligence layer can determine the relative importance of every network stream used by the application. The importance of each stream is sent to the intelligent proxy, which will use this information to reduce or block less important streams whenever the available bandwidth is exceeded.

3.3 Distribution of Available Bandwidth

Similar to the hierarchies used by hierarchical link sharing [8], we build a hierarchy of streams and components to manage the available bandwidth by the proxy. Individual streams are represented as leaves in this hierarchy, while other nodes implement a bandwidth distribution technique. We use the following distribution techniques:

- *Priorities*: The child node with the highest priority is assigned all requested bandwidth. Any remaining bandwidth is distributed to the second-highest child, etcetera.
- *Mutexes*: All bandwidth is allocated to a single child of the collection.
- *Weighted collection*: Each child of such a node is assigned a relative weight, and bandwidth is distributed among children based on their weight and their bandwidth consumption. The algorithm used is described below.

In a weighted collection node, every stream s_i is assigned a weight w_i between 0 and 1. Every unregulated stream s_i consumes m_i bandwidth. The maximum desired bandwidth M of all streams is $\sum_i m_i$. If M exceeds the maximum available bandwidth B, the bandwidth of each stream must be modified using the following algorithm:

Assume that every stream s_i can be transcoded to a set of bandwidths S_i. This set can contain a number of discrete values, or the continuous range between 0 and m_i. $h(i, b)$ is defined as the highest bandwidth of a stream s_i that is less than bandwidth b, and is the highest value of S_i that is less than or equal to b. The weighted sum W of all streams is $\sum_i w_i m_i$.

An initial estimate of the target bandwidth for each stream is $t(i) = h(i, w_i m_i B/W)$. In an ideal case where $\forall i$: $h(i, b) = b$, $\Sigma_i t(i) = B$ and the available bandwidth would be used optimally. It is more likely, however, that every stream consumes less bandwidth than they are allocated, resulting in a remaining bandwidth $R_0 = B - \Sigma_i t(i)$. The remaining bandwidth R is then allocated to all streams, starting with the stream with the highest weight. Assuming the streams are sorted by their weight, where w_0 is the highest weight, the final target bandwidth for each stream equals $T(i) = h(i, t(i)+R_i)$, and $R_i = R_{i-1} - (T(i - 1) - t(i - 1))$ for $i > 0$.

3.4 Communication Between Proxy and Client

The following tasks are handled by the communication between proxy and client:

- *Admission control*: The client's network intelligence layer can notify the proxy that streams destined for a specific port must be blocked by default. When a new incoming stream is received at this port by the proxy, the client is notified of this new stream.
- *Creation of a stream hierarchy*: The application uses the network intelligence layer to build the hierarchy of streams used to distribute bandwidth. When the client receives a new stream, a corresponding node is created on the proxy. The client also specifies the content type of this stream so the proxy can use the correct transcoder to modify that stream.
- *Communicating importance of streams*: Clients transmit weight values ranging from 0 to 1 that indicate the importance of individual streams to the proxy. These values are transmitted at regular time intervals. In our application, this weight value is determined by the distance to the sender's avatar.
- *Multicast tunneling*: The client's network intelligence layer intercepts the client's requests to join or leave multicast groups and transmits these to the proxy. The proxy subscribes to or unsubscribes from these groups, and unicasts the multicast traffic to and from the client.

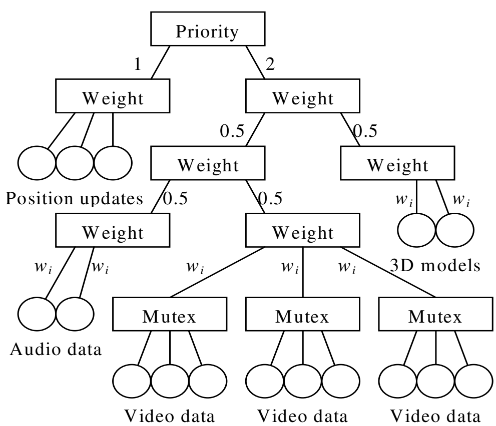


Fig. 3. Hierarchy of network streams of the client application

4 Experimental Results

In our experiments, four clients running the unmodified version of the NVE and an intelligent proxy are connected through a local area network. Behind the intelligent proxy is a different network, containing a client running a modified version of the NVE using the network intelligence layer.

The different streams of the client application are organized in the hierarchy shown in Fig. 3. Circles represent network streams, and rectangles represent distribution primitives as discussed in section 0. Position update streams are given the highest priority, because this information is needed by the interest manager to assign weights to other streams. The remaining bandwidth is allocated to the distribution of 3D models in the environment, and to audio/video streams. Because each client transmits three versions of the same video stream, these are grouped together using a mutex primitive.

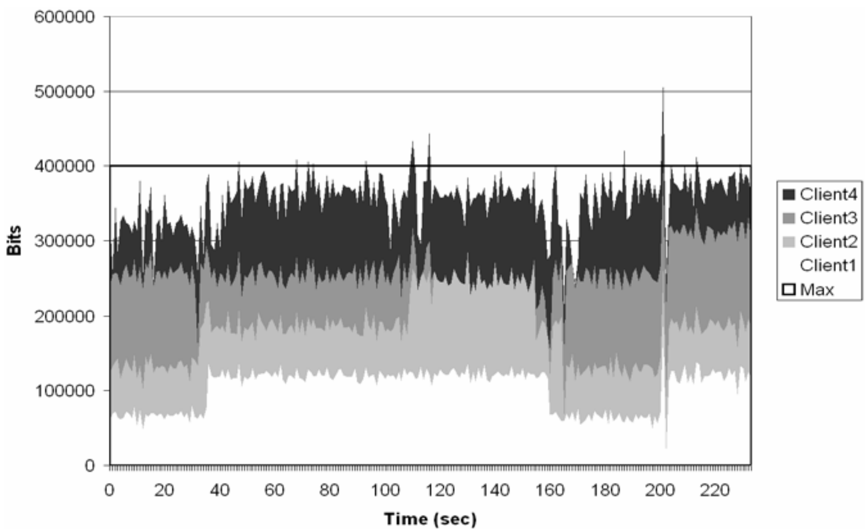


Fig. 4. Experiment 2: changing the relative importance of streams

The weight primitives assign weights according to the relative importance of the components. Therefore, even though all combined audio and video streams are both given a weight of 0.5, this does not mean they are each allocated the same amount of bandwidth. The distribution also depends on the amount of consumed bandwidth, and as audio streams typically consume less bandwidth than video streams, the audio streams will consume less bandwidth overall.

In the experiments that we performed, we only consider the video streams of the clients because these comprise the major part of the network traffic. In the traffic measurement graphs, different video streams are represented by different shades of grey. All three possible qualities of video stream are represented by the same shade of grey, but the amount of bandwidth they occupy varies. This can be observed by the height of the received traffic in the graphs.

In a first experiment, shown in Fig. 4, all clients remain stationary, but the maximum available bandwidth was gradually decreased. Initially, sufficient bandwidth is available to receive all video streams at the highest quality. The avatar of client 3 is nearest to the receiver, and its stream therefore has the highest weight. The avatars of clients 2 and 4 are farthest away, and therefore the quality of their video streams is lowered first by the proxy as available bandwidth is decreased. This is apparent from the lower bandwidth usage of their network stream in the graph.

In the second experiment, shown in Fig. 5, the available bandwidth was kept constant, but not high enough to receive all the streams at maximum quality. After 30 seconds, the avatar of client 3 slowly moves away from the receiver's avatar. As a result, the relative weight of this stream decreases and more bandwidth is allocated to the other streams. After 150 seconds, the avatar of client 3 rapidly moves back towards the receiver's avatar. The dark grey stream is now assigned the highest weight, decreasing the quality of the other video streams. This demonstrates that the bandwidth allocation of the application is dynamically adjusted based on the application's needs.

5 Conclusions

In this paper, we have presented a network intelligence layer that combines application logic and quality of service of networked applications. Downstream bandwidth usage is regulated by intelligent proxies inside the network, based on the relative importance of each network stream. These proxies manage the bandwidth by blocking or transcoding incoming streams. The network intelligence layer communicates with the application's interest manager to indicate the relative importance of each individual network stream. These weights are subsequently used by the proxy to allocate bandwidth to each stream, thereby maximizing the quality of experience for the application user.

The concepts presented here were integrated and tested in an existing NVE framework. The results show that it is possible to manage individual network streams dynamically based on the application's interests. Also, incorporating the network intelligence layer in the application proved to be reasonably straightforward.

Acknowledgement

Part of this research was funded by the IWT project number 020659, Alcatel Bell and the Flemish Government. Part of Maarten Wijnants' work was carried out at ANDROME NV, before he moved to the Limburgs Universitair Centrum. We also wish to thank ANDROME for the use of their video codec software.

References

[1] Barrus, J., Waters, R., Anderson, D.: Locales and Beacons: Efficient and Precise Support for Large Multi-User Virtual Environments. In: Proceedings of VRAIS'96, IEEE Computer Society (1996) pp. 204–213

[2] Quax, P., Jehaes, T., Jorissen, P., Lamotte, W.: A Multi-user Framework Supporting Video-based Avatars. In proceedings of the 2nd workshop on Network and system support for games. Redwood City, CA, USA. 2003. ACM Press, pp. 137 - 147

[3] Scholl, J., Elf, S., Parnes, P.: User-interest Driven Video Adaptation for Collaborative Workspace Applications. LNCS 2816, 5th COST 264 International Workshop on Networked Group Communications, NGC, pp. 3-12, 2003

[4] Katchabaw, M. J., Lutfiyya, H. L., Bauer, M. A.: Using User Hints to Guide Resource Management for Quality of Service. The 1999 International Conference on Parallel and Distributed Processing Techniques and Applications, July 1999

[5] Smith, J. R., Mohan, R., Li, C.-S.: Content-based Transcoding of Images in the Internet. In proceedings of IEEE International Conference of Image Processing, Oct '98 (ICIP-98)

[6] Smith, J. R., Mohan, R., Li, C.-S.: Transcoding Internet Content for Heterogeneous Client Devices. In proceedings of IEEE International Conference on Circuits and Systems, May '98 (ISCAS)

[7] Shen, B., Lee, S.-J.: Transcoding-enabled caching proxy for video delivery in heterogeneous network environment. In proceedings of IASTED, Internet and Multimedia Systems and Applications Conference, Aug. 2002

[8] Floyd, S., Jacobson, V.: Link-sharing and Resource Management Models for Packet Networks. In IEEE/ACM Transactions on Networking, Vol. 3 No. 4, August 1995

[9] Quax, P., Monsieurs, P., Lamotte, W.: Performance Evaluation of Client-Based Scalable Video Stream Selection using Autonomous Avatars. In proceedings of the ACM SIGCHI International Conference on Advances in Computer Entertainment Technology (ACE 2004), June 2004

[10] Quax, P., Flerackers, C., Jehaes, T., Lamotte, W.: Scalable Transmission of Avatar Video Streams in Virtual Environments. Proceedings of the 2004 IEEE International Conference on Multimedia and Expo (ICME 2004). Tapei, Taiwan ROC. 2004. IEEE

[11] Wijnants, M., Monsieurs, P., Lamotte, W.: Improving the Client Quality of Service by Incorporating Intelligent Proxies in the Network. Submitted to Networking 2005

[12] Li, W.: Overview of fine granularity scalability in MPEG4 video standard. IEEE Trans. Circuits and Systems for Video Technology, vol. 11, no. 3, pp. 301-317, Mar. 2001

UML-Based Approach for Network QoS Specification

Cédric Teyssié and Zoubir Mammeri

IRIT Laboratory, Paul Sabatier University, Toulouse, France
{cedric.teyssie, mammeri}@irit.fr

Abstract. New applications require Quality of Service from networks. Managing QoS increases even more the complexity of networks. Network development techniques must apprehend this complexity from a functional point of view but also from QoS point of view. The object paradigm and UML in particular can help reducing the network design complexity. In this paper, we propose a language formally defined and compliant with the object paradigm, intended to specify QoS in networked environments.

1 Introduction

To respect Quality of service (QoS) constraints, all the communication actors must understand in a coherent way the required QoS. All actors (customer, internet service provider …) must have the same definition of QoS, or at least they may not have divergent QoS definitions. In order to have high quality and efficient networks, the QoS notion must be integrated early in the network development process. Thus, the need for a description language of QoS is primordial.

Object paradigm has numerous capabilities for apprehending the complexity of large systems. UML [1], particular, is a widely used modeling language in development processes. However, UML is not well adapted for the specification of non-functional aspects such as QoS. Our contribution is to add to UML the ability to specify and handle the QoS that will be used in the networks.

The rest of the paper is organized as follows. Section 2 deals with related work. Section 3 presents our contribution to specify network QoS in UML. Modeling of QoS constraints in UML is presented in section 4. The integration of our extensions in UML models is dealt in section 5. Section 6 gives a consistent example of our approach use and the last section concludes the paper.

2 Related Work

The integration of QoS notion in development process is a very active research field. Several contributions have been made and several ways are still investigated but only few works have specifically focused on networks. We can classify the existing approaches into three categories: QoS in middleware, QoS integration in UML by objects, and QoS integration in UML by QoS dedicated languages.

The first way consists in the integration of QoS concept in system middleware such as Common Object Broker Request Architecture (CORBA). QML (QoS

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 277–285, 2005.

Modeling Language) [2] integrates non-functional elements (in textual form) in CORBA. While UML is used to model the system structure, Interface Definition Language (IDL) is used to specify the functional elements of the system interfaces and QML is used to describe the non-functional ones. But, QML lacks component notion and is too dependant on CORBA to be used efficiently. Component QML (CQML) [3] is based on the Open Data Processing (ODP) reference model and extends QML by adding component notion. QDL (QoS Description Language) is a part of the Quality Object solution (QuO) [4]. QDL notation is inspired from C++ and extends the IDL. QDL is composed of three dedicated languages: Contract Description Language (CDL), Structure Description Language (SDL), and Resource Description Language (RDL). CDL is concerned with the QoS specification and is QoS contract oriented. This orientation makes the integration to UML difficult and its dependence on CORBA environment makes this integration harder.

For UML to support QoS, OMG issued the RFP (Request For Proposal) [5]: Schedulability, Performance and Time Profile (named SPT Profile). It introduces a standard way of integrating non-functional elements in UML. Although this profile allows modeling resources or schedulability elements, its QoS definition is too imprecise to be used. A new RFP: The UML Profile for Modeling Quality of Service and Fault Tolerance Characteristics and Mechanisms (named QoS profile) [6] specifies the QoS notion. However, the QoS can not be verified using this RFP.

Most of the proposed QoS languages are not intended to work with UML. As an example we denote [11] (based on the temporal logic) and HQML [13]. Most of these languages are too far from the natural language and thus are difficult to handle inside UML models. Our QoS language must insure that QoS is captured and specified in an independent way to guarantee interoperability and maintain backward compatibility with existing tools and techniques.

3 QoS Definition Language

To allow QoS validation from UML models, we develop a formal QoS language that allows specification of QoS elements, QoS units and QoS structure. For space reasons, all the language rules (in BNF form) and semantics that are not presented in this paper may be found in [12]. As, we focus essentially on network QoS, we also present rules to specify Service Level Agreements.

QoS Model. The QoS used in the network and by the network customers is to be defined in the very early step of the design process. This ensures that, during all the development steps, every QoS element will be uniquely defined and universally known. It avoids QoS inconsistency issues. We propose to represent the QoS by a QoS UML model to keep UML strengths for reducing complexity. This model will be derived in our QoS language to check its consistency prior to the QoS elements to be used in the development process. The elements of our QoS definition language must support inheritance and composition to be compliant with UML approach and to allow apprehending efficiently network complexity. As some operations may be applied for several QoS notions, we informally define the QoS specification element to group these notions. A QoS specification element groups the QoS, QoS characteristic and QoS aspect notions.

QoS class. The basic element constituting of our lan130guage is the QoS class that represents the global QoS of a component. For example, it could be the QoS assumed by a network. We represent the QoS as a vector of QoS characteristics as proposed in [7]. A QoS can merge several QoS characteristics such as time or fault tolerance. The definition begins with the QoS keyword, followed by its label (QoS_label). A QoS can be built by two means:

- A QoS (with the "`inherits from`" keyword) may inherit all QoS characteristics of its QoS super class. It is still possible to add new QoS characteristics.
- A QoS can be specified by defining all its QoS characteristics (`QoS_Chars`).

```
<QoS> ::= QoS <QoS_label> (inherits from <QoS_label> [<qos_chars>])|
<qos_chars>
```

QoS Characteristic. A QoS characteristic represents a single part of a QoS. For example the network knows about its capacity (such as throughput or jitter aspects) while the network users know about relative quality (good, poor…). Each of these two elements may be specified using QoS characteristics. They can also be used to decompose different network QoS, like security, safety or time considerations that will be taken into account by separate services or components. We represent the QoS characteristics as a set of QoS aspects. The QoS_chars describe the QoS characteristics set of a QoS. Each QoS characteristic is uniquely identified by a label (QoS_char_label) and can be built according to two means:

- By specification of all of its QoS aspects. A QoS domain (`QoS_domain`) that expresses the application domain of the characteristic (for example, it may be time domain, security domain…) is linked to each QoS characteristics.
- By inheritance from an existing QoS characteristic. When a QoS characteristic inherits from another, the QoS domain is derived from the inherited QoS characteristic and does not need to be specified. New QoS aspects may be added.

```
<QoS_characteristic> ::= (<QoS_domain_label> : <QoS_char_label>
[<qos_aspects>]) |(new QoS_Characteristic <QoS_domain_label> :
<QoS_char_label> <qos_aspects>)
```

QoS Aspect. A QoS aspect is a particular element of a QoS characteristic. For example, for a QoS of a network that contains a time QoS characteristic, QoS aspect may be the jitter. The throughput QoS characteristic may contain QoS aspects such as maximum and minimal throughput. To deal with network components, QoS may be composed or compared. Our QoS language must then ensure interoperability between QoS aspects. QoS aspect type allows such operations. It classifies QoS aspects in a precise interest field and allows compliance between QoS aspects of same QoS domain. Each QoS aspect (QoS_aspect) can be specified or reused from existing QoS domains. For a new QoS aspect, the label (QoS_aspect_label) and the type (QoS_aspect_type) of this QoS aspect must be specified. The QoS aspect type specifies the nature of the aspect. The QoS aspect type may constrain the QoS units or the QoS_value_types to be used. For example, if the QoS aspect is time related, an aspect type may be jitter or duration. The QoS aspect type may be constrained by the QoS domain of the enclosing QoS characteristic. Two additional elements complete the QoS aspect definition: QoS_value_type and QoS_unit. The QoS value type specifies the means to represent the QoS aspect values. For example, duration may be represented as real values or a set of defined values. The QoS unit element represents the unit that represents the QoS aspect. For time aspects, the unit may be s (for second), ms… More complex units may also be represented. Properties (QoS_aspect_properties) may be added to a QoS aspect in order to specify the aspect behavior. Two types of properties may be added:

– Combination related properties are concerned with the manner to combine compatible (with same type) QoS aspects. We reuse the three values defined in [8]: Additive, Multiplicative and Concave.

– Comparison related properties specify the manner to compare compatible QoS aspects. As in [3], two values may be used: increasing and decreasing. An increasing value implies that higher is the aspect value, better is the QoS. A decreasing value implies that lower is the aspect value, better is the QoS.

```
<QoS_aspect> ::= (<QoS_aspect_type> <QoS_aspect_label>
        [is <QoS_value_type> in <QoS_unit>[<QoS_aspect_properties>]])
   | (new QoS_aspect <QoS_aspect_type > : <QoS_aspect_label>
         is <QoS_value_type> in <QoS_unit>[<QoS_aspect_properties>])
```

Service Level Agreement (SLA) support. As our approach is network-oriented we integrate in our QoS language the SLA concept. SLA is a very important concept in a world where inter-networks connectivity has to guarantee QoS. SLA is a means for networks to express their guaranties or specify their needs (like QoS properties, usage restriction, service validity conditions). Five elements compose a SLA:

```
<SLA> ::= SLA <SLA_name>{identified by <identification>[<validity>]
       SLS:(<QoS_name> | {<QoS>})[<TCA>] [Complements : <complement>
       [; <complement>]*] }
<TCA> ::= TCA { <Restriction> [; <restriction>]* }
<Restriction> ::= constraint{(<QoS_constraint_name>|<QoS_constraint>)
       [Priority][behavior : (<QoS_name>)+]* }
```

An SLA must be uniquely defined to be referenced in a networked environment. So the `Identification` artifact may represent a DiffServ Codepoint (DSCP) for DiffServ environments [9], a merge of several IP fields. Service Level Specification (`SLS`) item is used to characterize the QoS that must be enforced between the SLA peers and therefore is represented by a QoS. The Traffic conditioning agreement (`TCA`) expresses rules that must be enforced by the SLA peers such as bit error rate probability threshold. The TCA element is composed by the following elements:

A Restriction is a QoS constraint merged with a priority and associated to a QoS. The priority item is used to express hierarchy between simultaneously raised restrictions. It may be represented by several data types as integer for example. If the restriction is raised, the action to do depends on the criticality (see §4.-Criticality). For soft constraints, the QoS associated with the restriction overrides the SLS (behavior role). For Hard constraint, no QoS can be used. If such a restriction is raised, it denotes an unrecoverable error. The validity condition element is used to disable the SLA for particular reasons (administrative reasons like service availability, test reasons …). The complement element allows extensions to the SLA definition for future or customized use and allows keeping compatibility with existing SLA.

4 QoS Handling Language

QoS element instantiation. To be used in the models, all QoS specification elements must be instantiated from the QoS defined in the QoS model. In addition, the QoS element must be integrated in the context of its functional counter-part in the structure class diagram model. For example, the QoS of a network must be linked to the Network class in the UML model. QoS element instantiation can be done using the next syntax:

```
<QoS_instan>::= Use <QoS_elt_type><QoS_elt_instan_name> as <QoS_name>
```

Value Assignment for a QoS specification element. As value assignment operation is nearly the same for QoS, QoS characteristics and QoS aspects, we detail only the operation for QoS. Three means are defined to assign a value to a QoS: By direct value assignment of a QoS from the values of another QoS. The two QoS concerned must have the same characteristics. By assigning one or all the characteristics composing the QoS (QoS_built_assign); By assignment resulting from a combination between existing QoS (see below).

```
<qos_assign> ::= <QoS_name> '=' <qos_name> | <qos_built_assign>
             | <QoS_name> cmb <QoS_name> [cmb <QoS_name>]*
```

QoS Combination. The combination operation between QoS specification elements deals with associating two QoS specification elements by combining their values. Applied to compliant QoS, this operation combines all composing QoS Characteristics. Applied to QoS characteristics, the combination operation combines all QoS aspects of the characteristics. Applied to compliant QoS aspects, the combination operation combines the QoS aspect values according to the combination property defined in the aspect structure. Compliant QoS aspect implies that the aspects to combine have the same QoS aspect type or compliant inherited ones.

QoS constraint representation. The most important aspect in specifying QoS is the expression of system QoS needs. We base our QoS constraint notion on its definition in the QoS profile. For space reason, complete definition of validity and priority is given in [12]. QoS constraint concept is defined by the following rule:

```
<QoS constraint>::=<QoS_constraint_type>:{[validity : <validity> ;]
criticality : <criticality>;[priority :<priority> ;]<QoS_cons_expr> }
```

As in [6] and [7], a QoS constraint can take several forms. We denote three main forms: the QoS required, the QoS admitted and the QoS offered. Each of this QoS constraint type inherits from the QoS constraint class. A QoS constraint is required if the constraint demands a quality guarantee to a service provider. It is the case for the QoS that a component requires from another component is represented by a required QoS constraint. An Admitted QoS constraint denotes the QoS that a component can accept. A component may express the QoS it can accept from other components. This QoS type is useful for passive components that can only accept a certain QoS. A QoS is offered if it represents the quality a service provides to its client(s). We believe that the QoS constraint definition given [6] characterization is too imprecise for the QoS to be validated or checked. Thus, we extend this QoS constraint notion. We define a QoS constraint expression as a choice of three types: single valued expressions, list of constraining values and complex expressions.

```
<Single value>::=<qos_aspect_name> <comparison operator> <QoS value>
<QoS value> ::= <value> <QoS unit>
<comparison operator> ::= <  |  =  |  >  |  ≥  |  ≤  |  ≠
```

Single values are not always sufficient to constrain values. For this purpose, it is possible to specify value list. In this case, the comparison element is replaced by the set operators `included in` and `not included in` followed by the list of the values. This operation is built in the same way as for single value QoS constraints. Complex QoS constraint expressions are nested QoS constraints (complex or not) that may integrate logical, comparison, arithmetic or statistical expressions.

Criticality. Unfulfilled constraints may lead to consequences with different degrees of severity. It implies our language to support criticality notion (Hard or Soft) to allow different behaviors. We define Hard and Soft constraints as used in real-time environments. Hard of soft constraints may complete soft constraints to specify differentiated restriction on the same QoS aspects. It may be used for specifying constraints on a characteristic which may be exceeded (soft constraint) until a precise threshold (hard constraint).

5 Integration to UML Models

While UML is dedicated to capture and model network elements (services, components…), our QoS language is dedicated to non-functional elements. The next step is to link these two approaches in a coherent way. We prefer UML using light extensions mechanisms because it does not require modifying UML metamodel. Therefore, it preserves compatibility with existing tools. In [10] we proposed a network QoS oriented methodology using UML. This methodology is organized as a slightly modified V development cycle. Three abstractions levels are created: User, Provider and Designer level. Each one models one view of the system. QoS agents are defined to capture systems elements at each level and structure them suitably for QoS specification. However, our previous work does not provide any QoS language to represent QoS. A methodology was given to design networks and their QoS, but it was not possible to specify formally this QoS. In this present work, our contribution was to define a formal QoS specification language and to integrate this QoS language in the methodology. We integrate the QoS of the network elements into UML artifacts context. QoS specification elements are linked together by the UML model. In networks, several elements have to be combined for the network to provide multiple services and to guarantee a precise QoS level for them. We define restrictions on UML operations (composition, inheritance, association, and SLA definition) to ensure QoS coherence.

When a component inherits from another component, for example a DiffServ router component inheriting from a general router component, to insure QoS coherence, the QoS subclass must also be inherited from the QoS super class. This can be done using the inheritance facilities of our language.

When several router components are composed together to give the network component, the composition operation can cause QoS issues. The issue resides in the means of composing all the router QoS to give the network QoS. From the QoS point of view, a composition can not be seen as a QoS characteristics merge. If the QoS to be composed do not have characteristics in common, we assume that merging the characteristics is sufficient. Thus, we can see the composition operation as a "work in common" of the components each one guaranteeing a particular QoS characteristic (or aspect). When QoS have characteristics in common, we must combine the QoS according to their QoS aspects combination property.

When associating two components, the established relation is weaker than the composition operation. The issue is the means to "associate" the QoS. The association for components is translated into a QoS combination operation. QoS combination is done according to the combination properties of the composing QoS aspects.

When dealing with SLA, we must compute the SLS according to the required and offered QoS. Therefore, the QoS (offered and required) are combined together (if they are compliant). As we see in SLA definition, a TCA rule may be associated with a behavior (a QoS). As this QoS defines a new QoS of the SLA, it must be implemented as another QoS and attached to the components linked by the SLA of the considered system. In this sense, it implies defining multiple QoS in the model and echoing them in the lower levels of the development process.

6 Example: A QoS Communication

The system considered in this example is composed by one client (`Sender`) inheriting from `Station` class. A network is linked with the client and a SLA is setup between them. The QoS of the station has a loss characteristic with a value of maximum loss

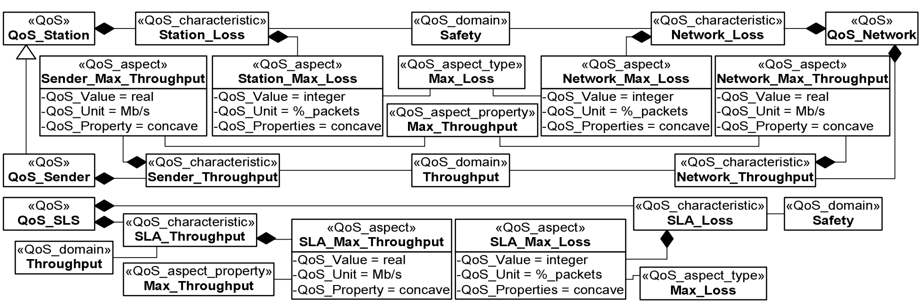


Fig. 1. QoS model

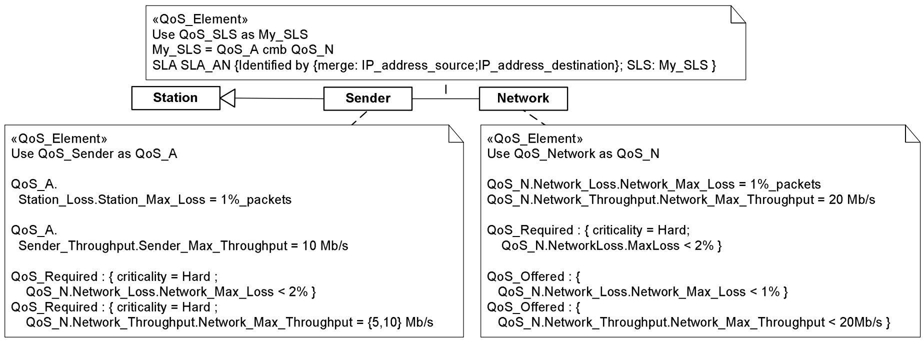


Fig. 2. System model

aspect of 2% of the total packets; the client requires a maximum throughput of 10Mb/s. The network offers a maximum loss percentage of 1% and a maximum throughput of 20Mb/s. The figure 1 gives the QoS model of our example. The `QoS_Sender` inherits from the `QoS_Station` to reflect the inheritance of `Sender` class from `Station` class. Figure 1 contains also the SLS corresponding QoS of the SLA (`QoS_SLS`). The second step consists in modeling the system in a UML class diagram as represented in figure 2. The QoS of each system element is linked in its context by the use statement. The values for each QoS are then given. The third step is to specify the QoS constraints of each system element. Sender has got two QoS required while Network element offers two QoS. The last step is to specify the SLA between the sender and the network:

```
SLA my_SLA {Identified by IP_address_source; SLS: My_SLS }
```

The SLA `my_SLA` is identified by the IP address of the source. The SLS to be enforced is modeled by the `My_SLS` QoS. The `My_SLS` QoS results in the composition of `QoS_A` (QoS of the `Sender`) and `QoS_N` (QoS of the `Network`). This operation can be done as the two QoS are compliant because all their inner QoS characteristics and QoS aspects become from the same QoS domain. The operation composes the two QoS characteristics of these QoS. As a result, the `SLA_Max_Throughput` value is given by the composition of `Sender_Max_Throughput` and `Network_Max_Throughput`. The composition property of these aspects is defined as concave. As a result, the value of the SLA QoS aspect `SLA_Max_Throughput` is fixed to 10Mb/s. The same operation is done with the other aspects. We can now check if the QoS constraints of the two system elements are fulfilled. It is the case as the maximum throughput is chosen between 5 or 10 Mb/s (QoS required for the `Sender`) and below 20 Mb/s (QoS offered from the `Network`).

7 Conclusions

In this paper, we present a language intended to specify QoS elements in QoS-aware networks. This language is suitable to define QoS and its components, in addition to QoS constraints. We give the means to handle the QoS once it is defined. We also present a means to specify Service Level Agreement. We explain how to integrate our language to a UML modeling approach suitable for networks. We extend [10] for our work to be integrated in a QoS oriented methodology based on UML for developing networks. We present restrictions for UML combination operations for UML to be fully compliant with our language. The way to complete this work includes the investigation of the dynamic QoS aspects. Another item to deal with is scalability, and studying ways to ease the modeling of large systems such as DiffServ domains.

Bibliography

1. Object Management Group, “Unified Modeling Language v1.5”, formal/03-03-01, 2003
2. Frølund S. and Koistinen J., “QML: A Language for Quality of Service Specification”, Hewlett-Packard Labs Technical Report, February 1998
3. Aagendal J.O., Quality of Service Support in Development of Distributed Systems, PhD Thesis, Department of Informatics, University of Oslo, 2001
4. Zinky J.A., Baken D.E., and Schantz R.E, “Architectural Support for Quality of Service for CORBA Objects”, Theory and Practice of Objects Systems, 1997
5. OMG, “Schedulability, Performance and Time Profile”, formal/03-09-01, 2003
6. OMG, “UML Profile for Modeling Quality of Service and Fault Tolerance Characteristics and Mechanisms”, document ptc/04-06-01, 2004
7. Mammeri Z., “Towards a formal model for QoS specification and handling in Networks”, IWQoS 2004, Montreal, Canada, June 7-9, 2004. pp. 148-152
8. Wang Z. and Crowcroft J., “Quality of Service Routing for Supporting Multimedia Applications”, IEEE JSAC, 14(7):1288-1234, 1996
9. Grossman D., “New Terminology and Clarifications for Diffserv”, RFC 3260, IETF, 2002
10. Teyssié C. and Mammeri Z., “QoS-aware Network Design with UML”, Lecture Notes in Computer Science n° 3079, pp1019-1032, Toulouse, France, July 2004
11. Donaldson A.J.M. and Turner K.J., “Formal Specification of QoS properties”, Proceedings of Workshop on Distributed Multimedia Applications and QoS Verification, pp 1-14. CRIM, Montreal, Canada, June 1994
12. Teyssié C., and Mammeri Z., “Integrating a Quality of Service Specification Language in UML”, internal report, IRIT Lab, Toulouse, 2004
13. Gu X. and Nahrstedt K., “Visual Quality of Service Specification for Distributed Heterogeneous Systems”

Modeling User-Perceived QoS in Hybrid Broadcast and Telecommunication Networks

Michael Galetzka[1], Günter Elst[1], and Adolf Finger[2]

[1] Fraunhofer Institute for Integrated Circuits, Branch Lab Design Automation,
Zeunerstr. 38, 01069 Dresden, Germany
{Michael.Galetzka, Guenter.Elst}@eas.iis.fhg.de
[2] Dresden University of Technology, Communications Laboratory
01062 Dresden, Germany
finger@ifn.et.tu-dresden.de

Abstract. In this paper, basic ideas for modeling user-perceived quality of services in hybrid networks will be presented. Such services are not restricted to an end-to-end data transmission, but may include, for example, local or cooperative caching mechanisms. Hence, a more general understanding of (data) services and an appropriate definition of parameters of user-perceived quality of these services will be discussed. Modeling in this context does not only include pure network characteristics like data rate and error probability, but also may cover parameters like application structure and characteristics, characteristics of the actual terminal equipment, user profiles, and information about current location or motion of the user. The objective of this modeling approach is to quantify these complex dependencies to support planning and operating services in future hybrid networks with an appropriate user-perceived quality.

1 Introduction

Discussions about next generation networks share a common vision: In the future, everyone is expected to be able to receive and exchange information regardless of his location and situation. For a user it has to be transparent which communication technology a dedicated service is based on. This becomes essential as there will be an increasing variety of converging network technologies within these scenarios. [10]

On the other hand, the provider of the service or service package – who again will use (and pay for) services of different network providers – must be able to plan, configure, and optimize the utilization of the underlying hybrid network according to the requirements of his customers and depending on the available network resources. Against this background, there is a need for the service provider as well as for the end-user's benefit to make the (user-perceived) quality of service (QoS) somehow calculable.

Traditionally, QoS is a term applied in telecommunication networks. With the upcoming of multimedia streaming services in the Internet, issues resulting from the original "best effort" policy have to be handled here, too. Relating QoS concepts in the Internet are focused on a packet-based end-to-end communication. Typical QoS parameters like "delay" and "loss" refer to this packet transport at different layers.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 286 – 295, 2005.

Alternatively, there is a common hope, that the continuously increasing bandwidth in the Internet will overcome these bottlenecks. This might have been true in the past, particularly for the core network. Anyway, there will be QoS relevant technological challenges in the converging next-generation networks caused by a variety of access networks. In particular, wireless (including broadcast) networks will be integrated for the distribution of multicast content. Broadcast networks are interesting in these scenarios not only because of the high bandwidth at a relatively high mobility, but also because of their specific characteristics relating to streaming and multicast.

In this context, the term "hybrid networks" describes scenarios of converging networks with broadcast networks (mainly for the distribution of multimedia content and/or other multicast content) and communication networks (stationary, portable, or mobile for the interaction and the unicast content). Several projects are dealing with this subject (e.g. [3]).

In fact, we have to discuss the question whether multicast is limited to isochronous data transmission only – which is a particular advantage of using broadcast networks within such scenarios. Rather, the exploitation of local or even cooperative caching mechanisms [4] may extend the usability of multicast from a strict isochronous transmission to a weaker interpretation of real-time data provision making broadcast networks usable for a broader range of data services.

Starting with definitions of the relevant terms, the basic modeling approach will be discussed in this paper. The relationship between technical constraints and the user-perceived quality of service will be illustrated with an example of data services in digital TV.

2 Quality of Service

In the field of telecommunications, a "service" is defined as the ability of a network to transmit dedicated information [12]. Historically, there is a close association between service, service provider and the network. Economically, a service is the non-material equivalent of a good. Even if we focus here on networks and hence our services are "data services" in some respects, we have to consider a service in a more general context and as independent of a dedicated network technology as possible.

For example, it could be imaginable that one service package "soccer information" covering applications like TV transmissions, online ticker, SMS or MMS service, radio broadcast etc. would be offered instead of single, separate services of different service providers in different networks to be paid for separately by the user. Of course, such service is technically not independent of the underlying networks. Especially, the quality of this service the user perceives is influenced by the quality of data transmission over the network(s).

2.1 Quality of Service

A lot of different definitions and parameters for "quality of service" can be found. In [7], QoS is described as "the collective effect of service performance which determines the degree of satisfaction of a user of the service". Related terms in a broader context are: Application level QoS, Quality of Experience (QoE), Quality of Business (QoB), Quality of Context (QoC) etc. [13], [11], [2].

In [8], a framework for communications QoS is defined. End-user multimedia QoS categories are characterized in [9]. Delay, delay variation and information loss are identified as the key parameters there. As shown in Fig. 1, this reveals an obvious difficulty: Only the service provider is able to define and measure the QoS parameters based on the QoS parameters of the network(s) used. However, these technical parameters have to be mapped to the user perceived categories of QoS which itself focus on certain user-perceived effects rather than on their causes within the network. There are related works and standards utilizing this framework (e.g. [1]).

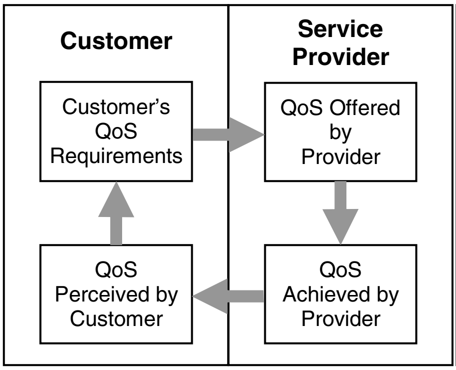


Fig. 1. The four viewpoints of QoS [8]

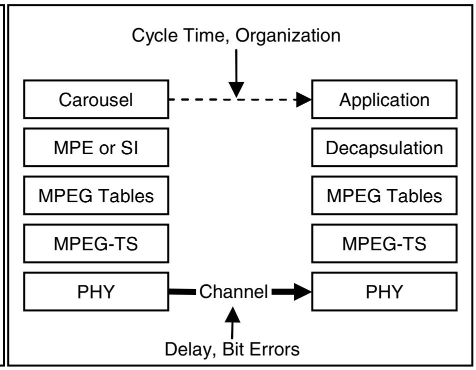


Fig. 2. Parameters across transport layers

2.2 Definition of QoS Parameters

The mapping problem mentioned above does not only exist between customer and service provider. Additionally, it has to be considered between service provider and several network providers, too. Moreover, each network layer has its own utilization of QoS parameters and therefore a mapping has to be performed between these layers, as well. Fig. 2 shows the layers interesting for digital TV (DVB) transmission as used in the example below. Note, that for example the delay at application level does not only depend on the delay at the lower layers. It is mainly affected by the cycle time of the carousel. Even more, bit errors on the channel will not necessarily result in a certain loss rate at application level but in additional cycles and hence a greater delay.

Following the explanations above, we are going to define general QoS components which cover the end-user view. Additionally, they can be extended to reflect network layer aspects as well. In [14] four parameters are mentioned: availability, performance, accuracy, and affordability. Even though this work focuses on IT services, these components may be used for our general purposes, as well.

– *Availability.* This term does not only define whether a certain service is available or not. It also may include statements about the completeness of data for a certain service or application.

– *Timeliness.* We use "timeliness" as the user perceived quality component of performance. This component gives us the measure whether the user's requests for data are satisfied in an appropriate time.
– *Accuracy.* This term generally tells us whether the data presented to the user are correct. Accuracy may be affected e.g. by data corruption during transmission as well as by outdated cache data.
– *Affordability.* This represents the costs necessary to achieve a certain degree of the other QoS components availability, timeliness, and accuracy.

These definitions of QoS components reflect the user's view of the service. The actual interpretation is specific to a certain service as well as the mapping to the related QoS parameters of the underlying network(s).

3 Modeling User Perceived QoS

If we want to quantify the QoS components defined above, we need a model which contains all aspects of our service. This starts with specific characteristics of the service represented by a certain application running on one or more kinds of terminals with different features. These terminals may be used by one or several person(s) with different needs and preferences. The persons may move and hence their location has to be considered in the model. Depending on their location, they may be faced with different reception conditions of the networks supported by a certain service and/or terminal – and so after all, the network QoS of the participating networks is (only) one of the parameters to be included in our model. Fig. 3 gives an overview of these dependencies.

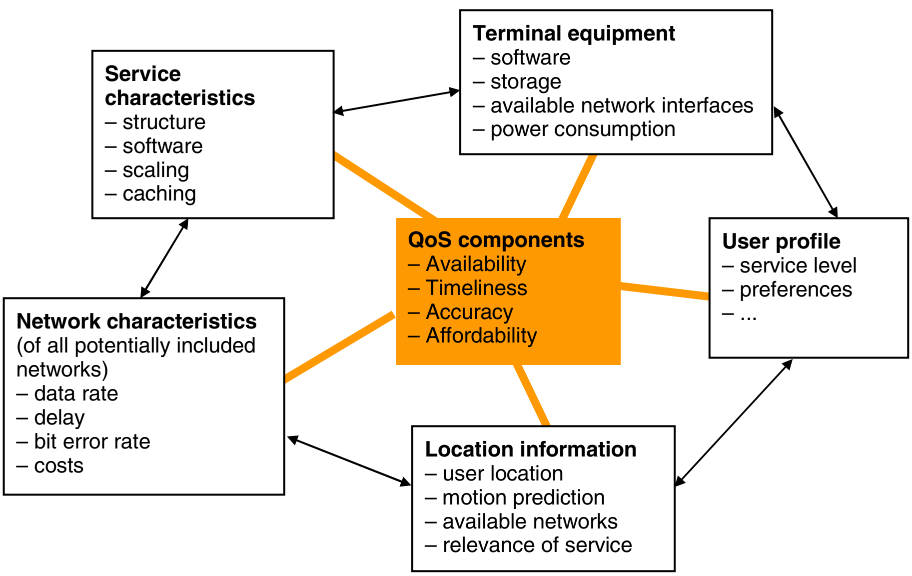


Fig. 3. Parameter dependencies for QoS modeling

We can define a set of u use cases describing in which way a user utilizes this service. For the vector of user-perceived QoS parameters

$$\underline{Q} = f_W\left(\underline{Q}_0,\ldots,\underline{Q}_u\right) \tag{1}$$

the function f_W might be some kind of service specific weight function for the QoS parameters of all use cases. For each use case the QoS parameters have to be determined from the model. This is again service specific, and that's why we will demonstrate it with an example.

3.1 Example

As an example, we intend to analyze a service which might be regarded as a key application in digital TV – an Electronic Program Guide (EPG) [6]. An EPG can be used as an advanced navigation tool through the great variety of TV programs. It assists in zapping through the channels as well as finding events at a certain time.

Fig. 4. Now & next view of an EPG

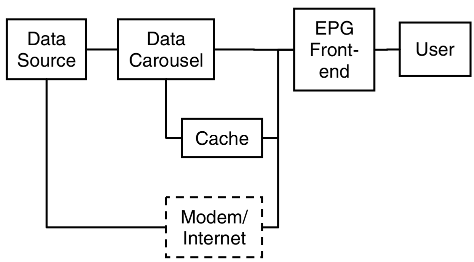


Fig. 5. Functional implementation of an EPG

The EPG application uses program data provided by the TV broadcasters, and hence it is a data service as defined above. EPG data may be transmitted within Event Information Tables (EIT) as part of the so-called DVB Service Information (SI) [5], or they may be provided by a service provider within a separate data stream. In both cases, data have to be transmitted continuously in a data carousel because of the broadcast nature of the data channel.

For example, the functional requirements are:

– Getting a quick overview of all currently running events and of the events starting immediately after them (Fig. 4).
– Getting an overview of events running at a certain time in the near future.
– Presenting extended information for selected events.

Additional requirements regarding the user-perceived quality are:

– Always presenting the up-to-date event information, even if the schedule changes.
– Accessing the EPG data within an appropriate time.

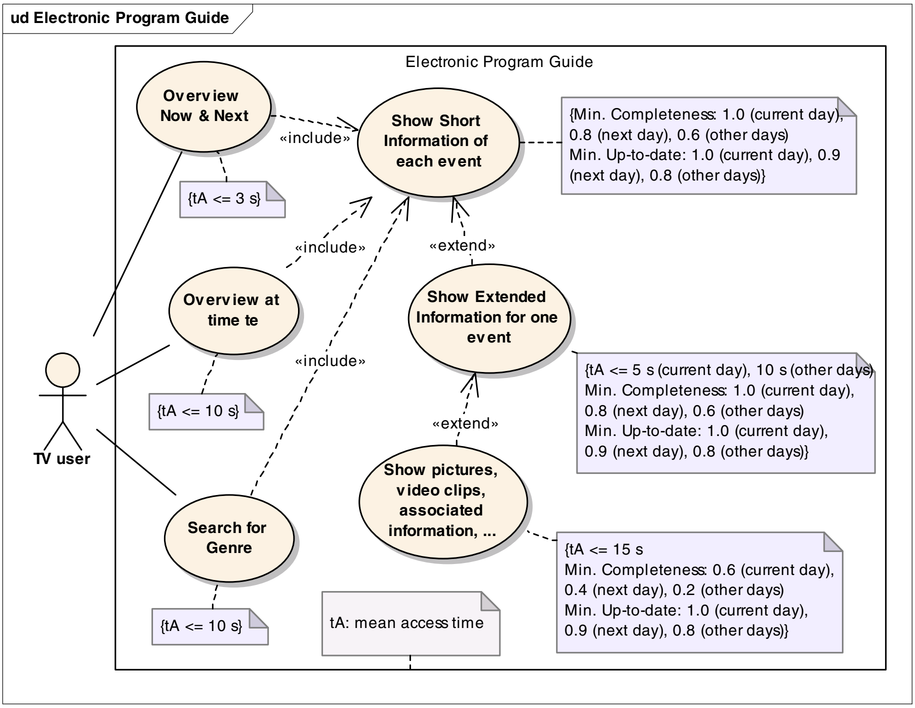


Fig. 6. Use case diagram for Electronic Program Guide

Fig. 6 shows a use case diagram for such EPG service including QoS requirements. Corresponding to the general QoS components defined above, we define the mean access time t_A (timeliness), the degree of completeness C (availability) and the degree of up-to-dateness A (accuracy) as the specific QoS parameters for the EPG service. The affordability will not be dealt with in this example. In the following we want to concentrate on use case "Overview at time t_e" (which in fact is a generalization of use case "Overview Now & Next", but with different QoS requirements) with short and extended event information. We suppose an implementation (Fig. 5) which optionally uses a caching mechanism to avoid long access times to the carousel. For simplification, we further assume a single private EPG data stream rather than using the EIT on several transport streams. Additionally, the figure shows a return channel which could serve as an opportunity to access supplementary EPG data, e.g. pictures, video clips. The latter case will not be discussed in detail here.

Modeling the Data Carousel. We presume an average number of events per day per channel of $e_{Day} = 25$. There are data available for $p_{max} = 100$ channels on 8 days with a data rate of $r_C = 1$ MBit/s. Data are sent in one file per day. Let $E_S = 100$ be the average size of short event information (in byte) and $E_E = 300$ the average size of extended event information.

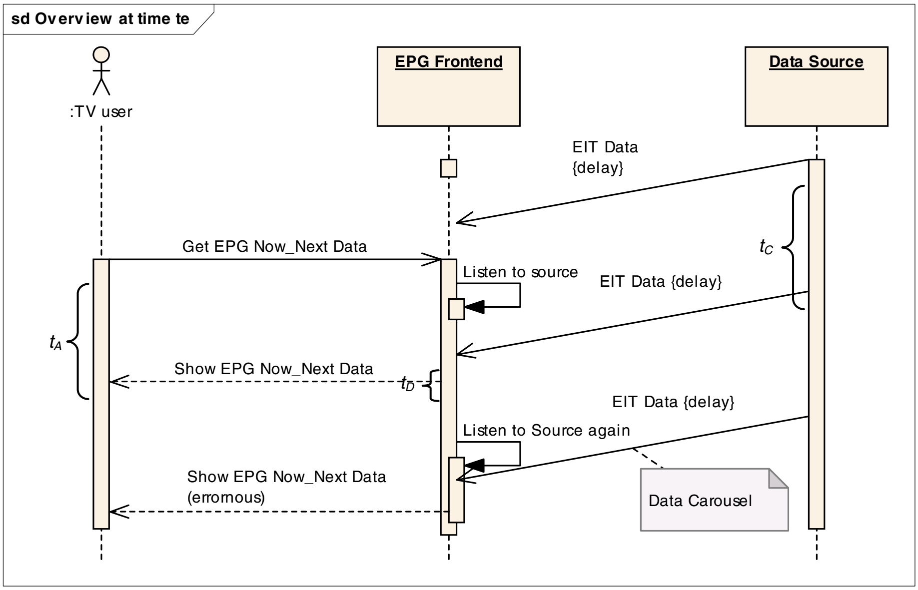


Fig. 7. Sequence diagram for EPG use case "Overview at time t_e"

Timeliness: For the cycle time t_c of the data carousel we get

$$t_C = 8 \cdot e_{Day} \cdot p_{\max} \cdot \left(E_S + E_E\right) / r_C = 64s\ . \tag{2}$$

Therefore, the download time for one data file is $t_D = 8\ s$. As we can see in Fig. 7, the actual access time is not dependent on any delay on the channel. It rather depends on the cycle time t_c of the carousel, the download time t_D and some processing time, which shall be neglected here. If we assume, that the user requests occur uniformly distributed within the cycle time interval, for the average access time t_A applies:

$$t_A = 0.5 \cdot t_C + t_D = 40s \tag{3}$$

If the data transmission is erroneous, the download time will increase by one cycle t_c. Supposing a bit error rate of lower than 10^{-11} for a satellite channel, t_A would increase only by about 5 ms. However, for a terrestrial channel and mobile reception, we would have to consider a location dependent bit error rate, which is expected to have a stronger influence on the QoS.

Availability, Accuracy: Assuming that the data source is complete and correct, both values would be 1.

Modeling the Cache

Timeliness: We only have to consider a constant processing time which can be neglected here.

Availability: Above we defined completeness C as the specific parameter for availability within the EPG service. It depends on the cache size and the caching strategy. We presume that events will be cached with short and extended information until half of the cache is filled, beginning with the earliest events. After that, only short information will be stored until maximum cache size is reached. If we normalize time to the average number of events per day, we get an event time t_e. $t_e = 0$ denotes the time of the currently running event, $t_e = 1$ the following and so on. $t_{e,full}$ shall be the event time of the latest event stored with both the short and the extended information, $t_{e,max}$ accordingly denotes the event time of the latest event stored in the cache at all. Let $n_c = 1$ MByte be the maximum cache size and $p = 50$ the number of TV programs configured for the EPG. Then for $t_{e,full}$ and $t_{e,max}$ applies:

$$t_{e,full} = \frac{0.5 \cdot n_C}{p \cdot (E_S + E_E)} \approx 26 \tag{4}$$

(which is about 1 day) and

$$t_{e,\max} = \frac{0.5 \cdot n_C + p \cdot t_{e,full} \cdot E_S}{p \cdot E_S} \approx 130 \tag{5}$$

(about 5 days). Then we get for the completeness at event time t_e:

$$C_S(t_e) = \begin{cases} 1 & 0 \le t_e \le t_{e,\max} \\ 0 & t_e > t_{e,\max} \end{cases} \tag{6}$$

for short information only and for short and extended information

$$C_E(t_e) = \begin{cases} 1 & 0 \le t_e \le t_{e,full} \\ 0 & t_e > t_{e,full} \end{cases} \tag{7}$$

Accuracy: We assume that the cache will be filled at a suitable time $t = -t_u$ before current time $t = 0$. Additionally we suppose that we know a distribution function $a_U(t)$ for the probability that one event information will be changed editorially in the data stream at a certain time t. Furthermore, let the distribution function $a_E(t)$ describe the probability that at a certain time t an originally planned TV event is out-dated because of some current incident. Both distributions are implied to be independent. Then, for the probability $A_{Cache}(t_e)$ that there are no out-dated events in our EPG service between now $t = 0$ and a certain time t_e there applies:

$$A_{Cache}(t_e) = 1 - \int_{-t_U}^{t_e} a_U(t)dt \cdot \int_{0}^{t_e} a_E(t)dt \tag{8}$$

Modeling the EPG Front-end. Finally, we have to compute the QoS parameters which directly influence the user's perception. Within the EPG front-end in our example the implementation of the EPG service controls whether a user request has to be forwarded to the cache or must be fulfilled by obtaining data directly from the carousel (Fig. 8).

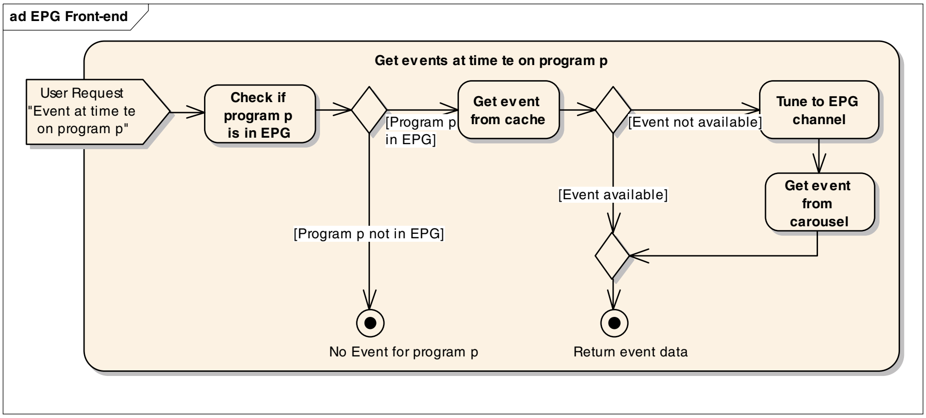


Fig. 8. Activity diagram for EPG front-end

Timeliness: The mean access time now depends on the event time t_e and can be expressed as

$$t_A(t_e) = \begin{cases} t_{A,Cache} & 0 \le t_e \le t_{e,\max} \\ t_{A,Carousel} & t_e > t_{e,\max} \end{cases} \tag{9}$$

Availability: Because of the possibility to receive data from the carousel, the completeness $C(t_e)$ of data should always be 1 (presumed the respective event information is provided at all).

Accuracy: If we obtain data from the cache we cannot check if these data are really up-to-date. Therefore, for accuracy $A(t_e)$ applies:

$$A(t_e) = \begin{cases} A_{Cache}(t_e) & 0 \le t_e \le t_{e,\max} \\ 1 & t_e > t_{e,\max} \end{cases} \tag{10}$$

The usage of a return channel as mentioned in Fig. 5 (e.g. for acquiring additional EPG data like photos, video clips) would contribute further parameters relevant for the overall user-perceived QoS (e.g. speed of the modem, setup time of a connection).

4 Summary and Outlook

The relatively simple example discussed above demonstrates a number of problems which have to be considered when modeling user-perceived quality of service in hybrid networks. Starting from a detailed understanding and modeling of the application realizing a certain service, each of the general QoS parameters has to be defined and described carefully. According to Fig. 3, our model may be influenced by a variety of parameters. Many of them need to be described as probabilistic models. Recalling the "soccer information" service package example in this paper, we need to model user's habits with distribution functions for their location (e.g. on Saturday afternoon) as well as the available networks and their quality at these locations. A hierarchical modeling methodology may even make use of existing methods for network planning and modeling of network QoS.

Quantifying the parameters for user-perceived QoS will be essential for planning and operating services in future complex network structures with service level agreements to guarantee a certain level of QoS for the users on the one hand, and to calculate the necessary resources and the corresponding cost for the service provider on the other hand. Rather than in the simple example above, this may be done by hierarchical simulation techniques for more complex service structures.

References

1. 3GPP TS 23.107: Quality of Service (QoS) concept and architecture v6.0.0. 3rd Generation Partnership Project (2003)
2. Buchholz, T., Küpper, A., Schiffers, M.: Quality of Context: What It Is And Why We Need It. In: HP OpenView University Association 10th Workshop, Geneva, Switzerland (2003)
3. CISMUNDUS: Field Test Results and Demonstration. CISMUNDUS IST- 2000-29255 Project Deliverable 9 (2004)
4. Cao, G., Yin, L., Das, C. R.: Cooperative Cache-Based Data Access in Ad Hoc Networks. In: IEEE Computer 2 (2004) 32-39
5. EN 300 468: Specification for Service Information (SI) in DVB systems - ETSI European Standard EN 300 468 V1.5.1. European Telecommunication Standards Institute (ETSI) (2003)
6. Galetzka, M., Lux, S.: Elektronische Programmführer - ein technischer Überblick. In: Fernseh- und Kinotechnik 10 (2004) 483-489
7. ITU-T E.800: Terms and definitions related to quality of service and network performance including dependability - ITU-T Recommendation E.800. International Telecommunication Union (1994)
8. ITU-T G.1000: Communications quality of service: A framework and definitions - ITU-T Recommendation G.1000. International Telecommunication Union (2001)
9. ITU-T G.1010: End-user multimedia QoS categories - ITU-T Recommendation G.1010. International Telecommunication Union (2001)
10. ITU: Broadband Mobile Communications Towards A Converged World. In: ITU/MIC Workshop an Shaping the Future Mobile Information Society, Seoul (2004)
11. Moorsel, A. v.: Metrics for the Internet Age: Quality of Experience and Quality of Business. In: Fifth International Workshop on Performability Modeling of Computer and Communication Systems, Erlangen (2001) 26-31
12. Siegmund, G.: Technik der Netze. Hüthig Verlag, Heidelberg (2002)
13. Siller, M., Woods, J.: Improving Quality of Experience for Multimedia Services by QoS Arbitration on a QoE Framework. In: IEEE Packet Video, Nantes (2003)
14. Sturm, R. C.: Managing Quality of Service. InfoVista White Paper (1997)

Holistic and Trajectory Approaches for Distributed Non-preemptive FP/DP* Scheduling

Steven Martin[1] and Pascale Minet[2]

[1] Université Paris 12, LIIA, 120 rue Paul Armangot,
94 400 Vitry, France
steven.martin@esiee.org

[2] INRIA, Domaine de Voluceau, Rocquencourt,
78 153 Le Chesnay, France
pascale.minet@inria.fr

Abstract. In this paper, we are interested in real-time flows requiring quantitative and deterministic Quality of Service (QoS) guarantees. We focus more particularly on two QoS parameters: the worst case end-to-end response time and jitter. We consider a non-preemptive scheduling of flows, called FP/DP*, combining fixed priority and dynamic priority established on the first node visited in the network. Examples of such a scheduling are FP/FIFO* and FP/EDF*. With any flow is associated a fixed priority denoting the importance of the flow from the user point of view. The arbitration between packets having the same fixed priority is done according to their dynamic priority. A classical approach used to compute the worst case end-to-end response time is the holistic one. We show that this approach leads to pessimistic upper bounds and propose the trajectory approach to improve the accuracy of the results. Indeed, the trajectory approach accounts for worst case scenarios experienced by a flow along its trajectory. It then eliminates scenarios that cannot occur.

Keywords: Fixed priority scheduling, QoS, holistic approach, worst case end-to-end response time, trajectory approach, deterministic guarantee.

1 Context and Motivations

In this paper, we are interested in real-time applications that require bounds on the worst case end-to-end response times and jitters to have a behavior compliant with their specifications (e.g. Voice over IP and control-command applications). That is why we focus on deterministic guarantees of end-to-end response times and jitters in a packet network. We will show how to determine these times depending on the flow scheduling used in the network. With regard to flow scheduling, the assumption generally admitted is that packet transmission is not preemptive. Moreover, *Fixed Priority* scheduling has been extensively studied in the last years [1, 2]. It exhibits interesting properties. Indeed, the impact of a

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 296–305, 2005.

new flow is limited to flows having equal or lower fixed priorities, it is easy to implement and well adapted for service differentiation.

In a network, several packets can share the same fixed priority: for example, if the number of fixed priorities is less than the flow number, or if flows are processed by service class and the flow priority is this of its class. We consider that such packets are scheduled according to their dynamic priorities (DP) and unlike the state of the art, we account for this arbitration to compute the worst case end-to-end response times. More precisely, we assume that packets are scheduled according to the non-preemptive FP/DP* scheduling. With FP/DP*, packets are first scheduled according to their fixed priority (FP). Packets with the same fixed priority are scheduled according to their dynamic priorities, computed on the first node visited (expressed by the star in DP*). Most famous FP/DP* scheduling algorithms are FP/FIFO* and FP/EDF*. Notice that with FP/DP*, the order of packet priority does not depend on the node considered: it is fixed. Unlike DP, DP* does not require to synchronize all clocks in the network but only those of ingress nodes, where dynamic priorities are assigned to flow packets.

The first approach introduced to compute the worst case end-to-end response time of a flow is the holistic approach that provides pessimistic bounds in some configurations. That is why we introduce the trajectory approach taking into account the worst case scenario experienced by a flow along its trajectory (path).

2 Problematic

2.1 Assumptions and Models

We investigate the problem of providing a deterministic guarantee (i.e., an upper bound) on the end-to-end response time to any flow in a network. As we make no particular assumption concerning the arrival times of packets in the network, the feasibility of a set of flows is equivalent to meet the requirement, whatever the arrival times of the packets in the network. In the following, we assume that time is discrete. Reference [3] shows that results obtained with a discrete scheduling are as general as those obtained with a continuous scheduling when all flow parameters are multiples of the node clock tick. In such conditions, any set of flows is feasible with a discrete scheduling if and only if it is feasible with a continuous scheduling. Moreover, we assume the following models.

Scheduling Model. All nodes in the network schedule packets according to the non-preemptive FP/DP* algorithm. For instance, we assume that either all nodes use FP/FIFO* or all nodes use FP/EDF*. Moreover, we assume that packet scheduling is non-preemptive. Hence, the node scheduler waits for the completion of the current packet transmission (if any) before selecting the next packet.

Network Model. We consider a network where links interconnecting nodes are supposed to be FIFO and the network delay between two nodes has known lower and upper bounds: L_{min} and L_{max}. Moreover, we consider neither network failures nor packet losses.

Traffic Model. We consider a set $\tau = \{\tau_1, ..., \tau_n\}$ of n sporadic flows. Each flow τ_i follows a path $\mathcal{P}_i$ that is an ordered sequence of nodes whose first node is the ingress node of the flow. Moreover, a sporadic flow τ_i is defined by:

- T_i, the minimum interarrival time between two successive packets of τ_i;
- C_i^h, the maximum processing time on node $h \in \mathcal{P}_i$ of a packet of τ_i;
- J_i, the maximum release jitter of packets of τ_i at its ingress node. A packet is subject to a release jitter if there exists a non-null delay between its generation time and the time where it is taken into account by the scheduler;
- D_i, the end-to-end deadline of τ_i, its maximum end-to-end response time acceptable. A packet of τ_i generated at time t must be delivered at $t + D_i$.

2.2 Notations and Preliminary Results

We consider any flow τ_i, $i \in [1, n]$, following a path $\mathcal{P}_i$. We focus on the packet m of τ_i generated at time t. Then, we denote P_i, the fixed priority of flow τ_i and $P_i(t)$, the dynamic priority of packet m. We then define the three following sets: $hp_i = \{j \in [1, n]\,,\ P_j > P_i\}$, $sp_i = \{j \in [1, n]\,,\ j \neq i\,,\ P_j = P_i\}$ and $lp_i = \{j \in [1, n]\,,\ P_j < P_i\}$. For flows τ_j sharing the fixed priority of flow τ_i, we have to distinguish between (i) flows that are able to generate packets with a dynamic priority higher than this of packet m and (ii) flows that are not able to generate such packets. Then, for any time $t \geq -J_i$, we denote: $sp_i(t) = \{j \in sp_i\,,\ P_j(-J_j) \geq P_i(t)\}$ and $\overline{sp}_i(t) = \{j \in sp_i\,,\ P_j(-J_j) < P_i(t)\}$.

Definition 1. *Let m be the packet of flow τ_i generated at time t. For any flow τ_j, if $j \in sp_i(t)$, $G_{j,i}(t)$ is the time beyond which τ_j can no longer generate packets with a dynamic priority higher than this of m: $\forall t' \in [-J_j, G_{j,i}(t)]$, $P_j(t') \geq P_i(t)$.*

For example, we get for any flow τ_j, $j \in sp_i(t)$, $G_{j,i}(t) = t$ if the scheduling algorithm is FP/FIFO*, whereas $G_{j,i}(t) = t + D_i^{first_i} - D_j^{first_j}$ if the scheduling algorithm is FP/EDF*, where $D_i^{first_i}$ is the relative deadline attributed to flow τ_i on its first node visited, denoted $first_i$. The assignment of this local deadline is out of the scope of this paper. The local deadline is computed from the end-to-end deadline. For instance, with a uniform assignment, the local deadline is equal to $\lfloor D_i/q \rfloor$, where q denotes the number of nodes visited by τ_i.

Hence, priority of packet m is higher than or equal to this of packet m' belonging to any flow τ_j and generated at time t' if and only if: $P_i > P_j$ or ($P_i = P_j$ and $P_i(t) \geq P_j(t')$). Moreover, we denote $h' <_i h$ (resp. $h' >_i h$) if node h' is visited before (resp. after) node h by flow τ_i. We also denote:

- $\mathcal{P}_i = [first_i, ..., last_i]$, the path followed by flow τ_i, with $first_i$ (resp. $last_i$) the ingress node (resp. the egress node) of the flow in the network;
- $|\mathcal{P}_i|$, the cardinal of path $\mathcal{P}_i$, that is the number of nodes visited by flow τ_i;
- $first_{j,i}$ (resp. $last_{j,i}$), the first (resp. the last) node of $\mathcal{P}_i$ visited by flow τ_j;
- $slow_i$, the slowest node visited by τ_i on path $\mathcal{P}_i$: $\forall\, h \in \mathcal{P}_i$, $C_i^{slow_i} \geq C_i^h$;
- $slow_{j,i}$, the slowest node visited by τ_j on path $\mathcal{P}_i$: $\forall\, h \in \mathcal{P}_i \cap \mathcal{P}_j$, $C_j^{slow_{j,i}} \geq C_j^h$;
- R_i, the worst case end-to-end response time of flow τ_i;
- R_i^h, the maximum response time of flow τ_i on node h;
- Jin_i^h, the jitter of flow τ_i when entering node h. Notice that $Jin_i^{first_i} = J_i$;
- $Smin_i^h$ and $Smax_i^h$, respectively the minimum and the maximum time taken by a packet of flow τ_i to go from its source node to node h;
- $W_i^h(t)$, the latest starting time on node h of the packet of τ_i generated at t.

Moreover, we assume, with regard to flow τ_i following path $\mathcal{P}_i$, that any flow τ_j, $j \in hp_i \cup sp_i$ following path $\mathcal{P}_j$ with $\mathcal{P}_j \neq \mathcal{P}_i$ and $\mathcal{P}_j \bigcap \mathcal{P}_i \neq \emptyset$ never visits a node of path $\mathcal{P}_i$ after having left this path.

Assumption 1. it For any flow τ_i following path $\mathcal{P}_i$, for any flow τ_j, $j \in hp_i \cup sp_i$, following path $\mathcal{P}_j$ such that $\mathcal{P}_j \cap \mathcal{P}_i \neq \emptyset$, we have either $[first_{j,i}, last_{j,i}] \subseteq \mathcal{P}_i$ or $[last_{j,i}, first_{j,i}] \subseteq \mathcal{P}_i$.

Definition 2. *The end-to-end jitter of any flow τ_i, $i \in [1, n]$, is the difference between the maximum and minimum end-to-end response times of τ_i packets, that is equal to: $R_i - (\sum_{h \in \mathcal{P}_i} C_i^h + (|\mathcal{P}_i| - 1) \cdot Lmin)$.*

Non-preemptive Effect. As packet scheduling is non-preemptive, if a packet m of any flow τ_i arrives on node h while a packet $m' \in lp_i \cup \overline{sp}_i(t)$ is being processed, m has to wait until m' completion. However, the non-preemptive effect is not limited to this waiting time. The delay incurred by m on node h directly due to m' may lead to consider packets $\in hp_i \cup sp_i(t)$, arrived after m on the node but before m starts its execution. Then, we denote $\delta_i(t)$, the maximum delay incurred by packet m while following its path, directly due to the non-preemptive effect.

Property 1. *Let τ_i, $i \in [1, n]$, be a flow following path $\mathcal{P}_i = [first_i, ..., last_i]$. When flows are scheduled FP/DP*, the maximum delay incurred by the packet of τ_i generated at time t directly due to flows belonging to $lp_i \cup \overline{sp}_i(t)$ meets:*

$$\delta_i(t) \leq \max(0\,; \max_{\substack{j \in lp_i \cup \overline{sp}_i(t) \\ first_{j,i} = first_i}} \{C_j^{first_i}\} - 1) + \sum_{\substack{h \in \mathcal{P}_i \\ h \neq first_i}} \max(0\,; \max_{\substack{j \in lp_i \cup \overline{sp}_i(t) \\ first_{j,i} = h}} C_j^h - 1\,;$$
$$\max_{\substack{j \in lp_i \cup \overline{sp}_i(t) \\ h \in (first_{j,i}, last_{j,i}] \\ first_{j,i} \neq first_{i,j}}} \{C_j^h\} - 1\,; 1_\alpha \cdot (\max_{\substack{j \in lp_i \cup \overline{sp}_i(t) \\ h \in (first_{j,i}, last_{j,i}] \\ first_{j,i} = first_{i,j}}} \{C_j^h\} - C_i^{pre_i(h)} + Lmax - Lmin)),$$

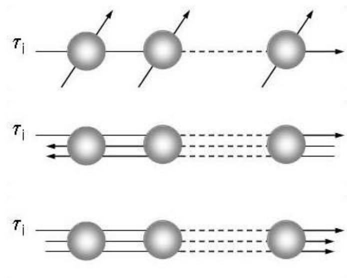
where $\forall\, h \in \mathcal{P}_i$, $\max_{j \in lp_i \cup \overline{sp}_i(t)} \{C_j^h\} = 0$ if $lp_i \cup \overline{sp}_i(t) = \emptyset$ and $1_\alpha = 1$ if $lp_i \cup \overline{sp}_i(t) \neq \emptyset$ and 0 otherwise.

Proof: See [4]. ∎

2.3 Configurations Studied

We compare the bounds provided by the holistic and trajectory approaches on:

Rake, where flow τ_i visits $q = n - 1$ nodes numbered from 1 to q whereas any other flow τ_j visits node j if $j < i$ and node $j-1$ otherwise.
Reverse path, where flow τ_i visits q nodes numbered from 1 to q whereas all other flows visit nodes q to 1.
Same path, where all flows visit the same sequence of nodes consisting of q nodes numbered from 1 to q.



3 Holistic Approach

The holistic approach [5, 6] considers the worst case scenario on each node visited by a flow, accounting for the maximum possible jitter introduced by the previous visited nodes. The minimum and maximum response times on a node h induce a maximum jitter on the next visited node $h+1$ that leads to a worst case response time and then a maximum jitter on the following node and so on. This approach can be pessimistic as it considers worst case scenarios on every node possibly leading to impossible scenarios. Indeed, a worst case scenario for a flow τ_i on a node h does not generally result in a worst case scenario for τ_i on any node visited after h. The holistic approach proceeds iteratively and starts with the first node visited. Knowing the value of $Jin_j^{first_j}$ for any $j \in [1, n]$, we compute $R_j^{first_j}$ using Property 3 giving the worst case response time in the uniprocessor case for the FP/DP* scheduling considered. We proceed in the same way for any node h, $h \in (first_j, last_j]$. Knowing the value of Jin_j^h, that is equal to $\sum_{k<_j h}(R_j^k - C_j^k + Lmax - Lmin)$ for any $j \in [1, n]$, we compute R_j^h using Property 3 and so on until node $last_j$. Notice that the computation can be divergent. In such a case, the algorithm is stopped as soon there exists a flow such that its end-to-end response time exceeds its end-to-end deadline. A bound on the end-to-end response time of flow τ_i is given by the following property.

Property 2. *When flows are scheduled FP/DP*, the worst case end-to-end response time of any flow τ_i, if bounded, is bounded by:*

$$R_i^{first_i} + \sum_{\substack{h \in \mathcal{P}_i \\ h \neq first_i}} \left(R_i^h - Jin_i^h\right) + (|\mathcal{P}_i| - 1) \cdot Lmax.$$

Proof: See [5]. ■

The computation of the worst case response time on any node visited depends on the scheduling policy. For FP/DP*, this worst case response time is given by the following property.

Property 3. *In the uniprocessor case, when flows are scheduled FP/DP* and* $\sum_{j \in hp_i \cup sp_i} C_j/T_j < 1$, *the worst case response time of flow* τ_i *is equal to:*

$R_i = \max_{t \in \mathcal{S}'_i} \{W_i(t) - t\} + C_i$, *with:*

$$W_i(t) = \sum_{j \in hp_i} \left(1 + \left\lfloor \frac{W_i(t)+J_j}{T_j} \right\rfloor\right) \cdot C_j + \sum_{j \in sp_i(t)} \left(1 + \left\lfloor \frac{\min(G_{j,i}(t); W_i(t))+J_j}{T_j} \right\rfloor\right) \cdot C_j + \left\lfloor \frac{t+J_i}{T_i} \right\rfloor \cdot C_i + \delta_i(t),$$

$\delta_i(t) = \max(0; \max_{j \in lp_i \cup \overline{sp}_i(t)}\{C_j\} - 1)$ *and* $\mathcal{S}'_i = \bigcup_{t_i^0=-J_i}^{-J_i+T_i-1} \mathcal{S}'_i(t_i^0)$, *with* $\mathcal{S}'_i(t_i^0)$ *the set of times* $t = t_i^0 + k \cdot T_i$ *such that:*

- $t_i^0 \in [-J_i, -J_i + T_i[$ *and* $k \in \mathbb{N} \cap [0, K]$, *with* K *being the smallest integer such that* $t_i^0 + (K+1) \cdot T_i \geq \min\left(W_i(t_i^0 + K \cdot T_i) + C_i \,;\, \mathcal{B}_i(t_i^0)\right)$ *and* $\mathcal{B}_i(t_i^0) = \sum_{j \in hp_i \cup sp_i(t)} \left\lceil \frac{\mathcal{B}_i(t_i^0)+J_j}{T_j} \right\rceil \cdot C_j + \left\lceil \frac{\mathcal{B}_i(t_i^0)-t_i^0}{T_i} \right\rceil \cdot C_i + \max(0; \max_{j \in lp_i \cup \overline{sp}_i(t)}\{C_j\} - 1)$;
- $\exists\ j$ *and* $l \in sp_i(t) \cup \{i\}$ *such that* $G_{j,i}(t) = -J_l + k_l \cdot T_l$, $k_l \in \mathbb{N}$.

Proof: See [4]. ■

Worst Case Response Time Computation Algorithm. To compute the worst case end-to-end response time of any flow τ_i with the holistic approach, we apply the following algorithm: (i) we first determine the set $\mathcal{S}_i$ of flows belonging to $hp_i \cup sp_i$ and crossing directly or indirectly flow τ_i, that is any flow τ_j belongs to $\mathcal{S}_i$ iff $j \in hp_i \cup sp_i$ and τ_j directly crosses τ_i or a flow $\tau_k \in \mathcal{S}_i$, (ii) we then initialize for the iteration $q = 1$ the value of $Jin_j^h(q)$ for any flow $\tau_j \in \mathcal{S}_i$: we have $Jin_j^h(1) = 0$ if $h \neq first_j$ and J_j otherwise, (iii) we proceed iteratively:

$q = 0$
Repeat
  q=q+1
  for any flow $\tau_j \in \mathcal{S}_i$, $R_j = J_j + (|\mathcal{P}_j| - 1) \cdot Lmax$
    for $h = first_j$ to $last_j$, compute R_j^h with $Jin_k^h(q)$ for any flow $\tau_k \in \mathcal{S}_i$
      if $h = first_j$ then $Jin_j^{suc_j(first_j)}(q+1) = R_j^h - C_j^h$
      else if $h <_j last_j$ then $Jin_j^{suc_j(h)}(q+1) = Jin_j^h(q) + R_j^h - C_j^h + Lmax - Lmin$
      $R_j = R_j + R_j^h - Jin_j^h(q)$
Until $(\exists \tau_j \in \mathcal{S}_i, R_j > D_j)$ or $(\forall \tau_j \in \mathcal{S}_i, \forall h \in \mathcal{P}_j, Jin_j^h(q+1) = Jin_j^h(q))$
where $suc_j(h)$ denotes the node visited by τ_j after node h.

4 Trajectory Approach

Unlike the holistic approach, the trajectory approach [7] is based on the analysis of the worst case scenario experienced by a packet m on its trajectory and not on any node visited. Then, only possible scenarios are examined. For instance,

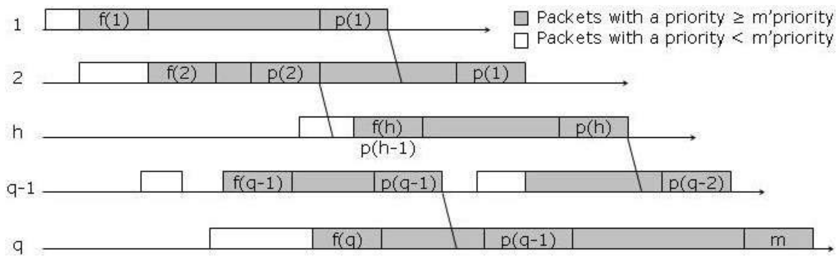


Fig. 1. Response time of packet m

the fluid model (see [8] for GPS) is relevant to the trajectory approach. More precisely, we consider any flow τ_i, $i \in [1, n]$, following a path $\mathcal{P}_i$ consisting of q nodes numbered from 1 to q. We focus on the packet m of τ_i generated at time t. As we consider a non-preemptive scheduling, we are interested in determining the latest starting time of packet m processing on its last node visited. For this, we determine the busy period of level corresponding to m's priority[1] in which m is processed on node q. Let bp^q this busy period. We define $f(q)$ as the first packet processed in bp^q with a priority higher than or equal to this of packet m. Due to the non-preemption, this packet can be delayed by at most one packet with a priority less than this of packet m (see Figure 1).

As flows do not necessarily follow the same path in the network considered, it is possible that packet $f(q)$ does not come from node $q-1$. We then define $p(q-1)$ as the first packet processed between $f(q)$ and m such that $p(q-1)$ comes from node $q-1$. Packet $p(q-1)$ has been processed in a busy period of level corresponding to m's priority on node $q-1$. Let bp^{q-1} this busy period.

We then define $f(q-1)$ as the first packet processed in bp^{q-1} with a priority higher than or equal to this of packet m. And so on until the busy period of level corresponding to m's priority on node 1 in which the packet $f(1)$ is processed. For the sake of simplicity, on a node h, we number consecutively the packets processed after $f(h)$ and before $p(h)$ (with $p(q) = m$). Hence, on node h, we denote $m' - 1$ (resp. $m' + 1$) the packet preceding (resp. succeeding to) m'.

We have thus determined the busy periods on nodes visited by m that can be used to compute the latest starting time of packet m on node q. Indeed, $W_i^q(t)$ is equal to: $\sum_{h=1}^{q} \left(\sum_{g=f(h)}^{p(h)} C_{\tau(g)}^h \right) + \delta_i(t) + (q-1) \cdot L_{max}$, with $p(q) = m$.

In [4], we have analyzed the term $\sum_{h=1}^{q} (\sum_{g=f(h)}^{p(h)} C_{\tau(g)}^h)$ and obtained Property 4, where: (i) the first term of $W_i^{last_i}(t)$ represents the maximum delay due to packets having higher fixed priorities, (ii) the second term represents the maximum delay due to packets having the same fixed priority but higher dynamic priorities, (iii) the difference between the third term and the fourth term rep-

[1] A busy period of level $\mathcal{L}$ is defined by an interval $[t, t')$ such that t and t' are both idle times of level $\mathcal{L}$ and there is no idle time of level $\mathcal{L}$ in (t, t'). An idle time t of level $\mathcal{L}$ is a time such that all packets with a priority greater than or equal to $\mathcal{L}$ generated before t have been processed at time t.

resents the maximum delay due to previous packets of flow τ_i, (iv) the term denoted $\delta_i(t)$ represents the delay directly due to the non-preemption, (v) the last term represents the maximum transmission delay. Notice that $\lfloor x \rfloor^+$ equals $\max(0\,; \lfloor x \rfloor)$ and $pre_i(h)$ denotes the node visited before node h by flow τ_i.

Property 4. *When flows are scheduled FP/DP***, if* $\sum_{j \in hp_i \cup sp_i(t)} C_j^{first_{j,i}}/T_j < 1$, *then the worst case end-to-end response time of any flow* τ_i *is bounded by:* $R_i = \max_{t \in \mathcal{S}'_i}\{W_i^{last_i}(t) - t\} + C_i^{last_i}$, *with :*

$$W_i^{last_i}(t) = \sum_{j \in hp_i} (1 + \lfloor \frac{W_i^{last_{i,j}}(t) - Smin_j^{last_{i,j}} - M_i^{first_{i,j}}(t) + Smax_j^{first_{i,j}} + J_j}{T_j} \rfloor^+) \cdot C_j^{slow_{j,i}}$$

$$+ \sum_{j \in sp_i(t)} (1 + \lfloor \frac{\min(G_{j,i}(t)\,;\, W_i^{last_{i,j}}(t) - Smin_j^{last_{i,j}}) - M_i^{first_{i,j}}(t) + Smax_j^{first_{i,j}} + J_j}{T_j} \rfloor^+) \cdot C_j^{slow_{j,i}}$$

$$+ (1 + \lfloor \frac{t + J_i}{T_i} \rfloor) \cdot C_i^{slow_i} - C_i^{last_i} + \sum_{\substack{h \in \mathcal{P}_i \\ h \neq slow_i}} \max_{\substack{j \in hp_i \cup sp_i(t) \cup \{i\} \\ first_{j,i} = first_{i,j}}} \{C_j^h\} + \delta_i(t) + (|\mathcal{P}_i| - 1) \cdot Lmax,$$

with $M_i^{first_{i,j}}(t) = \sum_{h = first_i}^{pre_i(first_{i,j})} \left(\min_{\substack{j \in hp_i \cup sp_i(t) \cup \{i\} \\ first_{j,i} = first_{i,j}}} \{C_j^h\} + Lmin \right)$ *and* $\mathcal{S}'_i$ *the set of times* t *such that:*

- $-J_i \le t < \bar{t}_i^{\emptyset} + \mathcal{B}_i^{slow_i}$, *where* $\bar{t}_i^{\emptyset}$ *is the first time* t *such that* $\overline{sp}_i(t) = \emptyset$ *and* $\mathcal{B}_i^{slow_i} = \sum_{j \in hp_i \cup sp_i(t) \cup \{i\}} \left\lceil B_i^{slow_i}/T_j \right\rceil \cdot C_j^{slow_{j,i}}$;
- $\exists\, j$ *and* $l \in sp_i(t) \cup \{i\}$ *such that* $G_{l,i}(t) = -J_j + k_j \cdot T_j + M_i^{first_{i,j}}(t) - Smax_j^{first_{i,j}}$, $k_j \in \mathbb{N}$.

Proof: See [4]. ■

Worst Case Response Time Computation Algorithm. To compute the worst case end-to-end response time of any flow τ_i when Assumption 1 is met, we apply the following algorithm: (i) we first determine the set $\mathcal{S}_i$ of flows in $hp_i \cup sp_i$ crossing directly or indirectly flow τ_i, (ii) we then initialize for the iteration $q = 1$ the value of $Smax_j^{first_{k,j}}(q)$ for any flow $\tau_k \in \mathcal{S}_i$ and for any flow τ_j crossing τ_k: we have $Smax_j^{first_{k,j}}(1) = \sum_{h = first_j}^{pre_j(first_{k,j})} (C_j^h + Lmax)$, (iii) we proceed iteratively:

$q = 0$
Repeat
  q=q+1
  for any flow $\tau_k \in \mathcal{S}_i$
    for $h = first_k$ to $last_k$
      if $(h = last_k)$ or $(\exists \tau_j$ crossing τ_k such that $h = last_{j,k}$ or $h = pre_k(first_{k,j}))$ then
        compute $W_k^h(t)$ using $Smax_k^h(q)$ for any flow $\tau_k \in \mathcal{S}_i$
        if $\exists j$ such that $h = pre_k(first_{k,j})$ then
          $Smax_k^{first_{k,j}}(q+1) = max_t(W_k^h(t) - t) + C_k^h + Lmax$
        if $h = last_k$ then compute $R_k = max_t(W_k^h(t) - t) + C_k^h$
Until $(\exists \tau_k \in \mathcal{S}_i,\, R_k > D_k)$
  or $(\forall \tau_k \in \mathcal{S}_i, \forall h = pre_k(first_{k,j}),\, Smax_k^{first_{k,j}}(q+1) = Smax_k^{first_{k,j}}(q))$

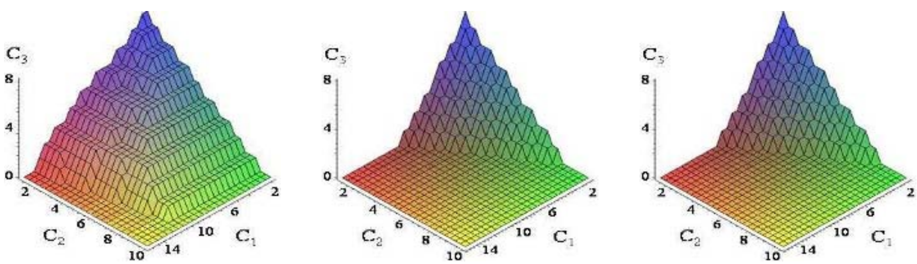
5 Comparison Between Holistic and Trajectory Approaches

In order to highlight the benefit of the trajectory approach, we compare the schedulability regions provided by both approaches in three configurations: rake, reverse path and same path. More precisely, we consider three sporadic flows τ_1, τ_2 and τ_3 with the following parameters: • τ_1: $T_1 = 50$, $D_1 = 100$, $P_1 = 2$, • τ_2: $T_2 = 10$, $D_2 = 20$, $P_2 = 1$, • τ_3: $T_3 = 10$, $D_3 = 45$, $P_3 = 1$. Each flow τ_i follows a specific path $\mathcal{P}_i$ defined in the configuration considered and the scheduling is assumed to be FP/FIFO* on any node. All flow parameters are fixed except the maximum processing time on a node visited. Moreover, the jitter of any flow is assumed to be null. The schedulability region is determined by the highest values of the maximum flow processing times beyond which the schedulability condition is not met: a flow does not meet its deadline. Figures 2 and 3 represent the schedulabity region obtained respectively by the holistic and the trajectory approaches in three configurations:

Configuration 1: Rake. Flow τ_3 visits nodes 1 and 2, whereas flow τ_1 visits node 1 and flow τ_2 visits node 2. The trajectory approach provides a bit larger schedulability region than the holistic one (see Figures 2.a and 3.a). However, the gain is limited as there exists no sequence of nodes visited by several flows.

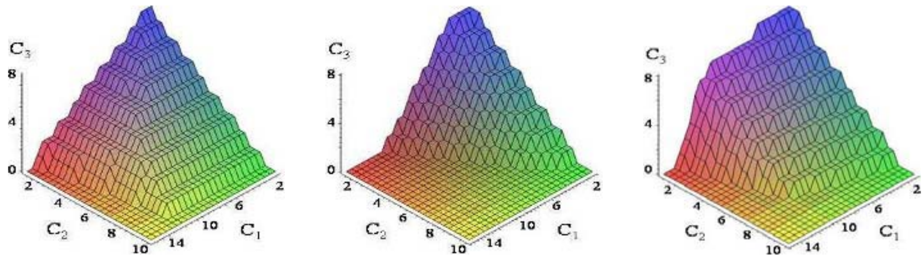
Configuration 2: Reverse Path. Flow τ_3 visits two nodes, also visited by τ_1 and τ_2 in the reverse direction. The schedulability region provided by the trajectory approach is the largest one (see Figures 2.b and 3.b). More precisely, as τ_1 and τ_2 visit the same sequence of nodes, the trajectory approach provides better results for these two flows. An admission control based on the trajectory approach will then accept more flows than based on the holistic approach.

Configuration 3: Same Path. Flows τ_1, τ_2 and τ_3 visit the same two nodes and in the same order. In this case, the benefit provided by the trajectory approach is very important (see Figures 2.c and 3.c). The trajectory approach does not account for impossible worst case scenarios, unlike the holistic approach.



(a) Rake (b) Reverse path (c) Same path

Fig. 2. Schedulability regions obtained with the holistic approach



(a) Rake (b) Reverse path (c) Same path

Fig. 3. Schedulability regions obtained with the trajectory approach

6 Conclusion

In this paper, we have shown how to obtain new results for non-preemptive Fixed Priority scheduling in the distributed case, assuming that flow packets sharing the same fixed priority are scheduled according to their dynamic priorities assigned on the first node visited. Examples of such a scheduling are FP/FIFO* and FP/EDF*. We have then recalled the principles of the holisitic approach, assuming the worst case scenario on any node visited. We have presented the trajectory approach taking into account the worst case scenario experienced by a flow packet on its trajectory. We have compared the end-to-end bounds given by both approaches in different configurations. We have identified the reasons of the holistic approach pessimism and illustrated the advantages of the trajectory approach, leading to a larger schedulability region.

References

1. K. Tindell, A. Burns, A. J. Wellings, *Analysis of hard real-time communications*, Real-Time Systems, Vol. 9, pp. 147-171, 1995.
2. J. Liu, *Real-time systems*, Prentice Hall, New Jersey, 2000.
3. S. Baruah, R. Howell, L. Rosier, *Algorithms and complexity concerning the preemptive scheduling of periodic real-time tasks on one processor*, Real-Time Systems, 2, p 301-324, 1990.
4. S. Martin, P. Minet, *Worst case end-to-end response times for non-preemptive FP/DP* scheduling*, INRIA Research Report, No 5418, December 2004.
5. K. Tindell, J. Clark, *Holistic schedulability analysis for distributed hard real-time systems*, Microprocessors and Microprogramming, Euromicro Jal, Vol. 40, 1994.
6. M. Spuri, *Holistic analysis for deadline scheduled real-time distributed systems*, INRIA Research Report No 2873, April 1996.
7. J. Le Boudec, P. Thiran, *Network calculus: A theory of deterministic queuing systems for the Internet*, Springer Verlag, LNCS 2050, September 2003.
8. A. Parekh, R. Gallager, *A generalized processor sharing approach to flow control in integrated services networks*, IEEE ACM Transactions on Networking, 1994.

Evaluating Evolutionary IP-Based Transport Services on a Dark Fiber Large-Scale Network Testbed

Francesco Palmieri

Università "Federico II" di Napoli, Centro Servizi Didattico Scientifico,
V. Cinthia, 45, 80126 Napoli, Italy
fpalmieri@unina.it

Abstract. Though once a research platform, today's Internet cannot serve as a testbed for direct experimentation due to its distributed ownership and its driving business functions. An alternative vehicle is needed to enable researchers to investigate new network architectures, services and functionalities. By deploying an highly over-provisioned metropolitan fiber ring and placing on it, at strategic locations, a collection of flexible MPLS-enabled network nodes, shareable in a seamless way for production and research, we created an overlay network testbed that can be used for testing in a real environment all the new networking technologies and services based on MPLS prior to their introduction in production environment. We describe the most significant experiences gathered from implementing the above technologies and the lessons learned in a form intended to assist others in realizing evolutionary services in production networks and organizing successful high-performance network testbeds.

1 Introduction

Telecommunication networks are evolving, especially at the backbone level, towards a deeper integration between the two most promising networking technologies: IP and Optical. The main drivers of this evolution are the continuous growth of bandwidth requests, the promise of cost improvements and, finally, the possibility of increasing revenues by offering new advanced services. In this context, some key aspects to be investigated are both the architecture and control plane services, in terms of transport and forwarding technologies and management, control and resilience capabilities of multi-layer geographically integrated networks. Though once a research platform, today's Internet cannot serve as a testbed for direct experimentation due to its technological heterogeneity and distributed ownership and its driving business functions. The same arguments hold for most of the available carrier or enterprise-owned high speed networks. Thus, an alternative vehicle is needed to enable researchers to investigate new network architectures, services and functionalities - especially to explore those dimensions beyond simply improving the speed performance and to carry out potentially service-disrupting experiments. Ideally, such networking research platform would consist of a sufficient number of dark fibers or very high speed communication links, geographically distributed to significantly test networking technologies on medium to long distance links, connecting evolutionary network nodes as well as end

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 306–317, 2005.

nodes, and supporting appropriate set of flexible and programmable resources that can be shared and controlled by the network researchers and production staff. Of course, when building such a network testbed, the most critical element to be taken into account is the availability and very high cost of the geographical links. Fortunately, the availability of optical fiber infrastructure that is currently in place but not being used is starting to change the scenario, and many competitive carriers and utilities are starting to sell dark fiber at reasonable prices. Until quite recently, however, most regional/metro network owners could not take advantage of this fiber, at least for their research purposes, for distances longer than 10 km. Transport terminals with expensive long-reach lasers, intermediate amplifiers and signal regenerators often structured in complex hierarchies (i.e. SDH/SONET) were necessary to provide the reach and reliability required. Today, however, many equipment manufacturers are including affordable long-range high performance interfaces (i.e. POS STM-16 and Giga Ethernet) in their routers or switches. With these devices, dark fiber owners can easily reach up to 100 km at multiple giga-speed on a fiber strand without repeaters. In this complex but amazing scenario we realized, since one year ago in Napoli, a dark fiber ring, covering the whole metropolitan area interested by academic and research institutions, to offer very high speed connections and Internet services to the whole research community located in the area. By placing, at strategic locations, a collection of flexible and evolutionary network nodes, shareable in a seamless way for production and research, and connecting them with a mesh of independent fiber links we created an overlay network testbed that can be used for testing in a real environment all the new networking technologies and services, prior to their introduction in the production environment, without necessitating protocol or policy changes in the underlying infrastructure, or requiring to buy and then dismiss new costly dedicated links. Such services and technologies, can be tested and evaluated for a long time without service disruption and then introduced on the production network only when they demonstrate sufficient stability guarantees. Accordingly, an experimental testbed has been implemented to test and evaluate the most interesting and evolutionary networking technologies based on MPLS transport (Traffic Engineering, Fast Reroute, DiffServ QoS etc.). This paper focus on describing the above testbed, realized by "Federico II" University on its next generation high speed production network, with the objective of assessing advanced networking functionalities in fiber-powered metropolitan and wide area networks. We describe the most significant experiences gathered from implementing the above technologies and the lessons learned in a form intended to assist others in realizing evolutionary services in production networks and organizing successful high-performance network testbeds.

2 The Testbed Architecture

In this section we describe in detail the architectural building blocks of the realized testbed to clarify, where necessary, the motivation of each design choice and explain the wide range of potentialities offered by such a testing and research infrastructure.

2.1 The Physical Layer

The physical fiber ring, on which our network is based, is approximately 40km long, consists of 156 9/125 G.652 single-mode fibers, contained in a loose tube glass-yarn Krone cable, connecting, in a differentiated two-way ring shape, some high performance optical switches/routers by Cisco Systems Inc., which realize the main transport infrastructure, or backbone that serves more than 20 level-2 distribution sites. The physical ring layout is reported in the fig. 1.

Fig. 1. Physical ring layout

Only two pairs of fibers are used for production traffic and the most part of the remaining ones are available for testing and research purposes. Several fiber pairs in the ring may be cascaded to realize longer concentric rings (with lengths multiple of the original ring one) to experimentally evaluate the effects of distance on the available optical transmission technologies.

2.2 Link Layer Technologies

The metro and wide area network testbeds in the past have been traditionally built on a circuit-based link layer service built with TDM technologies. Over the past decade, they have been enhanced by new link-level technologies including SONET/SDH and ATM and often structured in rings for better resiliency. While ATM has been widely deployed essentially as an overlay engineering plane for SONET/SDH, its use has not grown the way many predicted. This is due primarily to the protocol overhead associated with SONET/SDH and certain technical features of ATM that have not been as widely embraced by the marketplace as expected. In the last years the network has been transforming to a pure converged network offering high-speed broadband services over an optimized IP and Optical infrastructure. These modern networks are moving away from traditional circuit-switched networking technologies, towards packet-switched networking technologies, which are purpose-built for data, voice and video convergence. The advent of technologies including Gigabit Ethernet, Packet Over Sonet/SDH, IP and MPLS are combining to usher in a new era of metro and wide area networks. This motivates our choice to use these technologies, combined together to realize all the link layer connectivity in our testbed.

2.3 The Network Nodes

The network backbone is built on a fully meshed MPLS core realized between three high performance Cisco routers (a 12410 GSR and two 7606 OSRs – respectively LSR and LER1 and LER2 in fig. 2 below - when they were still on test bench), each acting as an access aggregation point (or POP) in the metropolitan area. Two high-end MPLS-capable routers/switches (Cisco Catalyst 6509 and 6006 – ER1 and ER2 in fig. 2) with the role of leaf access nodes connect each to two distinct POPs and acting, as needed, as Label Edge Routers, or simple leaf access nodes. We realized two distinct independent rings between the core nodes using both the primary and secondary branches on the ring. The links belonging to the primary ring are made on POS STM-16 (2.5 Gbps) interfaces and the links belonging to the secondary (or backup) ring and with the leaf access nodes are built on Gigabit Ethernet interfaces. All the connections between the routers are made with single mode optical fiber between long-range interfaces, STM-16 long reach (70Km) and Giga 1000baseLX/LH (10Km). Both the IS-IS and OSPF routing protocols, extended with traffic engineering facilities, can be used as the IGP of choice for the propagation of link status and resource availability information in the whole MPLS domain. Multiprotocol BGP (typically fully meshed IBGP sessions in the core) is used to carry VPN information through extended address families when the MPLS L2 or L3 VPN are used for testing and performance evaluation.

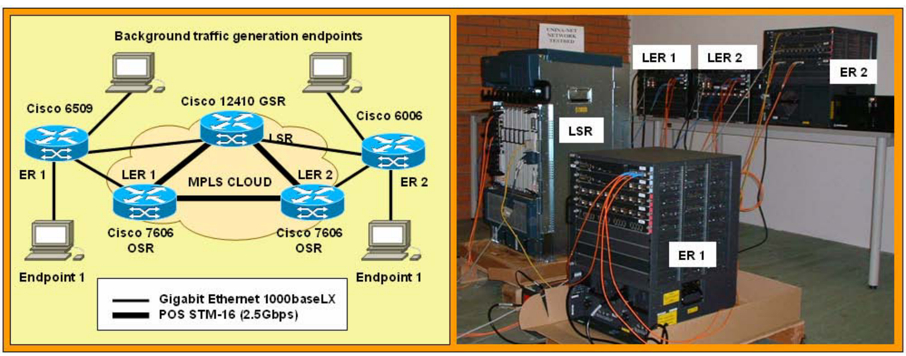


Fig. 2. The testbed architecture (with all network nodes together on test bench)

2.4 The Traffic Generation and Performance Analysis Applications

In order to assess the performance of the advanced functionalities that can be developed in such an evolutionary network, a proper mix of typical and critical network applications have been selected and integrated as sample traffic patterns in the testbed structure. A selection of the most common traffic streams, artificially simulated using the Chariot tool from NetIQ, starting from a first endpoint station, directly wired via Gigabit Ethernet to the first leaf/edge access node, traverses the core, reaching the other endpoint station wired in the same manner to the other leaf router. The great bandwidth availability on the whole testbed, starting from the testing endpoint perfectly reproduces the conditions of the next generation optical Internet. Furthermore, a consistent background aggregation of TCP flows, generated via multiple concurrent streams flowing between different Chariot endpoints (Background traffic generator endpoints, as in fig. 2), independently connected via Giga Ethernet to ER1 and ER2, simulates the effect of real Internet background traffic, resulting in heavy usage of the link. All the endpoints are built on HP Proliant DL-series Intel-based multiprocessor servers running the FreeBSD operating system and the Chariot endpoint software.

3 Network Testing Experience

An high performance fiber network testbed allows the assessment of a wide range of evolutionary network functionalities. In this section we present the most interesting and significant testing and performance evaluation experiences we have done and the lessons learned by their testbed implementation and observed results.

3.1 Routing Protocol Convergence

Link or node failures in an IP backbone cause packet losses until the network has reconverged around the failed link or node. These packet losses directly impact the network availability that can be committed in modern networks to support real-time and mission critical services. The time taken for an IP network to reconverge is dependent upon the size of the network, the interior gateway routing protocol (IGP) used and its specific configuration. For high availability targets to be offered, it is important that the routing protocol is tuned for rapid convergence. A key component of tuning IGP convergence is the tuning of the timers, which determine how frequently the main routing protocol events can occur. Historically, this resulted in a trade-off between rapid convergence and increased routing protocol stability: short timers lead to rapid convergence but with more potential for instability, where longer timers result in increased stability but slower convergence. The pragmatic result of this trade-off was that routing protocol timers were generally set conservatively and IP network convergence was typically a few tens of seconds. Experience gained from large-scale service provider deployment, however, indicated that such IGP implementations were very stable and hence that more emphasis could be placed on faster convergence. Further, recent developments to IS-IS and OSPF link state IP IGPs have focused on combining the best of both worlds leading to significant reductions in the convergence that can be achieved whilst still maintaining stability. It is obvious that a well-designed network testbed can be very useful in IGP timer tuning practice. Furthermore, whereas previously IGP timers were static and long, now with the introduction of dynamic timers they can adapt their responsiveness depending upon the stability of the network. This allows IGPs to be tuned such that when the network is stable, their timers will be short and they will react within a few milliseconds to any network topology changes. In times of network instability (e.g. caused by a flapping link), however, the IGP timers will increase in order to throttle the rate of response to network events. This scheme ensures fast convergence when the network is stable and moderate routing protocol overhead when the network is unstable. In addition to the advancements in the tuning of routing protocol timers, a number best practices for IGP design aim to reduce the number of routes carried in the IGP, significantly reducing IGP routing table computation times and hence resulting in faster IGP convergence. The combination of these optimizations techniques has resulted on our testbed in a reduction of IGP convergence times from approximately 10s seconds to 1-to-2 seconds being pragmatically achievable today. Further, with additional tuning and enhancements, sub-second IGP convergence may become a realistic possibility.

3.2 MPLS Traffic Engineering

In conventional IP networks routing protocols such as OSPF and IS-IS forward IP packets on the shortest cost path to the destination IP address of each IP packet. The computation of the shortest cost path is based upon a simple additive metric, where each link has an applied metric, and the cost for a path is the sum of the link metrics in the path. Availability of network resources, such as bandwidth, is not taken into account and, consequently, traffic can aggregate on the shortest path, consequently potentially causing links on the shortest path to be congested while links on alternative paths are under-utilized. This property of conventional IP routing protocols, of traffic aggregation on the shortest path, can cause sub optimal use of network resources, and can consequently impact the overall quality of service that can be offered (or require more network capacity than is optimally required). MPLS Traffic Engineering (TE) [1] uses the implicit MPLS characteristic of separation between the data plane (also known as the forwarding plane) and control plane to allow routing decisions to be made on criteria other than the destination IP address in the IP header, such as available link bandwidth. MPLS TE effectively provides an explicit routing capability at Layer 3, allowing paths to be used other than the shortest cost path to a destination, thereby avoiding traffic aggregation on the shortest path and providing more optimal use of available bandwidth. MPLS TE uses the following mechanisms:

- Information on available network resources, including a pool of available bandwidth maintained per link, are flooded by means of extensions to link-state based IP routing protocols such as IS-IS [2] and OSPF [3]
- A constraint-based routing (CBR) algorithm is used to compute the traffic path based upon a fit between the available network resources (advertised via IS-IS or OSPF) and the resources required, i.e. a requested amount of bandwidth
- The RSVP Protocol [4], with enhancements for MPLS TE [5], is used to signal and maintain an explicit route (termed a “traffic engineered tunnel”), from *head-end* to *tail-end*, in the form of an MPLS Label Switched Path (LSP). This LSP

follows the path determined by the constrain-based routing algorithm. In signaling the tunnel, admission control is performed at every hop.
– Traffic routed onto these LSPs or tunnels will then follow the traffic engineered explicit route to the destination, rather than the conventional IGP shortest path.

The following conditions can all be drivers for the deployment of MPLS TE [6]:

– *Network asymmetry.* Asymmetrical network topologies can often lead to traffic being aggregated on the shortest path whilst other viable paths are under-utilized. Network designers will often try to ensure that networks are symmetrical such that where parallel paths exist, they are of equal cost and hence the load can be balanced across them using conventional IGPs. Ensuring network symmetry, however, is not always possible due to economic or topological constraints. TE offers obvious benefits in these cases.
– *Unexpected demand.* In the presence of unexpected traffic demand (e.g. due to some new popular content), there may not be enough capacity on the shortest path (or paths) to satisfy the demand. There may be capacity available on non-shortest paths, however, and hence traffic engineering can provide benefit.
– *Long bandwidth lead-times.* There may be instances when new traffic demands are expected and new capacity is required to satisfy the demand, but is not available in suitable timescales. In these cases, traffic engineering can be used to make use of available bandwidth on non-shortest path links.

The use of TE gives flexibility in managing backbone bandwidth in order to achieve maximum service quality. The more effective use of bandwidth potentially allows higher service availability targets to be offered with the existing backbone bandwidth. Alternatively, it offers the potential of achieving the existing service availability targets with less backbone bandwidth. MPLS TE has been extensively tested on our testbed on a significant range of network topologies. Link utilization and end-to-end latency have been observed to evaluate the dynamic load-distribution and congestion adaptation capabilities of the backbone. We observed in almost all the test cases fair balancing in link utilization that clearly demonstrates the capacity of the technology to migrate load under demand from congested links to less congested links, as shown in the following figure.

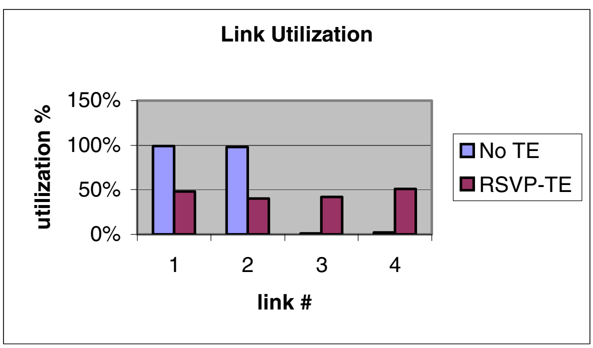


Fig. 3. Traffic balance between LSR1 and LSR2 with and without RSVP-TE

Furthermore, the significant improvements in end-to-end latency observed under heavy loads, demonstrated the effects of a more efficient network resource allocation.

3.3 DiffServ and MPLS TE

MPLS TE and Diffserv can be deployed concurrently in an IP backbone, with TE determining the path that traffic takes based upon aggregate bandwidth constraints, and Diffserv being used on each link for differential scheduling of packets on a per class basis. Whilst TE and Diffserv are orthogonal technologies they can be used in concert for combined benefit: TE allows distribution of traffic on non-shortest paths for more efficient use of available bandwidth, whilst Diffserv allows over/under-provisioning ratios to be determined on a per class basis. MPLS TE, however, computes tunnel paths for aggregates across all traffic classes and traffic from different classes may use the same TE tunnels. MPLS TE is aware of only a single aggregate *global pool* of available bandwidth per link and is unaware of what specific link bandwidth resources are allocated to which queues, and hence to which class. Consequently, MPLS TE has been extended with Diffserv-aware traffic engineering (DS-TE) [7], which introduces the concept of an additional and more restrictive pool of available bandwidth on every link. This more restrictive bandwidth pool is termed the *sub-pool*, while the regular TE bandwidth is called the *global pool* (the sub-pool is a portion of the global pool). The sub-pool may be used for constraint-based routing and admission control of tunnels for “guaranteed” or EF class traffic and the global pool used for regular (non-guaranteed) traffic. In supporting DS-TE, extensions have been added to IS-IS and OSPF to advertise the available sub-pool bandwidth per link as well as the available global-pool bandwidth. In addition, the TE constraint-based routing algorithms have been enhanced for DS-TE in order to take into account the constraint of available sub-pool bandwidth in computing the path of sub pool tunnels. RSVP has also been extended to indicate if it is signaling a sub-pool or global-pool tunnel. It is understood that setting an upper bound on the EF class (e.g. VoIP) effective utilization per link allows a way to restrict the effects of delay and jitter due to accumulated burst. DS-TE can be used to ensure that this upper bound isn’t exceeded. For evaluation purposes, we deployed DS-TE in our network testbed of fig. 2 to ensure that traffic is routed over the network so that, on every link, there will be never more than an assigned percentage (we choose 25% in our test) of the link capacity for EF class traffic, whilst there can be up to 100% of the link capacity for EF and AF class traffic in total. Each traffic endpoints is connected to its access node at 1Gbps speed. Endpoint1 send an aggregate of 400Mbps of traffic to Endpoint 2, and the background endpoints also send an equivalent aggregate of 400Mbps of traffic to each other. Both the IGP and non-Diffserv TE would pick the same route. The IGP would pick the top route (ER1→LSR→ER2) because it is the shortest path, whilst TE would pick the same path because it is the shortest path that has sufficient bandwidth available (1Gbps available, 800Mbps required). The decision to route both traffic aggregates via the top path may not seem appropriate if we examine the composition of the aggregate traffic flows. If each of the flows is comprised of 200Mbps of VoIP and 200Mbps of background data traffic then such routing decision would aggregate

400Mbps of VoIP traffic on the ER1→LSR→ER2 links, thereby exceeding our EF class bound of 25%. DS-TE can be used to overcome this problem: each link is configured with an available global pool bandwidth of 1Gbps, and an available sub pool bandwidth of 250Mbps (25%). A global-pool tunnel of 200Mbps is then configured from ER1 to ER2 for background data traffic, and a subpool tunnel of 200Mbps for VoIP traffic. Similarly, from the background traffic generators a global-pool tunnel of 200Mbps is configured for Business data traffic, and a subpool tunnel of 200Mbps for VoIP traffic. The DS-TE constraint based routing algorithm would then route the sub pool tunnels to ensure that the 250Mbps bound is not exceeded on any link, and of the tunnels from ER1 to ER2, one subpool tunnel would be routed via the top path (ER1→LSR→ER2) and the other via the bottom path (ER1→LER1→LER2→ER2). This simple test evidences how DS-TE can perform separate route computation and admission control for different classes of traffic. This enables the distribution of EF and AF class load over all available EF and AF class capacity making optimal use of available capacity. It also provides a tool for constraining the EF class utilization per link to a specified maximum thus providing a mechanism to help bound the delay and jitter. In order to provide these benefits, however, the configured bandwidth for the sub-pool and global pool must represent queuing resources, which are only available for traffic-engineered traffic, and hence non-traffic engineered traffic should be queued separately on each link. Our tests also demonstrated that DS-TE is a very effective technique to strictly limit the latency and jitter on the EF class giving it absolute priority and full bandwidth required but it is also useful to ensure proper service characteristics and fair bandwidth distribution between the AF classes configured.

3.4 MPLS Traffic Engineering Fast Reroute

In the previous section, when talking about IGP convergence, we highlighted that link or node failures in an IP backbone can significantly impact the availability that can be committed in modern multiservice networks empowering real-time services. Whilst sub second convergence for IP routing protocols is a realistic prospect, it is expected that IGP convergence will not be able to match the capabilities of SDH/SONET networks, which use the capabilities of Multiplexer Section Protection (MSP) and Automatic Protection Switching (APS) respectively to recover around failures in tens of milliseconds. This is because the functions are performed in fundamentally different ways: IGP convergence is based on a distributed computation, whereas SDH/SONET restoration is based upon local detection and pre-computed local protection around the failure. MPLS TE Fast Reroute (FRR) extends the concepts of local failure detection and protection to MPLS TE in order to provide very rapid recovery around failures (e.g. a few tens of milliseconds) prior to any distributed convergence/re-optimization. Without FRR, under failure conditions, the head-end of a TE tunnel determines a new route for the tunnel LSP but due to messaging and convergence delays, it cannot recover as fast as is possible. On the other hand local nature of FRR allows very rapid protection and restoration around failures over pre-determined backup paths. For SDH links, detecting the failure of a link is typically done in less than 10ms, and with FRR, many hundreds of protected tunnel-LSPs can be switched around the failure in less than 50ms. This is equivalent to the level of protection provided by MSP and APS and in SDH and SONET networks respectively. FRR is designed for backbone deployment where the number of network components is typically relatively low, but where the failure of those components can have severe impacts on services. The determination of optimal routing for FRR backup tunnels in different failure scenarios is, however, a complex problem and needs to take into account factors including the available bandwidth on potential backup paths, tunnel inter relationships and interdependencies on the lower layer network topologies. This subject is currently the focus of proper testing and tuning activity. To implement FRR in our testbed it was necessary to upgrade the IOS version of all the OSR catalyst nodes to latest 12.2(18)SXD version that was the first release supporting FRR on the above machines.

3.4.1 FRR Link Protection

Link protection is provided by backup tunnels that bypass the protected link and terminate at the Next Hop of the LSP's path. These backup tunnels reroute the LSP's traffic to the next hop as soon as a link failure is detected. To investigate the MPLS-TE FRR functionality on our testbed we used the usual physical connection layout as depicted in fig. 2. There, four unidirectional LSPs were set up, from LER1 to LER2; two of them (LSP1 and LSP3) were fast reroutable while the other two (LSP2 and LSP4) did not have this capability enabled. For the two FRR protected LSPs, the pre-provisioned backup LSP followed the path LER1-LSR-LER2. For the remaining LSP the new path should be automatically determined by the MPLS TE dynamic path auto-discovery facility and typically should follow the same path. Two Chariot endpoints were connected to ER1 and ER2 in order to generate data traffic and to count the number of lost packets to simplify the estimation of the interruption caused by the fault, the generator was set up to originate 1000 packets per second for each LSP, so that the number of packets lost was an estimate of the interruption time in milliseconds. Figure 4 below reports the obtained results. The fault was created pulling out the fiber corresponding to the transmission connector of the interface on LER1 towards LER2, i.e. the fiber carrying traffic for all four LSPs. FRR (activated for LSP1 and LSP3) is considerably faster than normal LSP reroute (invoked for LSP2 and LSP4), due to the local processing, fast error detection and availability of pre-established backup paths. It is important to note that, in the context of an operational network with a large number of nodes and higher propagation delay, rerouting time difference between LSP reroute and MPLS FRR could be even more significant and the measurements reported in fig 4 may be considered nearly as lower bounds. It is also important to understand that the FRR protection times are dependent on the number of LSPs that must be protected. Again, when an higher number of LSPs are rerouted in case of fault, the measured values would be expected to be higher.

3.4.2. FRR Node Protection

Node protection relies on backup tunnels that bypass Next HOP nodes along LSPs and terminate at the so-called Next Next HOP (NNHOP). These backup tunnels protect the bypassed node and reroute the LSPs' traffic to the NNHOP as soon as a node failure is detected. They also provide link protection because they recover from failures that may occur on link up to the protected node. Basically, to test FRR Node protection we slightly modified the same configuration already used for the assessment of link protection, where the LSPs are configured to follow the path ER1-LSR-ER2 and the pre-provisioned backup tunnel for fast-reroutable LSPs is between ER1-LER1-LER2-ER2. Fig. 4 below reports the average number of lost packets measured in case of a power shutdown on GSR that properly simulated the node failure. Note that most of the convergence time is due to the node failure detection time. In case of a node failure generating a link failure, similar results would have been found to the link failure scenario. Other way of simulating the node failure, like shutting down (via the CLI interface) the node or the interface connected to GSR resulted in no packet loss, because, in that case, the MPLS control plane signals the necessity of rerouting the LSPs before performing the operation.

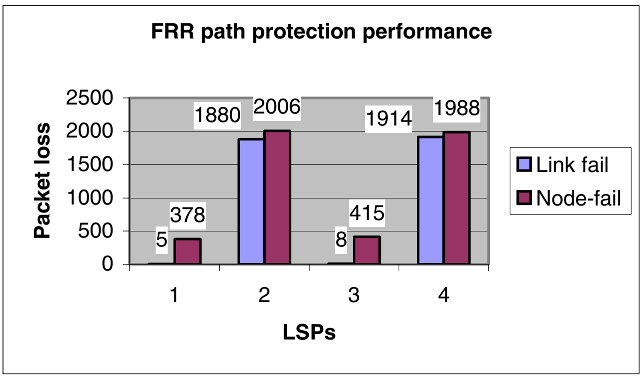


Fig. 4. FRR path protection performance

FRR, as expected, achieve tremendous improvements in link fault recovery time, keeping it lower than 25 msec in average, but this feature is usually available only on high-end router platforms and very high speed interfaces. This time is far better then classic SDH guaranteed protection mechanism (i.e. MSP, with less than 50 msec) but it should be taken into account that in our testbed, with a maximum connection length of 25Km there isn't a significant latency.

4 Conclusions

Our work shows that we can experimentally asses on a realistic real-world testbed how network convergence may be improved and emerging technologies such as MPLS TE combined with DiffServ, constraint-based routing and FRR can form a simple, scalable and efficient networking model capable of providing rapid, cost-effective and physical layer independent mechanisms for enhancing network resilience and reliability and supporting the QoS requirements needed by new real-time applications. Of course, the experimental results obtained in the presented test beds are an interesting but not exhaustive step in the demonstration of feasibility of advanced functionality for next generation IP over optical networks. Further investigations are required to assess the complete integration of all the different functionality in a multi-layer, multi-vendor and multi-domain environment.

References

1. Awduche, D. O. et Al.: *Requirements for TE Over MPLS*, IETF RFC 2702, 1999.
2. Smit, H., Li, T.: *ISIS Extensions for TE*, Internet Draft < draft-isis-traffic-traffic-02>
3. Katz, D. et Al: *Traffic Engineering Extensions to OSPFv2*, IETF RFC 3630, 2003
4. Braden, R. et Al: *RSVP version 1 Functional Specification*, IETF RFC 2205, 1997.
5. Awduche, D. O.: *RSVP-TE: Extensions to RSVP for LSP Tunnels*, IETF RFC 3209, 2001
6. Telkamp, T.: *Hot Interconnect*, Stanford, 2001
7. Le Faucheur F.et Al, *Requirements for support of Diff-Serv-aware MPLS Traffic Engineering*, RFC 3564, 2003

Pareto Optimal Based Partition Framework for Two Additive Constrained Path Selection

Yanxing Zheng[1], Turgay Korkmaz[2], and Wenhua Dou[1]

[1] School of Computer Science,
National University of Defense Technology, P.R. China
yxzheng@nudt.edu.cn
[2] Department of Computer Science, University of Texas, San Antonio, USA
korkmaz@cs.utsa.edu

Abstract. One of the challenging issues in QoS routing (QoSR) is how to select a multi-constrained path (MCP) that can meet the QoS requirements. We consider the concept of Pareto optimality in the context of MCP problems and establish a Pareto optimal based partition framework (POPF). Based on the key concepts in POPF, we propose algorithm (DA_2CP) to deal with two additive QoS metric.

Keywords: QoS routing, Pareto optimal, multi-objective optimization.

1 Introduction

One challenging issue in QoSR is how to determine a path subject to multiple QoS requirements. Typical QoS metrics (e.g., delay, jitter, bandwidth, cost, reliability etc.) are often divided into three categories: additive, concave, and multiplicative. The great challenge that QoSR algorithms must face up to comes mainly from additive metrics [1]. Hence, we formulate the problem at hand as follows:

Definition 1 (Multi-constrained path (MCP) problem). *Consider a network $G(N, E)$, N is the set of nodes. E is the set of links between nodes. Each link (u, v) is specified by a k-dimensional link metric vector $w = (w_1, w_2, \ldots, w_k)$, where $w_i(u, v)$ is an additive QoS metric and $w_i(u, v) \geq 0, i = 1, 2, \ldots, k$. For routing request $C = (c_1, c_2, \ldots, c_k)$, the problem is to find a path p from source node s to destination node d such that*

$$w_i(p) \stackrel{def}{=} \sum_{(u,v)\in p} w_i(u, v) \leq c_i, i = 1, 2, \ldots, k \tag{1}$$

Later we also write path p as $p(w_1(p), w_2(p), \ldots, w_k(p))$ and use P_{sd} to denote the paths between s and d. If we seek an optimal solution at the same time, i.e., path p^* should satisfy $w_i(p^*) \leq w_i(p), i = 1, 2, \ldots, k$, Where p^* and p satisfy (1). Then the problem becomes a Multi-Constrained Multi-Objective Path (MCMOP) problem. When we deal with a specific MCP problem, we can view it

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 318–325, 2005.

as a MCMOP problem since the solutions for the MCMOP problem are of course the solutions for the original MCP problem. An important property considered necessary for any feasible candidate solution to a MCMOP problem is Pareto optimality. The significance of Pareto optimal paths is that if none of the Pareto optimal paths between s and d can satisfy routing request C, then there is no path in P_{sd} that can satisfy C. With this in mind, we establish a Pareto optimal based partition framework (POPF) and use it to deal with QoS routing under two additive constraints.

2 Related Works

Jaffe [2] proposes an early linear aggregation based QoSR algorithm, in which link metrics are aggregated linearly into a single one. Jaffe's algorithm searches only in one direction. For systematic search in several directions, researchers have considered Lagrangian-based linear composition algorithms (LLCA) [4][5][6] that are dynamically adjusting search directions. Depending on the nature of the underlying problem, existing Lagrangian-based algorithms uses different strategies for adjusting search directions. Because we are interested in identifying several Pareto optimal paths, our search strategy enhanced with new heuristics will be different than that of other algorithms using the same idea. Moreover, we improve the performance (success rate and response time) of the basic LLCA-based search along with precomputation and look-ahead futures. In [7], the authors also use Linear aggregation along with a pre-computation algorithm called MEFPA to address MCP problems. The computation cost and performance of MEFPA is directly relevant to parameter b. To express more clearly, we use MEFPA(b) instead of MEFPA in the rest of the paper.

3 Pareto Optimal Based Partition Framework

3.1 Basic Concepts

Definition 2 (QoS Metric Space (QoSMS)). *$W^k = W_1 \times W_2 \times, \ldots, \times W_k$ is called QoSMS, if $w_i(p) \in W_j, j \in \{1, 2, \ldots, k\}$ for any $p(w_1(p), w_2(p), \ldots, w_k(p)) \in G(N, L)$, where '$\times$' denotes the Cartesian product.*

Definition 3 (Path Mapping f). *Mapping f maps path $p(w_1(p), w_2(p), \ldots, w_k(p)) \in G(N, L)$ to point $(w_1(p), w_2(p), \ldots, w_k(p))$ in W^k:*

$$f(p(w_1(p), w_2(p), \ldots, w_k(p))) = (w_1(p), w_2(p), \ldots, w_k(p)) \quad (2)$$

There may be several paths that correspond to one same point in W^k. Later we use $f^{-1}(w_1(p), w_2(p), \ldots, w_k(p))$ to denote one of these paths.

Definition 4 (Dominance). *Vector $u = (u_1, u_2, \ldots, u_k) \in W^k$ dominates vector $v = (v_1, v_2, \ldots, v_k) \in W^k$, denoted by $u \prec v$, if and only if u is partially less than v, i.e., $\forall i \in \{1, 2, \ldots, k\}, u_i \leq v_i \wedge \exists i \in 1, 2, \ldots, k : u_i < v_i$.*

Definition 5 (Pareto Optimal Path). *$p(w_1, w_2, \ldots, w_k) \in P_{sd}$ is a Pareto optimal pathif and only if there is no path $p'(w'_1, w'_2, \ldots, w'_k) \in P_{sd}$, for which $(w'_1, w'_2, \ldots, w'_k) \prec (w_1, w_2, \ldots, w_k)$.*

Definition 6 (Pareto front PF^*). *For a given MCMOP problem, Pareto front $PF^* = \{p(w_1, w_2, \ldots, w_k) \in P_{sd} | p(w_1, w_2, \ldots, w_k) \text{is a Pareto optimal path}\}$.*

Each element in PF^* is called as a Pareto optimal point (POP). Points lying in the convex part of Pareto front are called as convex Pareto optimal points (CPOP). The point that has the least w_i is called as the i'th bound point of Pareto front.

3.2 Partitioning W^k by POPs

Definition 7. *Dominated set of vector $w(w_1, w_2, \ldots, w_k) \in W^k$*

$$D(w) \stackrel{def}{=} \{(w'_1, w'_2, \ldots, w'_k) \in W^k | (w_1, w_2, \ldots, w_k) \prec (w'_1, w'_2, \ldots, w'_k)\} \quad (3)$$

Definition 8. *Feasible area $A_{feasible}$: $A_{feasible} = \bigcup_{v \in PF^*} D(v) \cup PF^*$*

Corollary 1. *If routing request $(c_1, c_2, \ldots, c_k) \in A_{feasible}$, then the request can be satisfied, i.e., there exists a path from s to d that meets the routing request.*

We omit all proofs for claims. These proofs can be found in [8].

Definition 9. *Unfeasible Area $A_{unfeasible}$: $A_{unfeasible} = A_1 \cup A_2$*

where $A_1 = \{(w_1, w_2, \ldots, w_k) | w_i < w'_i, (w'_1, w'_2, \ldots, w'_k)$ is the $i'th$ bound point of Pareto front$\}$; $A_2 = \{(w_1, w_2, \ldots, w_k) | \exists (w'_1, w'_2, \ldots, w'_k) \in PF^*, (w_1, w_2, \ldots, w_k) \prec (w'_1, w'_2, \ldots, w'_k)\}$

Corollary 2. *If routing request $C = (c_1, c_2, \ldots, c_k) \in A_{unfeasible}$, then there is no path from s to d that meets the routing request C.*

Definition 10. *NP-complete Area A_{NPC} : $A_{NPC} = W^k \backslash (A_{feasible} \cup A_{unfeasible})$*

3.3 Generating Pareto Optimal Paths

The key to partition QoSMS is finding the elements in Pareto front.

Definition 11 (Linear path length function (LPLF)). *For path $p = n_1 \rightarrow n_2 \rightarrow, \ldots, n_m$, each link is associated with k additive constraints. Linear Path length function (LPLF) is defined as $w(p) = \sum_{i=1}^{k} \alpha_i w_i(p)$, where $\sum_{i=1}^{k} \alpha_i = 1$ and $w_i(p) = \sum_{j=1}^{m-1} w_i(n_j \rightarrow n_{j+1}), i \in 1, 2, \ldots, k$.*

Link metrics associated with each link $n_i \rightarrow n_{i+1}$ are combined by coefficient $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_k)$ linearly. Thus Dijkstra's algorithm can be used directly to return a least length path w.r.t. $w(p)$. Later we denote Dijkstra's algorithm using LPLF and search direction $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_k)$ by Dijkstra(α). Path found by Dijkstra(α) is denoted by $p(w_1^\alpha, w_2^\alpha, \ldots, w_k^\alpha)$.

Corollary 3. *For any search direction* $(\alpha_1, \alpha_2, \ldots, \alpha_k)$, $\sum_{i=1}^{k} \alpha_i = 1$, $p(w_1^\alpha, w_2^\alpha, \ldots, w_k^\alpha)$ *is Pareto Optimal.*

The following two corollaries give special characters when searching in W^2.

Corollary 4. *For two search directions* $\alpha = (\alpha_1, \alpha_2)$ *and* $\beta = (\beta_1, \beta_2)$, *where* $\alpha_1 + \alpha_2 = 1, \beta_1 + \beta_2 = 1$, *if* $\alpha_1 > \beta_1$, *then (a)* $w_1^\alpha \leq w_1^\beta$ *and (b)* $w_2^\alpha \geq w_2^\beta$

Corollary 5. *Consider two search directions* $\alpha = (\alpha_1, \alpha_2)$ *and* $\beta = (\beta_1, \beta_2)$, *where* $\alpha_1 + \alpha_2 = 1, \beta_1 + \beta_2 = 1$. *If* $p(w_1^\alpha, w_2^\alpha)$ *is the same path with* $p(w_1^\beta, w_2^\beta)$, *then for any search direction* $\gamma = (\gamma_1, \gamma_2)$, *where* $\alpha_1 > \gamma_1 > \beta_1$, *there holds that* $p(w_1^\gamma, w_2^\gamma)$ *is the same path with* $p(w_1^\alpha, w_2^\alpha)$ *and* $p(w_1^\beta, w_2^\beta)$.

4 DA_2CP Algorithm

Using POPF and the corollaries above, we propose an efficient QoSR algorithm DA_2CP to address the MCP problem under two constraints. Pseudo code of DA_2CP is given in Fig 1. DA_2CP includes two phases, namely precomputation

DA_2CP(G, s, d**,**$C(c_1, c_2)$**)**
G: network topology; s: source; d: destination; C: routing request
(1) The first phase: (p_1, p_2) = Pre_Compute (G, s, d)
(2) The second phase:
  (a) Handle-Pre-Compute($p_1, p_2, C(c_1, c_2)$)
  (b) If step (a) returns undecided, start repeat:
    1) Get path p_3 by searching in direction $(\alpha, 1-\alpha)$ perpendicular to p_1p_2
    2) If p_3 is one of p_1 or p_2 : End(Fail)
    3) If $p_3 \prec C$ or $p_3 = C$: Return p_3;// success
    4) If $(\alpha \cdot c_1 + (1-\alpha) \cdot c_2) < (\alpha \cdot w_1 + (1-\alpha) \cdot w_2)$: Return; //fail
    5)If $c_1 < w_1$, then $p_2 = p_3$; Else $p_1 = p_3$;
  (c) End repeat

(p_1, p_2)=**Pre_Compute**(G, s, d)
(1) p_1=Dijkstra(1,0); (2) p_2=Dijkstra(0,1); (3) Return(p_1, p_2)

Handle-Pre-Compute($p_1, p_2, C(c_1, c_2)$)
1) If $(c_1 < w_1(p_1))$: DA_2CP return; //Fail
2) If $p_1 \prec C$ or $p_1 = C$: DA_2CP return ; //Success
3) If(p_1 and p_2 are the same): DA_2CP return;//Fail
4) If($c_2 < w_2(p_2)$): DA_2CP return; //Fail
5) If $p_2 \prec C$ or $p_2 = C$: DA_2CP return p_2;//Success
6) Else Return undecided.

Fig. 1. DA_2CP algorithm and the procedures it uses

and on-demand computation. In the first phase, DA_2CP precomputes the bound points of the Pareto front. If the routing requests cannot be satisfied by the bound points, DA_2CP will start the second phase to compute paths on-demand. In the second phase, DA_2CP repeatedly searches for CPOPs until the termination condition is met.

Precomputation is very appealing due to its ability to improve response time and thus used in the first phase. During the second phase, to further improve the response time, DA_2CP uses more heuristic information (e.g., look-ahead ability, see step (2).b.4 of DA_2CP in Fig 1). The following claim justifies this look ahead.

Corollary 6. *Given that Dijkstra finds CPOP $p(w_1, w_2)$ in direction $\alpha = (\alpha_1, \alpha_2)$. For routing request $C = (c_1, c_2)$, if $\alpha_1 c_1 + \alpha_2 c_2 < \alpha_1 w_1 + \alpha_2 w_2$, there is no path that can meet C.*

5 Performance Evaluation of DA_2CP

In respective of two additive constrained QoSR problems, H_MCOP [9][10] is very efficient in both success rate and execution time. Yong Cui [7] argues that MEFPA(b) outperforms H_MCOP in most cases. So in the simulations, we compare DA_2CP with MEFPA(b) to verify the efficiency of our algorithm. Furthermore, another linear aggregation based algorithm DWCBLA [11] is also used as comparison. All the following simulations are based on Waxman model [12]. Each link in the randomly generated topology has two additive metrics w_1 and w_2, and each metric w_i ∼uniform[1,300]. The destination node is selected at least two hops away from the source node.

5.1 Absolute Computation Cost (ACC) and Performance (AP)

We use metrics that are independent of routing requests to evaluate the performance of DA_2CP. We first consider the average number of iteration of Dijkstra's algorithm, that each evaluated algorithm needs to find 'all' CPOPs, as the absolute computation cost. We also consider the size of NPC area given in Definition 10 to evaluate the absolute performance. The sizes of networks are 50, 100 and 200 respectively and with each node number, 1000 topologies are generated.

For DA_2CP, finding all CPOPs means that it runs until the termination conditions are met. Note that MEFPA(b) searches in only several fixed directions and is not guaranteed to find 'all' CPOPs. So it is not evaluated in this evaluation. Fig 2 shows the simulation results. We can see that to achieve the best performance, DA_2CP needs averagely less computation cost than DWCBLA.

Fig 3 shows the size of the NPC area (SoNA) with respect to the iteration times b of Dijkstra's shortest path algorithm. Because the absolute size does not make much sense, the sizes are further handled while the relative relationship between them is kept. We can see that DA_2CP converges more quickly than DWCBLA and MEFPA(b).

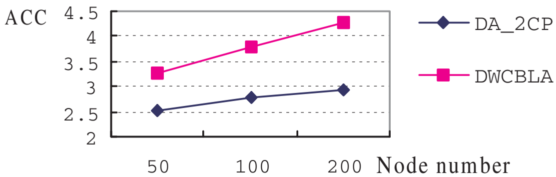


Fig. 2. Absolute computation cost comparison

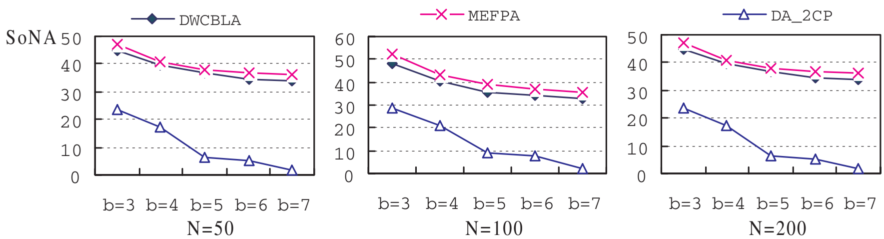


Fig. 3. Sizes of the NPC areas of various algorithms

5.2 Relative Computation Cost and Performance

Simulations in [7] show that MEFPA(7) outperforms H_MCOP [10] under two additive constraints. For DWCBLA, if only the length of the largest interval maintained is less than 1/(b-1), it can achieve the performance of MEFPA(b). We denote this changed algorithm by DWCBLA(1/(b-1)). For DA_2CP, if only $|\alpha_1 - \beta_1| \leq 1/(b-1)$ (suppose that p_1 is found in direction α_1, α_2 and p_2 is found in direction β_1, β_2. p_1 and p_2 take their meaning in Fig 1), it can achieve the performance of MEFPA(b). We denote this changed algorithm by DA_2CP($|\alpha_1 - \beta_1| \leq 1/(b-1)$). Thus by changing b, we can evaluate their relative computation cost when they achieve comparable performances. We generate routing requests (c_1, c_2) randomly for each pair (s, d), and c_1 ~uniform$[0.8 * w_1^{(1,0)}, 1.2 * w_1^{(0,1)}]$, c_2 ~uniform$[0.8 * w_2^{(0,1}, 1.2 * w_2^{(1,0)}]$. Thus the generated requests can cover the whole NPC area [11] and algorithms can be evaluated under the most critical conditions. We call routing requests generated in this way as critical routing requests (CRR).

Fig 4 shows the computation cost comparison when the three algorithms achieve the comparable performances of different b. Y-coordinate is the times of calling Dijkstra's algorithm (TCDA). We can see that to achieve comparable performance, DA_2CP has the least computation cost.

5.3 Evaluation of Response Time

A practical QoSR algorithm should have not only a high success rate, but also a quick speed of response. In this part, we look at how often routing requests can be responded immediately by DA_2CP. We count individually the precomputation success rate (PSR), the ratio of the number of routing requests satisfied by

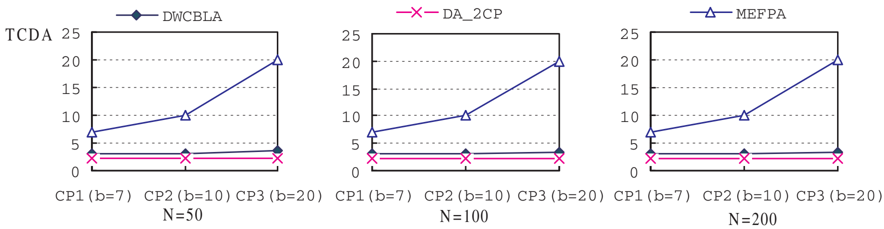


Fig. 4. Relative computation cost of the three algorithms

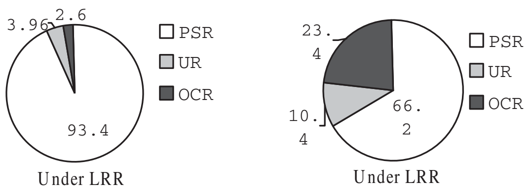


Fig. 5. Evaluation of response time

pre-computed paths to the number of routing requests generated, unfeasible rate (UR), the ratio of the times that on-demand computation is avoided to the number of total routing requests generated, and on-demand computation rate (OCR), the ratio of the number of routing requests that need on-demand computation to the number of total routing requests. It is obvious that the larger the sum of PSR and UR, the quicker DA_2CP responds. In addition to CRR, another kind of routing requests generated using the same method in [10] is also used. Routing requests generated in [10] are likely to be feasible, we call such routing requests as loose routing requests (LRR). Fig 5 shows the simulation results with node number N=200. We can see that when routing requests are loose, most routing requests can be responded immediately and only very small parts of routing requests need further on-demand computation. When routing requests are critical, again a considerable part of routing requests does not require further on-demand computation.

6 Conclusions

In this paper, we first propose Pareto optimal partition framework (POPF) of QoS metric space. The partition framework can be used as a guideline for designing QoSR algorithms. Lagrangian-based linear composition algorithm (LLCA), which is usually used to solve restricted shortest path problems, is improved with precomputation, look-ahead and new heuristic information to solve MCP problems. Simulations show that DA_2CP is efficient in both success rate and response time.

References

1. Zheng, Y., Dou, W., Tian, J., Xiao, M.: An overview of research on qos routing. In: Advanced Parallel Processing Technologies(APPT03), Springer LNCS(2834) (2003) 387–397
2. Jaffe, J.M.: Algorithms for finding paths with multiple constraints. Networks 14 (1984) 95–116
3. Korkmaz, T., Krunz, M., Tragoudas, S.: An efficient algorithm for finding a path subject to two additive constraints. Computer Communications Journal 25 (2002) 225–238
4. Handler, G.Y., Zang, I.: A dual algorithm for the constrained shortest path problem. Networks 10 (1980) 293–310
5. Blokh, D., Gutin, G.: An approximate algorithm for combinatorial optimization problems with two parameters. Australasian Journal of Combinatorics 14 (1996) 157–164
6. Juttner, A., Szviatovszki, B., Mecs, I., Rajko, Z.: Lagrange relaxation based method for the QoS routing problem. In: Proceedings of the INFOCOM 2001 Conference. Volume 2., IEEE (2001) 859–868
7. Cui, Y., Xu, K., Wu, J.: Precomputation for multi-constrained qos routing in high-speed networks. In: Proceedings of the INFOCOM 2003 Conference, San Francisco (2003)
8. Zheng, Y., Korkmaz, T., Dou, W.: Pareto optimal based partition framework for two additive constrained path selection. Technical report (2004) http://www.cs.utsa.edu/ korkmaz/yanxing/POPF.pdf.
9. Korkmaz, T., Krunz, M.: Multi-constrained optimal path selection. In: Proceedings of the INFOCOM 2001 Conference. Volume 2., Anchorage, Alaska, IEEE (2001) 834–843
10. Korkmaz, T., Krunz, M.: Routing multimedia traffic with QoS guarantees. IEEE Transactions on Multimedia 5 (2003) 429–443
11. Zheng, Y., Tian, J., Liu, Z., Dou, W.: An efficient dynamic weight coefficient qos routing algorithm. In: International Network Conference(INC04), UK (2004)
12. Waxman, B.: Performance evaluation of multipoint routing algorithms. In: Proceedings of the INFOCOM 93 Conference. Volume 3., IEEE (1993) 980–986

Joint Path Protection Scheme with Efficient RWA Algorithm in the Next Generation Internet Based on DWDM

Jin-Ho Hwang[1], Jae-Dong Lee[2], Jun-Won Lee[3], and Sung-Un Kim[1,4]

[1] Pukyong National University, 599-1 Daeyeon 3-Dong Nam-Gu, Busan, 608-737, Korea
jhhwang@mail1.pknu.ac.kr

[2] Kyungnam College of Information and Technology, Ju-Rye, 2-Dong, Sa-Sang Gu 167, Busan, 617-701 Korea
jdlee@kit.ac.kr

[3] Andong National University, 388 Song-chon Dong, Andong, Kyoungbuk 760-749, Korea,
leejw@andong.ac.kr

[4] Corresponding Author: kimsu@pknu.ac.kr

Abstract. In dense-wavelength division multiplexing (DWDM) networks, one of the critical issues is routing and wavelength assignment (RWA). And, guaranteeing network survivability is essential for sustaining traffic continuity even for network failures. In this paper, we propose the routing algorithm called survivability-guaranteed minimum interference path routing (SG-MIPR). And under SG-MIPR, we suggest a joint path search approach using shared risk link group (SRLG) information, while considering trap avoidance (TA) problem. The effectiveness of the proposed algorithm is verified through the simulation experiments.

1 Introduction

While coping with the rapid growth of IP and multimedia services, current Internet based on time division multiplexing (TDM) cannot supply sufficient transmission capacity for high bandwidth-needed services. However, the huge potential capacity of one single fiber, which is in Tb/s range, can be exploited by applying DWDM technology which transfers multiple data streams on multiple wavelengths simultaneously. So, DWDM-based optical networks have been a favorable approach for the next generation optical Internet (NGOI)[1].

The RWA problem is embossed as very important and plays a key role in improving the global efficiency for resource utilization in DWDM networks[2]. However, it is a combinational problem known to be NP-complete because routing problem and wavelength assignment problem are tightly linked together[3]. Since it was more difficult to work out RWA as a coupled problem, this problem has been approximately divided into two sub-problems: routing and wavelength

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 326–333, 2005.

assignment, and several RWA algorithms have been proposed in [4][5]. In previous studies, the routing scheme has been recognized as a more significant factor on the performance of the RWA than the wavelength assignment scheme[5].

Also, network survivability is a critical concern in network design. For example, a single failure will cause severe service disruptions. Therefore, SRLG has been proposed as a fundamental concept for fault management in layered networks (e.g., optical and IP/generalized multiprotocol label switching (GMPLS) over DWDM). And SRLG is exploited as a key constraint for route computation.

As the combination of routing problem and survivability, the route computation of a primary path and a backup path is generally based on the modified shortest path first (SPF) algorithm. This approach does not consider any optimization at all. It only attempts to find a best (lowest cost) path in each route computation. Moreover, this does not take into account TA problem. However, there is one possibility to perform a limited optimization. Although the global optimization for all calls is impossible, we can perform the individual optimization for the primary and backup path of each new call request.

In this paper, we propose the routing algorithm called SG-MIPR. And under SG-MIPR, we suggest a joint path search approach, while considering SRLG constraint and TA problem together. Finally, to verify the performance of the proposed algorithm, simulation experiments are carried out.

In the following sections, section 2 presents DWDM architecture and survivability requirements in DWDM networks. Then, section 3 illustrates SG-MIPR and joint path search approach, and performance is evaluated in section 4. Finally, some concluding remarks are made in section 5.

2 DWDM Architecture and Survivability Requirements

2.1 DWDM Architecture

Architecture of DWDM network is shown in figure 1, in which IP traffics are injected into DWDM ingress nodes from electronic domain based-networks. In this architecture, ingress nodes perform traffic aggregation and route optical data to egress nodes. And, at the egress node, the traffic is disaggregated and delivered to the destination network. Core DWDM nodes are interconnected with each other and perform forwarding of the optical data in the all-optical signal domain. An established lightpath between ingress and egress may cross a number of intermediate core nodes interconnected by fiber segments, optical amplifiers and optional taps. The optical components that constitute a core node, in general, include an optical switch, a demultiplexer comprising of signal splitters and optical filters, and a multiplexer made up of signal combiners. A core node may also contain a transmitter array, a receiver array and wavelength converters enabling wavelength conversion[6]. In this paper, we presume that core nodes are equipped with wavelength converters.

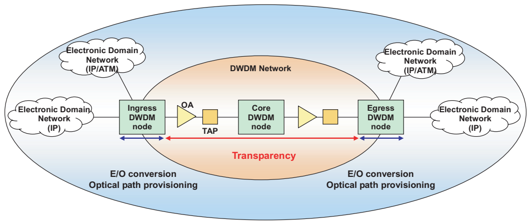


Fig. 1. An architectural model for DWDM networks

2.2 Survivability Requirements

Network survivability can be assured by various resilience schemes-protection, restoration, rerouting, etc.-that have very different recovery times and resource consumptions[7][8][9]. Since mission-critical data are supposed, we deal with a protection scheme that is the fastest resilience paradigm (with 10-100 ms recovery time) as the backup resources are previously reserved. Because it is necessary to guarantee connectivity even in case of network failures, so the protection plays more and more essential role in backbone networks.

As the key constraint to establish paths, SRLG is being researched intensively. SRLG is defined as a group of links or nodes that share a common risk component, whose faults can potentially cause the failure of all the links or nodes in the group[10]. For example, all fiber links that go through a common conduit belong to the same SRLG, because the conduit is a shared risk component whose failure, such as a conduit cut, may cause all fibers in the conduit to be broken simultaneously. This SRLG is introduced in the GMPLS and can be identified by a SRLG identifier, which is typically a 32-bit integer.

On the other hand, once a primary path is found, one may not be able to find a SRLG-disjoint backup path (even though a pair of SRLG disjoint paths do exist using a different primary path). This is the so-called trap problem[11], which is rarely present when finding link/node disjoint paths using SPF but can occur much more frequently (e.g., with a probability of up to 30 percent in a typical optical network) when finding SRLG-disjoint paths. In this paper, we find a joint primary and backup path by considering costs together among searched k-shortest paths. This can improve blocking probability and resource efficiency simultaneously.

3 Protection Mechanism Under SG-MIPR

3.1 SG-MIPR Algorithm

In this section, we propose the SG-MIPR algorithm as a new dynamic routing algorithm. This algorithm chooses a route that does minimize interference for potential future connection requests by avoiding congested links. These are links with the property that the available wavelengths on the minimum hop routes of one or more node-pairs decreases whenever a lightpath is routed over those

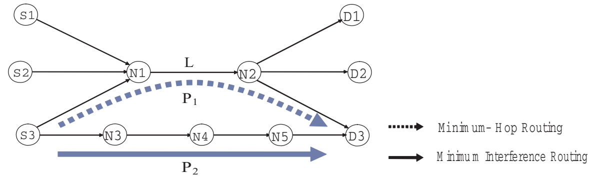


Fig. 2. SG-MIPR basic concept

links. By reducing the number of wavelengths in a congested link, the number of failed connections by a single failure can be decreased as well.

Figure 2 illustrates the SG-MIPR basic concepts. For example, SG-MIPR is to pick route P_2 for connection between (S3, D3) pair that has a minimum affect for other connection requests (S1, D1) as well as (S2, D2) even though the path is longer than P_1 with a congested link L. Before formulating the SG-MIPR algorithm, we define some notations commonly used in this algorithm as follows:

- $G(N, L, W)$: The given network, where N is the set of nodes, L is the set of links, and W is the total set of wavelengths per link.
- M: Set of potential source-destination node pairs that can request a connection in the future. Let (s,d) denote a generic element of this set.
- p_{sd} : The minimum hop lightpath between a (s,d)-pair, where ∀(s,d) ∈ L.
- π_{sd} : Set of links over the minimum hop path p_{sd}.
- $R(l)$: The number of available wavelengths on a link l, where ∀l ∈ L.
- Λ_{sd} : The union set of available wavelengths on each link l, where ∀l ∈ π_{sd}.
- F_{sd} : The set of available wavelengths on the bottleneck link that has the smallest residual wavelengths among all links within π_{sd}, i.e., ∀l ∈ π_{sd}.
- Ω_{sd} : Set of wavelengths assigned to the minimum hop path p_{sd}.
- C_{sd} : Set of critical links for a (s,d)-pairs, where ∀(s,d) ∈ M.
- α_{sd} : The weight for a (s,d)-pair, where ∀(s,d) ∈ M.

Among the above notations, C_{sd} and α_{sd} are key parameters in the SG-MIPR algorithm. C_{sd} indicates critical links belonging to π_{sd} of a (s, d)-pair. These links have higher congestion possibility for potential future requests than other links within π_{sd}. Thus, this notation is necessarily considered for determining a critical link. α_{sd} is the weight for each node pair, which is chosen in order to reflect the "importance" of a (s, d)-pair where ∀(s,d) ∈ M. Based on these notations, the ultimate object of SG-MIPR is represented below in Equation 1. It is a maximum available wavelengths problem for each source-destination pair in M except the current demands between nodes a and b, where ∀(a,b) ∈ M.

$$max \sum_{(s,d)\in M\setminus(a,b)} \alpha_{sd} \cdot F_{sd} \tag{1}$$

To achieve Equation 1, the SG-MIPR algorithm routes the current demand along a path that does not interfere too much with potential future requests. We define a route between a (a, b)-pair selected by SG-MIPR as p^m_{ab} and similar to the above-mentioned notations, we use π^m_{ab}, $\bigwedge^m_{ab}$, $F^m_{ab}$and Ω^m_{ab}. And for the wavelength assignment problem on the route p^m_{ab} selected by SG-MIPR, we adopt first-fit as the wavelength assignment algorithm, which requires no global information, so that it achieves not only low cost but also good efficiency.

The number of available wavelengths on a link is regarded as an important factor to improve network performance in terms of blocking probability. Therefore, we add a new notation Δ as a threshold value of the available wavelengths on a link to choose the minimum interference path for potential future connection requests with consideration of critical links as well as non-critical links with few wavelengths. Therefore, the appropriate choice for threshold values is very important for efficient wavelength utilization. In this paper, we set the threshold value Δ within 20% or 30% of the total wavelength number on a link. This ratio is assumed by our simulation results regardless of the number of wavelengths per a link. Based on notations such as C_{sd} and Δ, we determine links with congestion possibility for a potential future demand between a (s, d)-pair according to Equation 2, where ∀(s,d)∈M\(a,b) and ∀l∈L, and call them SG_CL$_{sd}$.

$$SG\_CL_{sd} : (l \in C_{sd}) \cap (R(l) < \Delta), \forall(s,d) \in M \setminus (a,b), \forall l \in L \tag{2}$$

If a link l belongs to the set of critical links, i.e., l∈ C_{sd} and the number of residual wavelengths on that link is lower than the threshold value, i.e., R(l)< Δ, then the link l is the critical link. The SG-MIPR algorithm gives appropriate weights to each link based on the amount of available wavelengths on a link l where ∀l∈L, so that the current request does not interfere too much with potential future demands. The link weights are estimated by the following procedures. First, let $\partial F_{sd}/\partial$R(l) indicates the change of available wavelengths on the bottleneck link for the potential connection request between a (s, d)-pair when the residual wavelengths of link l are changed incrementally. With respect to the residual wavelength of the link, the weight w(l) of a link l is set to

$$w(l) = \sum_{(s,d)\in M\setminus(a,b)} \alpha_{sd}(\partial F_{sd}/\partial R(l)), \forall l \in L \tag{3}$$

Equation 3 determines the weight of each link for all (s, d)-pairs in the set M except the current request when setting up a connection between the (a, b)-pair, i.e., (s,d)∈M\(a,b), but computing weights for all links is very hard, where ∀l∈L. To solve this problem, we consider more restricted links than other links for routing with Equation 4 if a link belongs to the set of congestion links for a certain (s, d)-pair, i.e., l∈SG_CL$_{sd}$. Therefore, computing the link weights is simplified as shown in Equation 5.

$$\begin{cases} \partial F_{sd}/\partial R(l)=1[if(s,d):l\in SG\_CL_{sd}] \\ \partial F_{sd}/\partial R(l)=0[otherwise] \end{cases} \tag{4}$$

$$w(l) = \sum_{(s,d):l\in SG\_CL_{sd}} \alpha_{sd} \tag{5}$$

Once the weight of each link l where ∀l∈L is determined, SG-MIPR routes the current traffic between the (a, b)-pair along the path with the smallest w(l) to achieve Equation 1.

3.2 Joint Path Selection Approach Under SG-MIPR

In this subsection, we formulate equations for the joint primary path and backup path selection under SG-MIPR algorithm. From Equation 5, we define the equations as the primary path cost WP and the backup path cost WB as Equations 6 and 7, respectively.

$$WP(p) = \sum_{l \in (a,b) \backslash p} \sum_{(s,d) \in M \backslash (a,b)} \alpha_{sd} \tag{6}$$

$$WB(p) = \sum_{l \in (a,b) \backslash p} \sum_{(s,d) \in M \backslash (a,b)} \Theta(\alpha_{sd}, l) \tag{7}$$

While deploying only dedicated path protection, $\Theta(\alpha_{sd}$,l) equals to α_{sd}. Consequently, we accomplish the optimization by finding minimum TC=WP+WB.

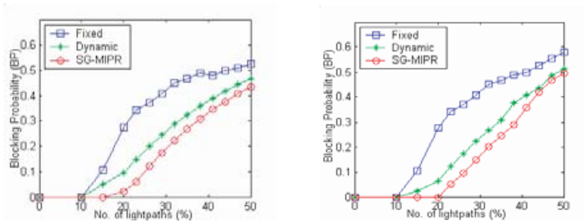
4 Performance Evaluation

Simulations are carried out to prove the efficiency of SG-MIPR algorithm and wavelength utilization and restorability of joint path protection scheme. We use two test networks:(14 nodes, 20 links), (30 nodes, 61 links) and three service classes:Premium Service(PS), Assured Service(AS) and Best Effort Service(BES). Each service requires 1:1 dedicated protection, 1:3 shared protection and dynamic path restoration, respectively. Also, we assume the connection requests arrive randomly according to the Poisson process, with negative exponentially distributed connection times with unit mean.

First, we compare the proposed SG-MIPR to the existing routing (fixed routing and dynamic routing) algorithms. The plots of blocking probability in both test networks are illustrated in figure 3. In both test networks, the results indicate that the proposed SG-MIPR algorithm has lower blocking probability than dynamic routing (improved by about 5–10%) because of selecting the minimum interference path with potential future setup requests.

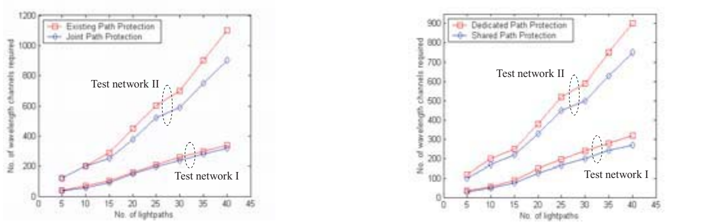
Figure 4(a) depicts the benefit of the joint path protection scheme over the existing path protection scheme using modified SPF. This is evaluated in PS. And the performance metric used is the number of wavelength channels in all links as a function of number of lightpaths. This shows the wavelength saving and also better performance in larger network (test network II). Moreover, figure 4(b) illustrates the effect of both dedicated and shared path protection schemes in 1:3 shared protection. In the shared protection, the number of wavelength channels presents that it achieves wavelength saving.

Figure 5 shows the survivability ratio for each service class in case of single SRLG failure and double SRLG failures. PS achieves 100% restorability for any



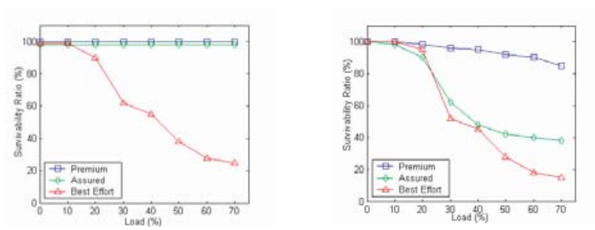
(a) Test network I (14 nodes and 20 links) (b) Test network II (30 nodes and 61 links)

Fig. 3. Blocking probability for fixed, dynamic and SG-MIPR



(a) Existing path protection vs. Joint path protection (b) Dedicated path protection vs. Shared path protection

Fig. 4. The numerical results for the number of wavelength channels required



(a) Single SRLG failure (b) Double SRLG failure

Fig. 5. Survivability ratio of PS, AS and BES in test network II

single failure and almost 90% for double failures. And AS (1:3 shared protection), when single SRLG failure occurs, achieves 100% restorability because AS is established by considering SRLG constraint. However, for double SRLG failures, AS has lower survivability ratio, but it is possible to utilize the capacity more efficiently while still achieving over minimum 30%. As for BES, dynamic path restoration can guarantee only relative survivability, according to residual wavelengths. This phenomenon occurs due to discovering a backup path after the primary path fails, not to reserve a backup path in advance.

5 Conclusion

In this paper, we proposed a routing method by choosing a route that does not interfere too much with potential future connection requests, called SG-MIPR. Furthermore, under SG-MIPR algorithm, we suggested a joint primary and backup path selection scheme under SRLG constraint and TA problem. To verify the performance of the proposed approaches, simulations were carried out in terms of blocking probability, number of wavelength channels required and survivability ratio. Through the simulation results, the proposed SG-MIPR algorithm improved blocking probability about 5–10% than the existing dynamic routing algorithm. And the proposed joint path protection scheme achieved the wavelength saving and better performance in larger networks. Moreover, for the differentiated services under joint path protection scheme, PS achieved almost 100% restorability for any single failure and approximately 90% for double failures. For future research, we envisage that the proposed routing algorithm and recovery schemes can be applied to GMPLS for control protocol in DWDM networks.

Acknowledgment. This work was supported by grant No.(R01-2003-000-10526-0) from Korea Science and Engineering Foundation.

References

1. T. E. Stern and K. Bala, Multiwavelength Optical networks: A layered approach, Addition Wesley Publishers, 1999.
2. I. Chlamtac et al., Lightpath communications-an approach to high-bandwidth optical WANs, IEEE Trans. on Comm., vol.40, no.7, pp.1171-1182, July 1992.
3. J. S. Choi et al., Classification of routing and wavelength assignment schemes in DWDM networks, Proc. of OPNET 2000, pp.1109-1115, Jan. 2000.
4. Jong-Gyu Hwang et al., A RWA Algorithm for Differentiated Services with QoS Guarantees in the Next Generation Internet based on DWDM Networks, Photonic Network Communications, vol.8, no.3, pp.319-334, November 2004.
5. S. Xu et al., Dynamic routing and assignment of wavelength algorithms in multifiber wavelength division multiplexing networks, IEEE Selected Areas in Comm., vol.18, no.10, pp.2130-2137, Oct. 2000.
6. M. Frey et al., Wavelength conversion and call connection probability in WDM networks, IEEE Transactions on Comm., vol.49, no.10, pp.1780-1787, Oct. 2001.
7. S. Ramamurthy and B. Mukherjee, Survivable WDM Mesh Networks, Part I - Protection, Proceedings of IEEE INFOCOM'99, pp.744-751, Mar. 1999.
8. O. Crochat et al., Design Protection for WDM Optical Networks, IEEE Journal Selected Areas Comm., vol.16, no.7, pp.1158-1165, Sep. 1998.
9. D. Stamatelakis et al., IP Layer Restoration and Network Planning Based on Virtual Protection Cycles, IEEE Sel. Areas, vol.18, no.10, pp.1938-1949, Oct. 2000.
10. D. Papadimitriou et al., Inference of Shared Risk Link Groups, draft-many-inference-srlg-02.txt, Internet Draft, Nov. 2001.
11. Dahai Xu et al., Trap avoidance and protection schemes in networks with shared risk link groups, IEEE Lightwave Tech., vol.21, no.11, pp.2683-2693, Nov. 2003.

On Integrated QoS Control in IP/WDM Networks

Wei Wei[1,2], Zhongheng Ji[2], Junjie Yang[1], and Qingji Zeng[1]

[1] Department of Electrical Engineering, Shanghai Jiaotong University,
Shanghai 200030, P.R. China
{Wwei, Qjzeng, Yangjunjie}@Sjtu.edu.cn
[2] Institute of Information technologies, Information Engineering University,
Zhengzhou 450002, P.R. China
{Ww, Jzh}@Ndsc.com.cn

Abstract. In order to facilitate convergence of networks and services, we propose a new hybrid and integrated QoS control scheme that combines electrical IP layer features with reconfigurable optical layer, and investigate a comprehensive service differentiation mechanism of integrating both IP and optical QoS functionalities. The proposed integrated QoS control scheme can: 1) provide appropriate transport service for various applications relating to different service categories; 2) maintain high flexibility/scalability for integrated services provisioning; and 3) meet carrier-class QoS requirements.

1 Introduction

IP/WDM network (i.e., optical Internet) is becoming a common backbone for most of network providers, which will simultaneously offer multiple service classes capable of supporting both real-time (e.g., streaming media traffic) and non-real-time traffic (e.g., data traffic). For the next generation optical Internet based on Generalized Multi-Protocol Label Switching (GMPLS), integrated network architecture can make more efficient use of network resources both at IP and optical layers [1-3]. In this architecture, properly designed multi-service optical routers with multi-granularity switching capability based on GMPLS can improve network's forwarding performance for their functionalities of flexible traffic aggregation/grooming, dynamic virtual topology adaptation/reconfiguration, optical bypassing, etc.

For multi-service multi-granularity integrated IP/WDM networks, different QoS mechanisms may be possible but no single "best" layer can be derived to maintain the required QoS in a cost-effective manner. The problem of providing QoS guarantees in a cost-effective manner to different services remains largely unsolved in IP/WDM networks. The majority of research works consider the IP and optical QoS control mechanisms separately [4,6,7-15]. Harmonization between the two approaches is becoming the main issue so that one technology can complement with the other towards QoS provisioning in integrated IP/WDM networks. Recent technological developments in both IP and optical networking are inevitably bringing the two domains closer together [1-5], which indicate that we could combine multi-layer separate QoS mechanisms (i.e., IP and optical QoS mechanisms) into a single one.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 334–341, 2005.

In this paper, we investigate the problem of integrated QoS control in IP/WDM networks. The rest of this paper is organized as follows. Section 2 proposes a differentiated architectural model of integrated QoS control in IP/WDM networks. In Section 3, we present the experimental results. Section 4 concludes the paper.

2 Differentiated Service Model of Integrated QoS Control

The large variety of QoS requirements and the needs of carrying multiple services efficiently, will lead to the coexistence of several kinds of connection modes (e.g., circuit connection, virtual circuit connection, and datagram forwarding) in future integrated IP/WDM networks.

2.1 Architectural Model

Based on the GMPLS framework, we propose an architectural model of multi-layer integrated QoS control in a comprehensive and efficient way as shown in Fig. 1.

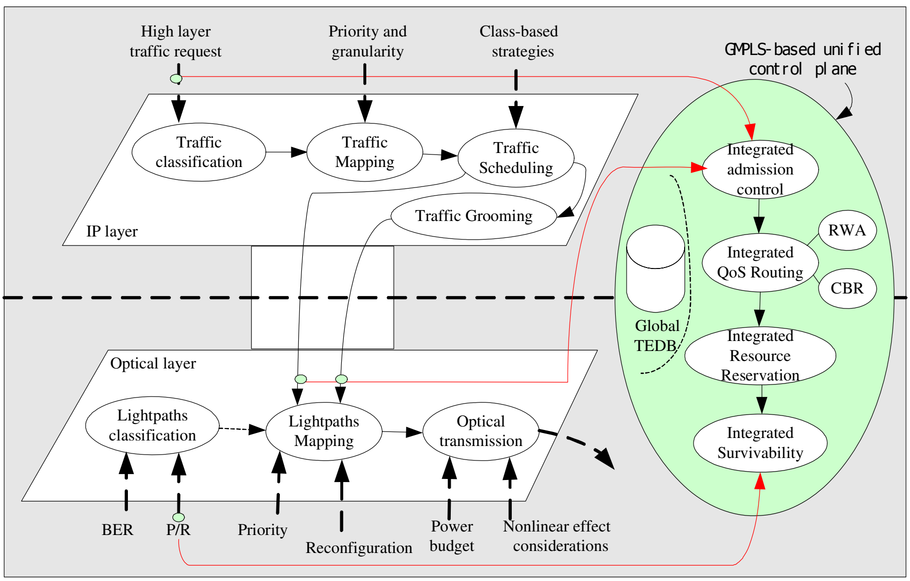


Fig. 1. Architectural model of integrated QoS control in IP/WDM networks

The proposed differentiated model implicitly implements a distributed traffic-based prioritization mechanism in a comprehensive way by providing adaptive traffic/lightpath classification, integrated admission control, traffic grooming/traffic mapping strategies. For example, for multi-priority traffic requests with multi-granularity bandwidth requirement, according to network conditions (resource usage and traffic load), optical routers can intelligently conduct the differentiated forwarding operation (e.g., single-hop or multi-hop forwarding) subject to the QoS constraints of the traffic flows, where the optical layer QoS can be adaptive to meet the current differentiated QoS requirement in IP layer. In this architectural model, the following components play key roles. 1) Integrated admission control (IAC) can be described as making admission decisions by comparing the resources required by an incoming traffic request with the resources currently available in both IP layer and optical layer, which take into account both packet level QoS constraints and lightpath level constraints. 2) Integrated QoS routing algorithm selects a near-optimal path to meet the required QoS of a traffic request, taking into account the combined topology and resource usage information of both IP and WDM layers [16]. 3) Integrated survivability scheme merges IP layer survivability mechanisms with optical layers, which would lead to efficiently recover from a fault [17]. 4) Adaptive traffic/lightpath classification/mapping strategies, based on QoS policies in management plane, are combined to meet the subscriber's QoS requests. 5) Intelligent traffic grooming has been used as a simple and robust interworking strategy to coordinate the two-layer QoS mechanisms. In addition, interactions between layers such as virtual topology adaptation/reconfiguration are also needed.

Table 1. Classes of service in IP layer

CoS	**Traffic Classes**	**Bandwidth Requirement**		**Delay/Jitter/PLR**	**Queuing**
A	Hard QoS guaranteed (e.g. CES, VPN tunnel)	Peak bandwidth guaranteed		Minimum packet delay, jitter, and packet loss ratio	Usually single-hop transport
B	Soft QoS guaranteed (e.g. VoIP trunks, DTV, grid computing)	Under subscription	Guaranteed	To meet the given QoS performance metrics	Usually multi-hop transport
		Oversubscription	Not guaranteed		
C	Best Effort (e.g. FTP, e-mail)	Only provide basic connectivity		N/A	Usually multi-hop transport

Table 2. Classes of service in optical layer

CoS	Traffic Classes	BER	Survivability	Security	Preemptible
A	High quality (HQ) lightpaths	$>10^{-9}$ (Under full link load)	50ms 1+1/1:1 protection	Guaranteed	No
B	Low quality (LQ) lightpaths	$>10^{-5}$ (Under full link load)	Dynamic Restoration (or no)	Not guaranteed	Yes

2.2 Traffic Classification Strategies

The proposed traffic classification strategies are illustrated in Table 1 and Table 2. It employs three categories for aggregated traffic requests and two categories for lightpaths.

2.3 Traffic Mapping Strategies

The issue of QoS mapping strategies in a given optical Internet domain is divided into two topics: 1) "vertical" mapping linking the QoS mechanisms of application layer, electrical layer, and optical layer; 2) "horizontal" mapping mechanism to link the QoS control mechanisms between optical router nodes. As shown in Fig. 2 and Fig. 3, the proposed integrated mapping model of application layer to IP layer QoS to the optical layer QoS is to allow applications to select the IP layer service (class A, B and C) that best suits their needs.

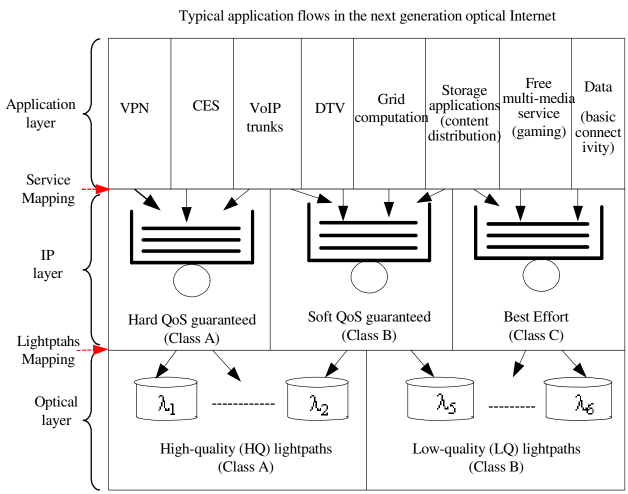


Fig. 2. Traffic vertical mapping model

The transmission impairments can impact the packet loss ratio of the connection carried over the lightpaths [18]. In fact, amplified spontaneous emission (ASE) noise in optical amplifiers, insertion loss and crosstalk introduced by optical routers and attenuation and polarization mode dispersion (PMD) effects introduced by the fibers can degrade the optical signal resulting in a very high BER. The proposed mapping strategies consider the routing of a connection over a single- or multi-hop (HQ or LQ) lightpath adaptively according to the QoS requirements. For example, if a packet-loss sensitive traffic flow (e.g., DTV) is carried over a single-hop HQ lightpath experiencing less transmission impairments, less signal-noise-ratio (SNR) degradation could meet the signal quality requirements. In contrary to this, if a connection carrying delay-sensitive traffic (e.g., VoIP) is routed over multi-hop LQ lightpaths, the output signal could suffer very high end-to-end delay due to the electrical grooming and queuing delays experienced along the path, which will violate the requirement of QoS.

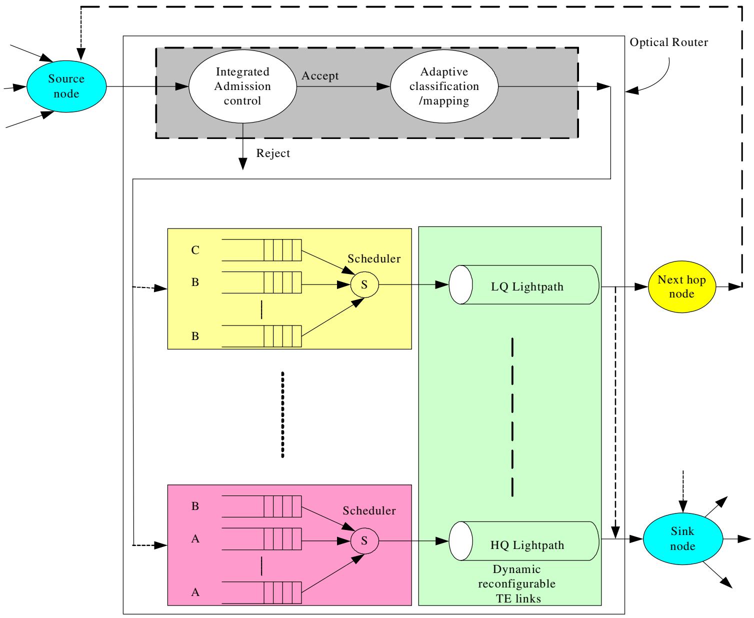


Fig. 3. Traffic horizontal mapping model

3 Performance Evaluation

To evaluate the proposed multi-layer integrated QoS scheme, we establish a simple experiment of a 3-node topology constituted of a source optical router, an intermediate optical router, and a sink optical router interconnected by two WDM links. This topology also consists of four traffic flows generators with their destinations respectively; one of them set up circuit emulated service (CES) traffic flows, two of them set up packet voice/video flows and there is one for the traditional data traffic flows. For comparisons with the single-layer QoS control, we replace the optical router model with the so-called Big Fat Router (BFR) model [19] without admission control[1]. In the experiment, the setups are as follows. We design four kinds of typical traffic flows request: a) $R1$ represents high-speed CES traffic (e.g., CBR traffic source) which belongs to traffic class A, it is assumed that $V_a^f = 155$ units; b) $R2$ represents large-granularity streaming media traffic which belongs to traffic class B,

[1] BFR model means offering only a single IP layer of QoS control (no optical-layer QoS control is employed), here we assume it uses First In First Out (FIFO) packet servicing (i.e., 'best effort' service) for all traffic flows.

it is assumed that $V_{b,averge}^{f} = 100$ units, $V_{b,peak}^{f} = 150$ units; c) $R3$ represents small-granularity streaming media traffic which belongs to traffic class B, it is assumed that $V_{b,averge}^{f} = 30$ units, $V_{b,peak}^{f} = 80$ units; d) $R4$ represents BE traffic flow request which belongs to traffic class C, it is assumed that $V_{c,\max} = 1000$ units, $V_{c,\min} = 1$ unit, which means that for each $R4$ request, the minimum guaranteed bandwidth is 1 unit, the maximum bandwidth does not exceed 1000 units.

As shown in Fig. 4, we evaluate the basic QoS performance of the proposed integrated QoS control as well as the comparisons with the single IP layer QoS control (FIFO service). Fig. 4(a) shows the results of call blocking probability versus network load under various kinds of traffic requests. We can find that $R1$ is greater than $R2$, $R3$, and $R4$, while the single IP layer QoS control is less than $R2$, $R3$, and $R4$, which indicates that the call blocking probability, is mainly affected by the bandwidth granularity of traffic request. It relies on the above assumption of the same call intensity of each kinds of traffic request. However, for real traffic distribution in current optical Internet, the BE traffic requests are far greater than real-time (RT) traffic. In Fig. 4(b), we have shown that the proposed method provides small average packet loss probability (PLP) for the high-priority traffic class. $R1$ always has the best performance. For the traffic class B requests, $R2$ tends to give smaller average PLP than $R3$ because $R3$ is usually transported by LQ lightpath. However, under higher call load conditions (approximately greater than 100 Erlangs, when network becomes congested), the PLP does not exceed $PLR_b = 10^{-6}$. As for the single layer QoS control, the performance of average packet loss probability is severely affected by higher load. We observe that even under relatively high congestion, the integrated control scheme can provide QoS guarantees for RT traffic flows in contrast to single-layer QoS control. This robustness seems to be preferred for QoS provisioning in the next generation multi-services optical Internet in order to alleviate the problem of congestion. Fig. 4(c) shows the performance simulation results of the average packet delay vs. call load for various kinds of traffic requests. From Fig. 4(c), we can find that the integrated QoS control for RT traffic performs significantly better than the single IP layer QoS control in terms of the performance of average packet delay under higher network load conditions. The reason lies in two aspects: 1) forwarding differentiation-BE traffic is usually transported on the multi-hop LQ lightpaths by grooming, while RT traffic is usually transported on the single-hop HQ lightpaths directly, and 2) queuing differentiation-BE traffic needs more queuing time than RT traffic because of different scheduling strategies. We show the simulation results of the lightpath resource utilization (which is defined as the amount of used bandwidth over the total amount of bandwidth offered in a given lightpath) in Fig. 4(d). Since the HQ lightpaths usually carry $R1$, $R2$ traffic requests, and the LQ lightpaths usually carry $R3$, $R4$ traffic requests, we observe that the higher resource utilization of either HQ lightpaths or LQ lightpaths is basically not affected by the call load because of the adaptive optical layer reconfiguration. In additional, when call load increases, the $R4$ resource utilization decreases, while $R3$ increases. It is because $R3$ has higher priority than $R4$. Although a simple network is used in our experiment, we can predict that the QoS performance results will even become better in a large network (e.g., NSFnet) because abundant networking resource will be configured and more QoS control mechanisms will be employed (e.g., integrated QoS routing [16] and flexible traffic grooming/bypassing/management).

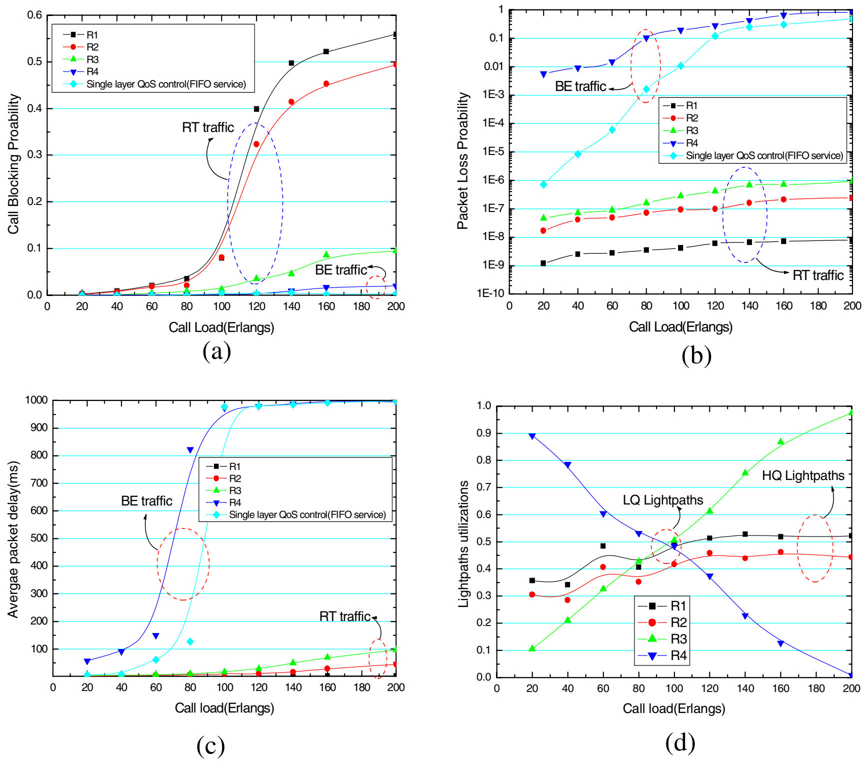


Fig. 4. Performance evaluation results of the integrated QoS control compared with the single-layer QoS control for various kinds of traffic requests: (a) call-blocking probability vs. call load; (b) average packet loss probability vs. call load; (c) average packet delay vs. call load; (d) lightpath utilization vs. call load

4 Conclusion

For the motivations of efficiently controlling the resource allocation and getting improved traffic quality in IP/WDM networks, we propose a differentiated QoS control framework by the integration of both electrical and optical QoS mechanisms as an effective and comprehensive scheme. The study reveal that the proposed scheme capture the better tradeoff between the finer QoS granularity of the IP layer and the coarse QoS granularity of the optical layer to support multiple levels of service performance in an integrated IP/WDM network.

References

1. Ghani, N., Dixit, S., Wang, T.: On IP-over-WDM integration, IEEE Communications Magazine. 38(3) (2000) 72-84.
2. Qiao, C.: Labeled Optical Burst Switching for IP and WDM Integration, IEEE Communications Magazine. 38(9) (2000) 104-114.
3. Mannie, E., Smith, P.A., Awduche, D., et al.: Generalized Multi-Protocol Label Switching (GMPLS) Architecture, Internet Draft. Draft-ietf-ccamp-gmpls-architecture-07.txt, May 2003.
4. Xiao, X., Ni, L.M.: Internet QoS: A Big Picture, IEEE Network. 13(2) (1999) 8-18.
5. Benjamin, D., Trudel, R., Shew, S., et al.: Optical services over the intelligent optical network, IEEE Communications Magazine. 39(9) (2001) 73-78.
6. Braden, R., Clark, D., Shenker, S.: Integrated Services in the Internet Architecture: an Overview, IETF RFC 1633, June 1994.
7. Blake, S., Black, D., Carlson, M., et al.: An architecture for differentiated services, IETF RFC 2475, December 1998.
8. Armitage, G.: MPLS: The Magic Behind the Myths, IEEE Communications Magazine. 38(2) (2000) 124-131.
9. Guerin, R., Peris, V.: Quality-of-service in packet networks: Basic mechanisms and directions, Computer Networks. 31(3) (1999) 169-189.
10. Perros, H.G., Elsayed, K.M.: Call admission control schemes: a review, IEEE Communications Magazine. 34(11) (1996) 82-91.
11. Chen, S., Nahrstedt, K.: An overview of quality-of-service routing for next-generation high-speed networks: Problems and solutions, IEEE Network. 12(6) (1998) 64-79.
12. Kaheel, A., Khattab, T., Mohamed, A., et al.: Quality-of-Service Mechanisms in IP-over-WDM Networks, IEEE Communications Magazine. 40(12) (2002) 38-43.
13. Golmie, N., Ndousse, T.D., Su, D.H.: A Differentiated Optical Services Model for WDM Networks, IEEE Communications Magazine. 38(2) (2000) 68-73.
14. Jukan, A., van As, H.R.: Service-specific resource allocation in WDM networks with quality constraints, IEEE Journal on Selected Areas in Communications. 18(10) (2000) 2051-2061.
15. Gravey, P., Gosselin, S., Guillemont, C., et al.: Multiservice Optical Network: Main Concepts and First Achievements of the ROM Program, IEEE/OSA Journal of Lightwave Technology. 19(1) (2001) 23-31.
16. Kodialam, M., Lakshman, T.V.: Integrated Dynamic IP and Wavelength Routing in IP over WDM Networks, in: Proceedings of IEEE Infocom, vol. 1, March 2001, pp. 358-366.
17. Wei, W., Zeng, Q., Wang, Y., Multi-Layer Differentiated Integrated Survivability for Optical Internet, Photonic Network Communications. 8(3) (2004) 267-284.
18. Ramamurthy, B., Datta, D., Feng, H., et al.: Impact of transmission impairments on the teletraffic performance of wavelength-routed optical network, IEEE/OSA Journal of Lightwave Technology. 10(17) (1999) 1713-1723.
19. Hjalmtysson, G., Yates, J., Chaudhuri, S., et al.: Smart routers-simple optics: an architecture for the optical Internet, IEEE/OSA Journal of Lightwave Technology. 18(12) (2000) 1880-1891.

Optical Hybrid Switching Using Flow-Level Service Classification for IP Differentiated Service

Gyu Myoung Lee and Jun Kyun Choi

Information and Communications University (ICU),
103-6, Munji-dong, Youseong-ku, Daejeon, Korea
{gmlee, jkchoi}@icu.ac.kr

Abstract. In a new optical hybrid switching environment which combined Optical Burst Switching (OBS) and Optical Circuit Switching (OCS), we propose flow-level service classification scheme for IP differentiate service. In particular, this classification scheme classifies incoming IP traffic flows into short-lived and long-lived flows for Quality of Service (QoS) provisioning according to traffic characteristics such as flow bandwidth, loss and delay. In incoming IP service classes, long-lived flows include premium service and loss sensitive service to take an advantage of OCS. On the other hand, short-lived flows include Best-Effort service and delay sensitive service to take an advantage of OBS. Therefore, optical hybrid switching network can take advantages of both switching technologies using the proposed flow classification scheme. The aim of proposed technique is to maximize network utilization while satisfying user's QoS requirements. We show results for the delay, OBS burst size and burst assembly times.

1 Introduction

Optical network technologies are evolving rapidly in terms of multiplexing bandwidth and control capability. There has been considerable attention given to IP over optical networks to combine the optical and the electronic worlds by network service providers, telecommunications equipment vendors, and standards organizations.

From the optical switching technology point of view, it is known that the Optical Circuit Switching (OCS) networks achieve low bandwidth utilization with burst traffic such as Internet traffic. So, sophisticated traffic grooming mechanism is needed to support statistical multiplexing of data form different users. Optical Burst Switching (OBS) technology has been emerging to utilize resources and transport data more efficiently than the existing circuit switching [1]-[2]. OBS is accepted as an alternative switching technology due to the limitation of optical devices that do not support buffering.

The OBS and OCS have the advantages and disadvantages in performance point of view. So we can consider the so-called hybrid switching. The optical

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 342–349, 2005.

hybrid switching [3]-[4] is a new switching technique which combines OCS and OBS to take advantages of both switching technologies and to improve their performance degradation. The OCS module of optical hybrid switching can avoid the several overheads for long-lived flows and reuse the current OCS network technology. On the other hand, the OBS can improve the resource utilization for short-lived flows such as bursty IP traffic. In this paper, we consider a combined OCS and OBS system and a hierarchical Quality of Service (QoS) mapping architecture.

We propose the flow-level service classification scheme for IP differentiated service. This scheme classifies incoming IP traffic flows into short-lived and long-lived flows for QoS provisioning according to traffic characteristics in an optical hybrid switching environment. The incoming IP traffic flows divided into premium service, assured service and best-effort service for IP differentiated service as described in [5]-[6]. Short-lived flows are composed of a few packets and better suited for OBS which has a short-delay characteristic than OCS. Long-lived traffic typically indicate loss-sensitive or real-time video streams that are better suited for circuit (or wavelength) switching which has an advantage of loss-less through connection establishment. Therefore, the optical hybrid switching technique using the proposed flow classification scheme takes advantages of both OBS and OCS. The aim is to maximize network utilization while satisfying users' QoS requirements.

The remainder of the paper is organized as follows. In Section 2, we explain the hierarchical QoS mapping architecture of optical hybrid switching network. In Section 3, we propose the new flow-level service classification scheme in optical hybrid switching networks for IP differentiated service and show the example of implementation. Then, in Section 4, we give numerical results for the proposed network.

2 Hierarchical QoS Mapping Architecture in Optical Switching Network

We consider the architecture of optical IP network for optical hybrid switching as shown in [8]. This network is composed of IP/Multiprotocol Label Switching (MPLS) network and optical sub-network. In this network architecture, IP/MPLS routers are attached to an optical sub-network, and connected to their peers over dynamically established switched lightpaths. The IP/MPLS routers handle the Label Switched Paths (LSPs) for end-to-end virtual flows. Fig. 1 shows the hierarchical QoS mapping architecture. The optical edge router which performs optical hybrid switching is required the flow-level QoS mapping through classification of MPLS flows. For end-to-end QoS provisioning, each node performs the QoS mapping of different level.

The IP/MPLS router in an IP/MPLS network is generally able to perform various operations in packet level. These operations include label swapping, label merging and label stacking. Packet level QoS mapping between IP/MPLS routers is performed for the end-to-end QoS provisioning. Incoming IP packets

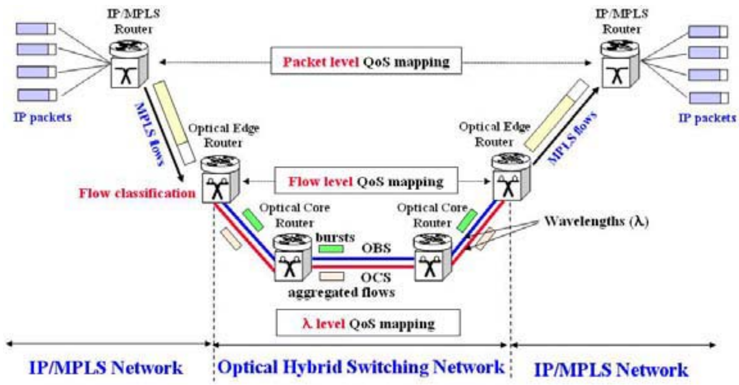


Fig. 1. Hierarchical QoS mapping architecture

are mapped into MPLS flows. Here, a flow is defined as a set of packets travelling between a pair of hosts with the same destination address but the different source addresses [9]. The optical edge router in the optical hybrid switching network performs flow classification and QoS mapping in flow level. Here is the start point of optical hybrid switching. Thus, outing flows are divided into aggregated flows (e.g., long-lived flows) for OCS and data bursts (e.g., short-lived flows) for OBS. The detailed operation will be discussed in the next Section. Lambda level QoS mapping between optical core routers in the optical switching network is performed. Each wavelength is mapped onto the corresponding optical switching interface which is satisfied with QoS constraints.

3 IP Differentiated Service Using Optical Hybrid Switching

3.1 Flow Classification Model for Optical Hybrid Switching

In the considered optical hybrid switching environment, flow classification is performed at the border of access network and core network. Fig. 2 shows the flow classification model for optical hybrid switching.

In this model, the incoming IP differentiated services [10] of access network are classified into two kinds of mode: OBS using one-way reservation and optical fast circuit switching with real time two-way reservation. The main goal of flow classification model is to provide a mechanism for offering IP differentiated service in optical network through hybrid switching scheme.

Time Division Multiplexing (TDM) trunks (leased line) in the form of DSx or tributaries on SONET/SDH, such as from electric circuit switches, and Synchronous Optical NETwork (SONET)/ Synchronous Digital Hierarchy (SDH) formatted optical links should be separated at the edge of optical core network. In our flow classification model, TDM and SONET/SDH use optical fast circuit switching using real-time two-way reservation.

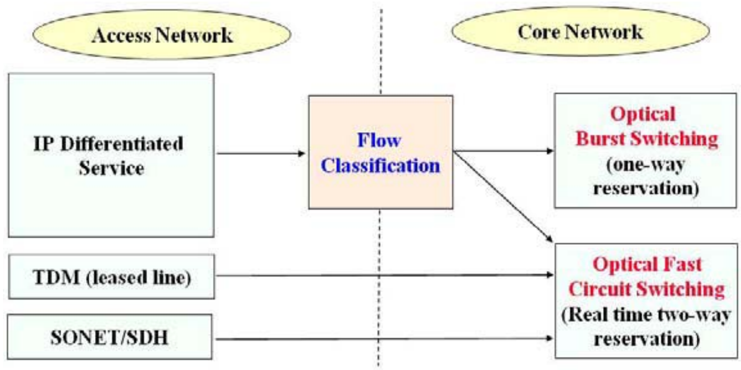


Fig. 2. Flow classification model for optical hybrid switching

We can classify the incoming traffic types using the value of flow bandwidth threshold in the relationship of flow bandwidth and the number of packets. Short-lived flows are composed of a few packets such as e-mail, light-loaded FTPs and so on. These flows are better suited for OBS. Long-lived flows contain a large number of packets, that is, stream media. These flows are better suited for OCS. We can consider other type of traffic. For example, big burst such as very high-load FTPs and images require very high bandwidth for a short period of time and require special reservation. This case is better suited for wavelength routed OBS (WR-OBS) [11]. The reservation of this switching scheme is made for the entire burst before it is transmitted.

Table 1 shows the proposed QoS classification in optical hybrid switching network with hierarchical QoS mapping architecture. The packet level QoS service is divided into three services for IP differentiated service [6]. We propose the flow level QoS service which classifies incoming IP differentiated service into

Table 1. The proposed QoS classification in optical hybrid switching network

Packet level	Flow level	λ level
Premium service (EF PHB) • Virtual leased line • Bandwidth pipe for data service	**Long-lived flow (loss sensitive traffic)** • Guaranteed service	**Class 1** • Survivability – 90% • Secure • 1R (regeneration)
Assured service (AF PHB) • Minimum rate guarantee service • Qualitative Olympic service • Funnel service		**Class 2** • Survivability – 70% • Unsecure • 2R (1R+reshaping)
	Short-lived flow (delay sensitive traffic) • Class-based priority service	**Class 3** • Survivability – 20% • Unsecure • 3R (2R+retiming)
Best Effort service (Default PHB)		

long-lived flows and short lived flows. For the lambda level QoS service, we can use the differentiated optical service model [12] according to survivability, security and provisioning.

3.2 Flow-Level Service Classification for IP Differentiated Service

For the purpose of Traffic Engineering (TE) and control it is most convenient to characterize demand at flow level. Optical edge router has a role to classify the incoming traffic flow for operating switching system in hybrid mode. Therefore, we proposed the flow classification scheme which classifies the incoming IP differentiated service flows into long-lived and short-lived flows. Table 2 shows the flow-level service classification and features for IP differentiated service.

In incoming IP service classes, long-lived flows include premium service and loss sensitive service among assured services to take an advantage of OCS. On the other hand, short-lived flows include Best-Effort service and delay sensitive service among assured services to take an advantage of OBS. OCS network is connection-oriented network which can support lossless transmission. So the loss-sensitive service is better suited for OCS. The reason why the delay sensitive service use OBS in short-lived flows is that the pretransmission latency of OBS is lower than that of OCS due to one-way reservation (link-by-link) and OBS requires limited or even no delay of data intermediate nodes as OCS. OBS cannot avoid the loss because of connection-less network The outgoing optical services at optical edge router are divided into guaranteed service and class-based priority service. In the cased of guaranteed service, we use admission control to provide guaranteed QoS for users. OCS can provide TE and QoS guarantee using two-way reservation (end-to-end) scheme. In the case of class-based priority service, we can provide service differentiation using burst scheduling algorithm, offset time adjustment scheme, and other schemes.

Next, we explain a new implementation scheme for optical hybrid switching using flow-level service classification in optical edge router. For QoS provision-

Table 2. Flow-level service classification and features

Classification	Long-lived flows		Short-lived flows		
Incoming IP service class	Premium service	Assured service			Best-Effort service
		Loss-sensitive	Other classes	Delay-sensitive	
Outgoing optical service class	Guaranteed service (admission control – call blocked)		Class-based priority service		
			Class 1	Class 2	Class 3
Switching	OCS		OBS		
Reservation	Two-way reservation (end-to-end)		One-way reservation (link-by-link)		
Loss rate	Loss-less		low	medium	high
Connection	Connection-oriented		Connection-less		

ing according to traffic characteristics, an incoming IP traffic flows are classified into short-lived and long-lived flows. The specific classification mechanism uses the existing adaptive flow classification [13]. For short-lived traffic flows, we use OBS using one-way reservation scheme with only request procedure to achieve better bandwidth utilization because it allows statistical sharing of each wavelength among bursts that may otherwise consume several wavelengths. So, these flows are performed per class burst assembling process and then data burst is created. On the other hand, for long-lived traffic flows such as video streaming, we consider the aggregation of these flows into aggregated flows for optical circuit/wavelength switching using two-way reservation scheme with request and acknowledgement procedure. Flow aggregator performs traffic aggregation according to flow characteristics. These aggregated flows require buffering and scheduling because flows are grouped together subject to specific constraints such as QoS class and destination.

4 Performance Results

In this section, we present result of end-to-end performance for OBS. In particular we show you the relation of burst size concerning end-to-end delay constraints [7]. While the burst size and the assembly time have a potent influence on the delay performance, OBS has more advantages for delay than OCS due to one-way reservation. Through the end-to-end delay analysis, we would like to give you a good guideline to find the optimal burst size and offset time for efficiently operating optical hybrid switching.

Fig. 3 shows the end to end delay versus the offered load for different burst size when hop distance is 10 and the fixed offset time is $70\mu s$. From this result, we can see that the end-to-end delay is related to the burst size. Next, we show

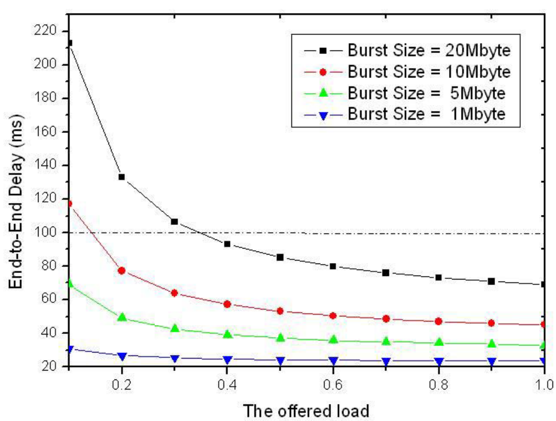


Fig. 3. End-to-end delay vs. the offered load for different burst size (n=10, fixed offset time=$70\mu s$)

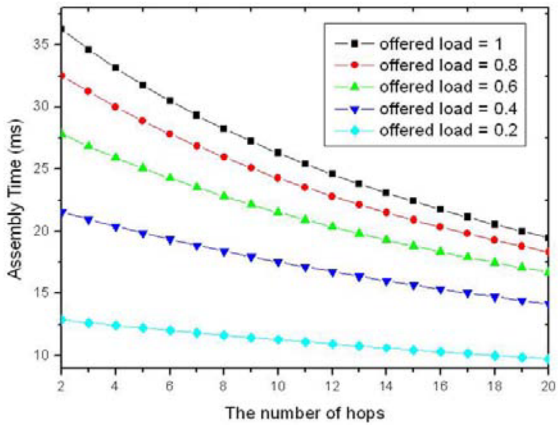


Fig. 4. Assembly time versus the number of hops for different offered load (end-to-end delay=100ms)

the performance relationships of assembly time concerning end-to-end delay constraints. The large burst size requires enough time to assembly the packets, then the delay increases due to assembly time as shown in Fig. 4. Fig. 4 shows the assembly time versus the number of hops when the end-to-end delay is 100ms. Here, we show the effect of hop count change. The assembly time is not significant at the high offered load, but for the low offered load, the assembly time to generate the fixed size burst is very critical in the performance of delay.

5 Conclusions

In this paper, we have proposed QoS provisioning algorithm using flow-level service classification in a new optical hybrid switching system which combines OBS and OCS. To support IP differentiated service in optical hybrid switching network, the proposed flow classification scheme classifies the incoming IP differentiated service flow into long-lived and short-lived flows. The aim is to maximize network utilization while satisfying user's QoS requirements. We also have shown the performance results for end to end delay characteristic of optical hybrid switching. In particular, we have considered a flow classification scheme that is easy and cost-effective to implement in optical hybrid switching systems. Evaluation of the efficiency of hybrid switching compared with OCS and OBS is left as a topic for further study.

Acknowledgements. This work was supported in part by the Korea Science and Engineering Foundation (KOSEF) through the Ministry of Science and Technology (MOST) and Institute of Information Technology Assessment (IITA) through the Ministry of Information and Communication (MIC), Korea.

References

1. C. Qiao, M. Yoo.: "Choice, and Feature and Issues in Optical Burst Switching", *Optical Net. Mag.*, vol.1, No.2, Apr. 2000, pp.36-44.
2. Ilia Baldine, George N. Rouskas, Harry G. Perros, Dan Stevension.: "JumpStart: A Just-in-time Signaling Architecture for WDM Burst-Switching Networks", *IEEE Comm. Mag.*, Feb. 2002.
3. Gyu Myoung Lee, Bartek Wydrowski, Moshe Zukerman, Jun Kyun Choi, Chuan Heng Foh.: "Performance evaluation of optical hybrid switching system", *Proceedings of Globecom'2003*, vol.5, pp.2508-2512, December 2003
4. Chunsheng Xin, Chunmming Qiao, Yinghua Ye, Sudhir Dixit.: "A hybrid optical switching approach", *Proceedings of Globecom'2003*, vol.7. pp.3808-3812, December 2003.
5. S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, W. Weiss.: "An Architecture for Dif-ferentiated Services", RFC 2475, Dec 1998.
6. Panos Trimintzios, et al.: "A management and control architecture for providing IP differen-tiated services in MPLS-based networks", *IEEE Comm. Mag.*, pp.80-88, May 2001.
7. ITU-T Recommendation Y.1541.: "Network performance objectives for IP-based services", May 2002.
8. Gyu Myoung Lee, Jun Kyun Choi.: "Evolutional architecture of control plane for optical IP network", *Proceedings of COIN 2002*, pp. 25-27, July 2002.
9. Ken-ichi Kitayama, Kiyoshi Onohara, Masayuki Murata.: "Capability of optical code-based MPLS (OC-MPLS)", *Proceedings of. Sixth IFIP Working Conf. on Optical Network Design and Modelling Conference (ONDM 2002)*, Torino, Italy, Feb. 2002.
10. Chyan Yang, Chen-Hua Fu, Yueh-Heng Tu.: "Enterprise traffic with a differentiated service mechanism", *Inter. Journal of Network Management*, vol.11, pp. 113-128, 2001.
11. M. Dueser, I. de Miguel, P. Bayvel and D. Wischik.: "Timescale analysis for wavelength-routed optical burst switched (WR-OBS) networks", *Proceedings of OFC 2002*. March 2002.
12. Nada Golmie, et al.: "A differentiated optical services model for WDM networks", *IEEE Comm. Mag.*, p.p.68-73, Feb. 2000.
13. Hao Che, San-qi Li, Arthur Lin.: "Adaptive resource management for flow-based IP/ATM hybrid switching systems", *IEEE/ACM Trans on Networking*, vol. 6. no. 5. October 1998.

Delay Constraint Dynamic Bandwidth Allocation for Differentiated Service in Ethernet Passive Optical Networks

Lin Zhang, Lei Li, and Huimin Zhang

School of Information Engineering, Beijing University of Posts and Telecommunications, Beijing, China
Tel (FAX): 86-10-62283147
Zhanglin.bupt@gmail.com

Abstract. Ethernet Passive Optical Network (E-PON), which leverages the ubiquity of Ethernet at subscriber locations, seems destined for success in the optical access network. Dynamic Bandwidth Assignment (DBA) provides statistical multiplexing between the optical network units for upstream channel utilization. To satisfy the services with heterogeneous QoS characteristics, it is very important to provide QoS guaranteed network access while utilize the bandwidth efficiently. We propose a QoS-enabled DBA Algorithm for differentiated service in EPONs. The new DBA algorithm is based on the weights of the different classes and the current queue information to perform better per class bandwidth allocation. The specific QoS requirements of different classes are mapped into deterministic effective bandwidth and further used to assign the according weight. We conduct detailed simulation experiments to study the performance and validate the effectiveness of the proposed protocols.

1 Introduction

Ethernet Passive Optical Network (E-PON) [1] is considered to be one of the most cost-effective solutions for supporting the increased Internet data traffic, with the efficient bandwidth assignment function by which the upstream bandwidth can be shared among access users. A typical E-PON topology usually consists of an Optical Line Terminal (OLT) and N Optical Network Units (ONUs). All transmissions in a PON are performed between OLT and ONUs.

One distinguishing feature in EPON is Dynamic Bandwidth Assignment (DBA) [2], the ability to deliver services to emerging IP-based multimedia traffic with diverse quality-of-service (QoS) requirements. The basic concept of DBA replies on the possibility to allocate dynamically upstream bandwidth based on customers' real activity. Thus, bandwidth management for fair bandwidth allocation among different ONUs will be a key requirement for the MAC protocols in the emerging EPON based networks [3]. Diffserv [4] is an IETF framework for classifying network traffic into classes, with different service level for each class. In this propose, we discuss an EPON architecture that supports Diffserv. According to a multi-service access network, the proposed DBA algorithm in EPON should at least support a multitude of

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 350 – 357, 2005.

services, i.e., Low-Priority Best Effort service and Delay-Sensitive QoS guaranteed service.

Kramer et al in [1] provides a dynamic protocol called IPACT that is based on interleave polling to realize the dynamic bandwidth distribution. Although IPACT can provide the bandwidth "on-demand" according to end-users' queue length information, it has some difficulty to provide heterogeneous QoS guarantee to different end users. Another possible drawback is that IPACT considers the one ONU as a whole, which makes it difficult to realize different QoS access in one ONU. The authors of [8] proposed to use strict priority queuing and presented control message formats that handle classified bandwidth. However, no simulation results were reported to show the performance of their proposed DBA. In [9], the authors proposed a new DBA scheme that allocates the bandwidth according to the service level agreement between each subscriber. Typically, the model proposed in [9] will not be the case in future access network, where one single ONU must be capable of provisioning different QoS services for different user requirement.

In this paper, we propose a QoS-enabled Dynamic Bandwidth Assignment Algorithm for differentiated service in Ethernet Passive Optical Networks. The new DBA algorithm is based on the weights of the different classes and the current queue information to perform better per class bandwidth allocation. The specific QoS requirements of different classes are mapped into deterministic effective bandwidth and further used to assign the according weight. We conduct detailed simulation experiments to study the performance and validate the effectiveness of the proposed protocols.

2 Class and Queue Information Based Intra-ONU Scheduling and Inter-ONU Scheduling

We propose a QoS-enabled Dynamic Bandwidth Assignment Algorithm for differentiated service in Ethernet Passive Optical Networks. The overall goal of bandwidth allocation is to effectively and efficiently performs fair scheduling of timeslots between ONUs in EPON networks. A new DBA algorithm based on the weights of the different classes and the current queue information is considered at the OLT to perform better per class bandwidth allocation. These mechanisms include intra-ONU scheduling and inter-ONU scheduling as shown in Fig. 1.

We consider an EPON access network with N ONUs, which support M class of service (CoS). The transmission speed is R Mb/s. Assum $\square_m$ denotes the weight of the class of m, m=1,2,…, M; $\square_{nm}$ denotes the weight of the class of m at ONU n, ($\square_{nm} = \square_m$ for n=1,2,…,N; m=1,2,…,M); q_{nm} denotes the current queue length of class m at ONU n, n=1,2,…,N; m=1,2,…,M; so the granted bandwidth B_{nm} to class m at ONU n can be expressed at follows,

$$B_{nm} = \min\{q_{nm}, \frac{\square_{nm}}{\square_{n,m} \square_{nm}} \cdot R\} \tag{1}$$

In the case of $\frac{\phi_{nm}}{\sum_{n,m}\phi_{nm}} \cdot R > q_{nm}$, which the allocated bandwidth according to the weight cannot be consumed by the class m at ONU n, the excess bandwidth will be further allocated to those class which still has data in the queue. This scheme shifts the complexity of the queue management of the ONU to the OLT and realizes a better fairness among the different classes at different ONU. Fig 2 shows a simple case of the mentioned scheme.

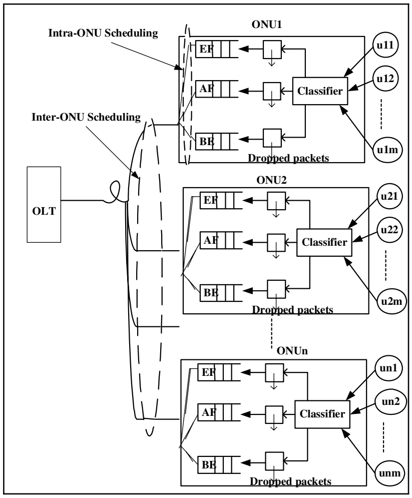
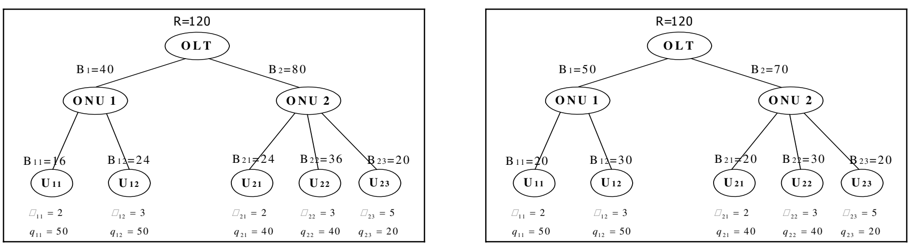


Fig. 1. Inter-ONU and Intra-ONU scheduling



(a) Weight based DBA plus ONU based queue management

(b) Weight and queue length based DBA

Fig. 2. Comparison between two different DBA

3 Delay and Jitter Guaranteed Dynamic Bandwidth Assignment

In this section we present a new scheme called Deterministic Effective Bandwidth-based Generalized Processor Sharing scheduler (DEB-GPS scheduler), in which the specific QoS requirement is mapped into deterministic effective bandwidth and further used to assign the according weight in GPS scheduler. Our proposed scheme can provide delay-constraint and loss-less QoS guarantee to QoS service and maximize the bandwidth to best-effort service. We have proved that our DEB-GPS scheduler requires less bandwidth than rate-based GPS scheduler to provide the same QoS guarantee. In order to further improve the efficiency of proposed algorithm, the queue length information of each best-effort source is reported to OLT to calculate the backlog clearing time. As each session completes its backlog, the bandwidth released will be distributed among the still backlogged sessions in proportion to their weights. And the service rates of backlogged sessions can be increased as more and more sessions are completing their backlogged periods.

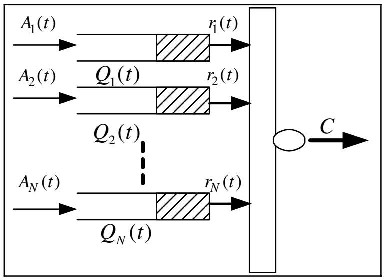


Fig. 3. Fair bandwidth allocation model

Below we will define the parameters used in the problem formulation.

- $A_i(t)$ denotes the incoming traffic of ith stream that is constrained by leaky bucket scheme (M_i, P_i, R_i), which are respectively the maximum burst size, the peak rate, and the mean rate of the source. So the incoming traffic is constrained by $A_i(t) \square Min(P_i t, R_i t + M_i \cdot {}^{(P_i \square R_i)}\!/\!_{R_i})$.
- $\square_i$ denotes the weight of ith stream.
- $W_i(t)$ denotes the amount of ith stream traffic transferred by the server.
- $Q_i(t)$ denotes the queue length of ith stream. A stream is called backlogged if there is always traffic queued for that stream.
- r_i denotes the allocated rate of ith stream from the server.
- D_i and δ_i be the delay and delay variation tolerances of ith stream, respectively.
- d_i and $\square_i$ be the experienced delay and delay variation of ith stream, respectively.
- C denotes the available bandwidth from the server

We formulate the problem of fair bandwidth allocation as a single server serves N traffic streams (Fig. 3). We consider different QoS services, which imply that the delay and delay variation tolerances for different streams are different from each other. We also Assum all the buffers are infinite.

Our traffic management scheme is based on deterministic QoS guarantees. Consider a access system that serves a flow in a work conserving manner at a constant rate, the deterministic effective bandwidth of an incoming traffic $A_i(t)$ is defined as a constant rate $e_D(A_i(t))$ that guarantees a delay bound of Di to this flow, that is:

$$e_D(A_i(t)) = \sup_{t \geq 0}\left(\frac{A_i(t)}{t+D_i}\right) \tag{2}$$

So the deterministic effective bandwidth $e_D(A_i(t))$ that guarantees a delay bound of Di to the above incoming traffic is defined as follows:

$$e_{D_i}(A_i(t)) = \begin{cases} \frac{M_i}{(D_i+\frac{M_i}{P_i})} & if \quad 0 \leq D_i \leq M_i \cdot (\frac{1}{R_i} - \frac{1}{P_i}) \\ R_i & If \quad D_i \geq M_i \cdot (\frac{1}{R_i} - \frac{1}{P_i}) \end{cases} \tag{3}$$

Since our traffic management scheme guarantee a delay-constrained and loss-less deterministic QoS requirement to QoS services, a deterministic effective bandwidth is provided to every QoS source. This is done by using the GPS scheduler. The scheduler works as follows: different types of incoming sources are mapped into QoS-aware source and best-effort source. Their respective weights ϕ_{QoS} and ϕ_{BE} are defined as follows once a source has been accepted into the system:

$$\phi_{QoS_i} = e_{D_i}(A_i(t))\, and \quad \phi_{BE} = \frac{1}{N_{BE}}(C - \sum_i A_i(t)) \tag{4}$$

in which N_{BE} equals to the number of total Best-Effort sources and C is the available service rate. With such weight assignment, each QoS-aware source receives a minimum service rate equal to its deterministic effective bandwidth. This ensures that the QoS-aware source experiences a delay-constrained and loss-less deterministic QoS service.

4 Architecture of Dual DEB-GPS Scheduler and Its Operation for Dynamic Bandwidth Allocation

Compared to traditional DBA algorithm that only OLT schedules upstream access for ONU, our proposed dual-scheduler scheme allows both OLT and ONU to participate in bandwidth allocation process, in which we have implemented a two-layer multiplexing scheme. For OLT part, requests from different QoS classes are gathered from different ONUs and multiplexed to provide the bandwidth allocations among QoS classes. For ONU part, ONU can further select the proper bandwidth or grants allocated to it from OLT based on its own queuing status and QoS contract, as shown in Fig. 4.

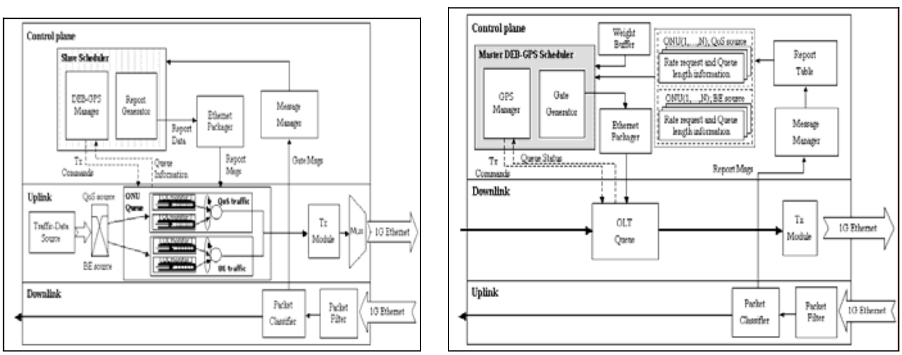
The following is an operation mechanism of the proposed dual scheduler;

Step1: Master scheduler in OLT periodically allocates divided slot grants to ONUs to collect the rate request from ONUs.
Step2: After receiving the connection request from ONU, the master scheduler in OLT will:

1. Updates the request table and sums up all requests of each class.
2. Fixed bandwidth is assigned by the value of deterministic effective bandwidth of QoS-aware source.
3. Recalculate the weight assignment to current best-effort sources, and the surplus bandwidth is allocated to best effort sources by their re-calculated weight.

Step3: In this step, slave scheduler in each ONU schedules its own cells by using the grant numbers that have been delivered from OLT.

1. The arrival and predicted service time stamp of each packet of each source is calculated by the local GPS scheduler according to the arrival function and the allocated bandwidth.
2. Compute the delay of each source and sort this delay in decrease order.
3. Compare the variation of the first and last delay in this sorted line with the predefined delay variation tolerance □:
   A) if no violation, each source will get their allocated bandwidth;
   B) if this □ is violated, the allocated bandwidth of the first and last sources will be re-allocated as the arithmetic mean of their former allocated bandwidth.
4. Repeat (2)-(3) until no delay variation is violated or the delay bound is violated.



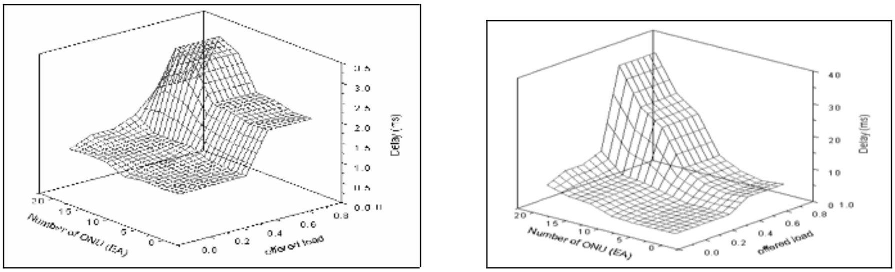
(a) ONU (b) OLT

Fig. 4. Proposed Architecture of ONU and OLT

5 Simulation Results and Analysis

In this section, we demonstrate the properties of our proposed scheme by simulation results. In the simulation, we assume a random Round Trip Time (RTT) from OLT and each ONUs and we increase the number of ONUs from 1 to 24. The transmission rate of user access link to each ONU is set to 100 Mb/s, and the upstream link rate to be 1000 Mb/s. We further change the access rate of traffic from 10 Mb/s to 90 Mb/s, which equals to the change of offered load from 0.1 to 0.9 compared to the transmission rate of the access link of each ONU. In order to demonstrate the properties of our proposed dual DEB-GPS scheduler, we define two classes of incoming traffic as QoS-aware source and BE sources. The ratio of offered load between QoS services and BE services in an ONU is set to 0.3:0.7.

We demonstrate the average packet delay for different services of our proposed scheme as a function of an ONU's offered load and the number of ONU in Fig. 5. We find that our proposed scheme can provide different access services to different sources. QoS services meet very low average delay, and the delay performance of BE sources also show reasonably good performance even at high load. As the offered load increases, the experienced delay of BE source has notable increase compared to that of QoS service.

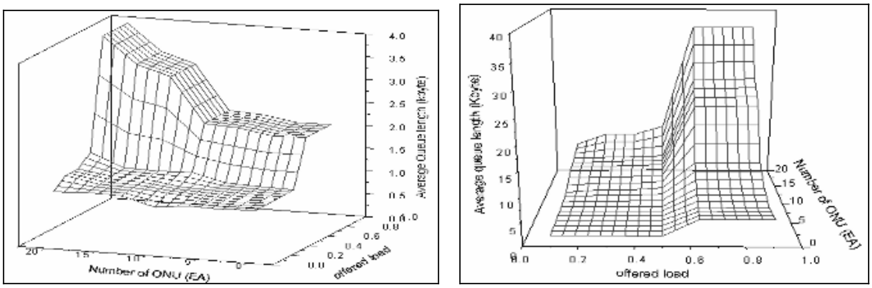


(a) QoS (b) BE

Fig. 5. Average Delay of QoS and BE service

We are also interested in the performance of best-effort service as a function of the offered load and the number of ONUs in the whole network shown in Fig.5 (b). When the number of ONUs is low, all packets of BE service meet a very little delay, no matter what the ONU's offered load is. This is because our proposed scheme implements a slave scheduler at ONU part to re-allocate the bandwidth according to the queue length and clearing time of each source, which improves the bandwidth utility and the delay performance of best-effort services. We also notice that increasing the number of ONUs yet keeping the offered load of each ONU low may result in a higher delay to BE service. The reason is that the allocated bandwidth to each ONU is decided by OLT in advance, some unused bandwidth in one ONU cannot be shared by other services of different ONUs.

Fig.6 shows the average queue length under the number of ONUs and offered traffic load. From this Figure, we can see the varying characteristics of the different traffic source type when the offered load in each ONU and the number of ONU in an EPON change. We find that our proposed scheme can guarantee the requested QoS services while keeping the queue length of best-effort services to an acceptable range. This characteristic shows up more clearly as the number of ONUs gets bigger. The queue length of QoS service is small generally. Also, the queue length of BE service remains small when the number of ONUs is small. This is because of the relative abundant bandwidth. However, as the number of ONUs gets larger, the queue length of best-effort source also gets longer. It means that the proposed scheme provides the guaranteed bandwidth for QoS service.



(a)QoS (b) BE

Fig. 6. The Average Queue length under the incoming traffic type

6 Conclusion

The E-PON system offers an economical solution to the provision of broadcast services and high-speed data communications services. Recently one of the issues in E-PON access network is how to design a DBA algorithm to use the limited bandwidth efficiently and at the same time keeping the characteristics of traffic contracts. In this paper, an efficient dynamic bandwidth allocation based on Generalized Processor Sharing (GPS) scheduler is presented and the measured performance of an E-PON system with the proposed algorithm is demonstrated under different traffic parameters using computer simulations.

References

1. G. Kramer, B. Mukherjee, and A. Maislos, “Ethernet Passive Optical Network (EPON): a missing link in an end-to-end optical internet,” in Multi-Protocol Over WDM: Building the Next Generation Internet, S. Dixit, Ed. New York: Wiley, Mar. 2003.
2. C. M. Assi et al., “Dynamic Bandwidth Allocation for Quality-of-Service over Ethernet PONs,” IEEE JSAC, vol. 21, no. 9, pp. 1467–77, Nov. 2003.
3. Michael P. McGarry et al., “Ethernet PONs: A survey of dynamic bandwidth allocation (DBA) algorithms”IEEE Optical Communications pp. s8–s15, August 2004.
4. S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, An Architecture for Differentiated Services, IETF, RFC 2475, Dec. 1998.
5. Virtual Bridged Local Area Networks, IEEE Standard 802.1Q, 1998.
6. Media Access Control Parameters, Physical Layers and Management Parameters for Subscriber Access Networks, IEEE Draft P802.3ah/D1.0TM, Aug. 2002.
7. W. Willinger, M. S. Taqqu, and A. Erramilli, “A bibliographical guide to self-similar traffic and performance modeling for modern high-speed networks,” in Stochastic Networks. Oxford, U.K.: Oxford Univ. Press, 1996, pp. 339–366.
8. S. Choi and J. Huh, “Dynamic bandwidth allocation algorithm for multimedia services over Ethernet PONs,” ETRI J., vol. 24, no. 6, pp. 465–468, Dec. 2002.
9. M. Ma, Y. Zhu, and T. H. Cheng, “A bandwidth guaranteed polling MAC protocol for Ethernet passive optical networks,” in Proc. IEEE INFOCOM, San Francisco, CA, Mar.–Apr. 2003, pp. 22–31.

An Architecture for Efficient QoS Support in the IEEE 802.16 Broadband Wireless Access Network

Dong-Hoon Cho, Jung-Hoon Song, Min-Su Kim, and Ki-Jun Han*

Department of Computer Engineering, Kyungpook National University, Korea
{firecar, pimpo, kiunsen}@netopia.knu.ac.kr
kjhan@bh.knu.ac.kr

Abstract. In this paper, we propose a new QoS architecture for the IEEE802.16a MAC protocol and present a bandwidth allocation and admission control policy for the architecture. Our architecture provides QoS support to real-time traffic with high priority while maintaining throughput performance to an acceptable level for low priority traffic. Analytical and simulation results assure advantages of our architecture.

1 Introduction

The emerging 802.16e and 802.20 standards will both specify new mobile air interfaces for wireless broadband. On the surface the two standards seem very similar, but there are some important differences between them. For one, 802.16e will add mobility in the 2 to 6 GHz licensed bands, while 802.20 aims for operation in licensed bands below 3.5GHz. More importantly, the 802.16e specification will be based on an existing standard (802.16a). while 802.20 is starting from scratch. This means that products based on 802.16e will likely hit the market well before 802.20 solutions. The IEEE approved the 802.16e standards effort in February with the avowed intent of increasing the use of broadband wireless access (BWA) by taking advantage of the "inherent mobility of wireless media." The amendment to 802.16, which is also called the wireless metropolitan area network (WMAN) standard, will enable a single base station to support both fixed and mobile BWA. It aims to fill the gap between high data rate wireless local area networks (WLAN) and high mobility cellular wide area networks (WAN).

However, IEEE 802.16 standard left the QoS based packet-scheduling algorithms, which determine the uplink and downlink bandwidth allocation, undefined. This paper proposes an efficient QoS architecture, based on priority scheduling and dynamic bandwidth allocation. The system performance is analytically evaluated and is verified through a simulation.

The remaining of this paper is organized as follows. Section 2 reviews the BWA Systems and IEEE 802.16 MAC Protocol. In section 3, we describe the existing IEEE802.16 QoS architecture. We present a new QoS architecture for QoS support to real-time traffic with high priority while maintaining throughput performance to an acceptable level for low priority traffic in section 4. Section 5 provides simulation results of our QoS architecture and we conclude in Section 6.

---

* Correspondent author.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 358 – 365, 2005.

2 BWA Systems and IEEE 802.16 MAC Protocol

IEEE 802.16 architecture consists of two kinds of fixed (non-mobile) stations: subscriber stations (SS) and a base station (BS). The communication path between SS and BS has two directions: uplink channel (from SS to BS) and downlink channel (from BS to SS). The downlink channel, defined as a direction of data flow from the BS to the SSs, is a broadcast channel, while the uplink channel is a shared by SSs. Time in the uplink channel is usually slotted (mini-slots) called by time-division multiple access (TDMA), whereas on the downlink channel BS uses a continuous time-division multiplexing (TDM) scheme as shown in Fig. 1. [2]

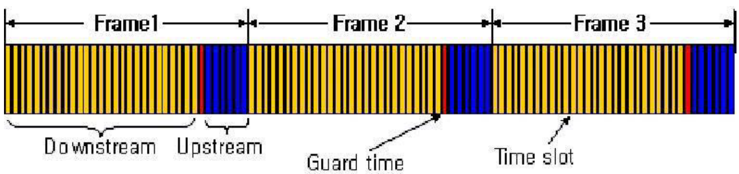


Fig. 1. IEEE 802.16 TDD frame structure

The BS dynamically determines the duration of these subframes. Each subframe consists of a number of time slots. SSs and BS have to be synchronized and transmit data into predetermined time slots. To support QoS, IEEE 802.16 defines four QoS services: Unsolicited Grant Service (UGS); Real-Time Polling Service (rtPS); Non-Real-Time Polling Service (nrtPS) and Best Effort (BE) service. UGS service is prohibited from using any contention requests, there is no explicit bandwidth requests issued by SS. The BS must provide fixed size data grants at periodic intervals to the UGS flows. The rtPS and nrtPS flows are polled through the unicast request polling. However, the nrtPS flows receive few request polling opportunities during network congestion and are allowed to use contention requests, while the rtPS flows are polled regardless of network load and frequently enough to meet the delay requirements of the service flows.

3 QoS Architecture for IEEE 802.16 MAC Protocol

In IEEE 802.16 standard, there are two modes of transmitting the BW-Request: contention mode and contention-free mode (polling). In contention mode, SSs send BW-Request during the contention period. Contention is resolved using back-off resolution. In contention-free mode, BS polls each SS and SSs reply by sending BW-request. Due to the predictable signaling delay of the polling scheme, contention-free mode is suitable for real time applications. IEEE 802.16 defines the required QoS signaling mechanisms described above such as BW-Request and UL-MAP, but it does not define the Uplink Scheduler, i.e. the mechanism that determines the IEs in the UL-MAP.

Fig. 2 shows the existing QoS architecture of IEEE 802.16. Uplink Bandwidth Allocation scheduling resides in the BS to control all the uplink packet transmissions.

Since IEEE 802.16 MAC protocol is connection oriented, the application first establishes the connection with the BS as well as the associated service flow (UGS, rtPS, nrtPS or BE). BS will assign the connection with a unique connection ID (CID). The connection can represent either an individual application or a group of applications such as multiple tenants in an apartment building (all in one SS) sending data with the same CID. [1]

IEEE 802.16 defines the connection signaling (connection request, response) between SS and BS but it does not define the admission control process. All packets from the application layer in the SS are classified by the connection classifier based on CID and are forwarded to the appropriate queue. At the SS, the Scheduler will retrieve the packets from the queues and transmit them to the network in the appropriate time slots as defined by the UL-MAP sent by the BS. The UL-MAP is determined by the Uplink Bandwidth Allocation Scheduling module based on the BW-request messages that report the current queue size of each connection in SS. [1]

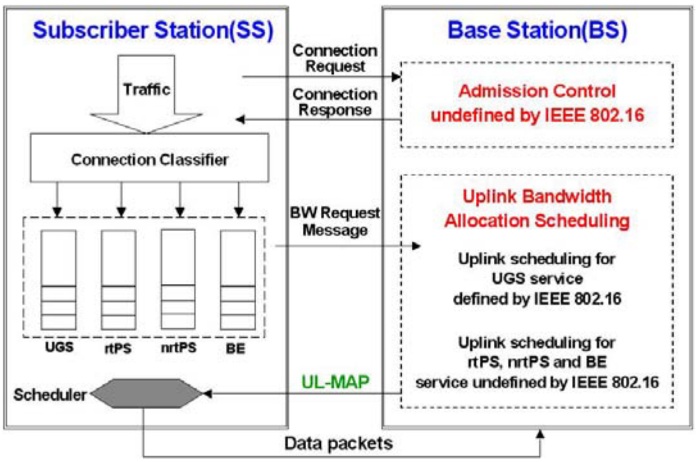


Fig. 2. QoS architecture of IEEE 802.16

4 A New QoS Architecture of IEEE 802.16

Now, we propose a QoS architecture that completes the missing parts in the IEEE 802.16 QoS architecture. As shown in Fig 3, at the BS we add a detailed description of the Uplink Bandwidth Allocation Scheduling part (scheduling algorithm that which supports all types of service flows), and admission control part. At the SS we add a traffic management module.

For each of the UGS, rtPS, nrtPS, BE service, multiple connections are aggregated into their respective service flow. The schedule process is divided into two steps. The first step is performed at the BS according to the information of the request from the SS. Then the uplink scheduler of SS is responsible for selection of appropriate packets from all queues and sends them through the uplink data slots granted by the Packet

Allocation Module of BS. The BS must provide fixed size data grants at periodic intervals to the UGS flows

Here is a brief description of the connection establishment using the QoS architecture in Fig 3:

(1) An application that originates at an SS establishes the connection with BS using connection signaling. The application includes in the connection request the traffic contract (bandwidth and delay requirement).
(2) The admission control part at the BS accepts or rejects the new connection.
(3) If the admission control part accepts the new connection, it will notify the Uplink Bandwidth Allocation Scheduling part at the BS and provide the token bucket parameters to the traffic management module at the SS.

After the connection is established, the following steps are taken:

(1) Traffic management enforces traffic based on the traffic contract of the connection.
(2) At the beginning of each time frame, the data packet analysis module collects the queue size information from the BW-requests received during the previous time frame. The data packet analysis module will process the queue size information and update the traffic management table.
(3) The packet allocation module retrieves the information from the traffic management module and generates the UL-MAP.
(4) BS broadcasts the UL-MAP to all SSs in the downlink subframe.
(5) The scheduler of SS transmits packets according to the UL-MAP received from the BS.

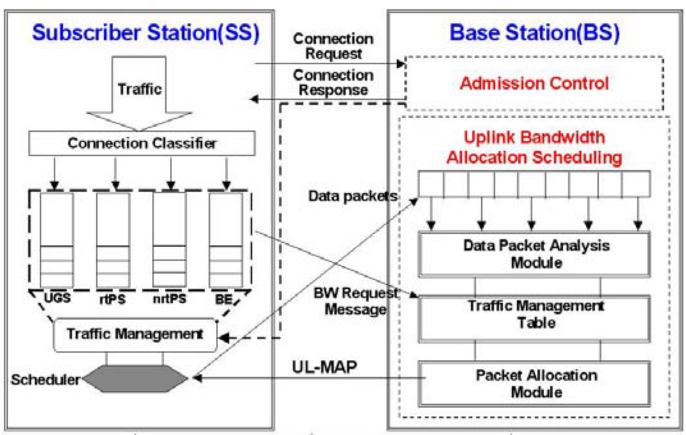


Fig. 3. Proposed QoS architecture of IEEE 802.16

4.1 Uplink Bandwidth Allocation Scheduling

After SSs transmit UGS packets by uplink data slot, Data Packet Analysis Module (DPAM) of BS separates UGS data and virtual packets arrival time of rtPS. This information manages at Traffic Management Module and uses the Polling schedule in next frame. The rtPS is time-bounded data. Therefore, BS is apt to give Poll to SS coming up close to deadline. In this work, we assume BS has the ability to detect collision in each contention period mini-slot. The BS broadcasts a common back-off window size "B" to all the competing SSs. SSs will then randomly choose a reservation slot numbered between 1 and B to transmit its request.

We assume that there are N SSs in the system and BS broadcasts a back-off window size B. Since each user will choose between 1^{st} and B^{th} reservation slots to send its bandwidth reservation, the probability of choosing a given slot is p=1/B. As a result, the probability of a given slot that is not selected by any SS is given by:

$$PS_{NS} = (1-p)^N \tag{1}$$

The probability of a successful transmission is equal to the probability that a single user selects a given slot. Thus, the system throughput is given by:

$$P_{th} = Np(1-p)^{N-1} \tag{2}$$

To maximize system throughput, we have to get:

$$\begin{aligned} &\frac{dP_{th}}{dp} = N(1-p)^{N-1} - N(N-1)p(1-p)^{N-2} = 0 \\ &p = \frac{1}{N} \\ &\therefore\ p = \frac{1}{B} = \frac{1}{N} \ \Rightarrow\ \mathrm{N = B} \end{aligned} \tag{3}$$

In other words, the maximum throughput can be obtained when BS broadcasts a back-off window size (B) which is equal to the number of competing SSs (N).

4.2 Channel Utilization for Data Flow of IEEE 802.16 MAC Protocol

Here, we find out an analytical model for channel utilization. We assume that there are *k* classes of priority queues: Class 1 is the highest priority traffic, and class 2 is the second highest priority traffic, and class *k* means the lowest priority. We also assume that arrival events are mutually independent. Let *C* and denote the server capacity and channel utilization for each class *i*, respectively. Then we have

$$C = \rho_1 + \rho_2 + \ldots + \rho_k + \mathrm{D_{con}}\ ,\quad \mathrm{C} \le 1 \tag{4}$$

In our scheme, the higher priority class is allocated the bandwidth first, and then the lower priority class is allocated the remaining bandwidth late. For example, the capacity of class 2 uses the remainder of capacity left over class 1. Similarly, the server allocates the remainder of capacity to class 4 after class 1, 2 3 are allocated. So, we have

$$\rho_1 = \lambda_1 E[\tau_1] \tag{5a}$$

$$\rho_2 = \begin{cases} \lambda_2 E[\tau_2] \ , \ (1-\rho_1-\lambda_2 E[\tau_2])>0 \\ (C-\rho_1) \ , \ (1-\rho_1-\lambda_2 E[\tau_2])<0 \end{cases} \tag{5b}$$

$$\rho_3 = \begin{cases} \lambda_3 E[\tau_3] \ , \ (1-\rho_1-\rho_2-\lambda_3 E[\tau_3])>0 \\ (C-\rho_1-\rho_2) \ , \ (1-\rho_1-\rho_2-\lambda_3 E[\tau_3])<0 \end{cases} \tag{5c}$$

$$\rho_k = \begin{cases} \lambda_k E[\tau_k] \ , \ (1-\rho_1-\ldots-\lambda_k E[\tau_k])>0 \\ (C-\rho_1-\ldots-\rho_{k-1}) \ , \ (1-\rho_1-\ldots-\lambda_k E[\tau_k])<0 \end{cases} \tag{5d}$$

where λ_k is offered load for class k and $E[\tau_k]$ is service time for class *k*. Using these equations, we can get channel utilization for each class of priority traffic.

5 Simulations

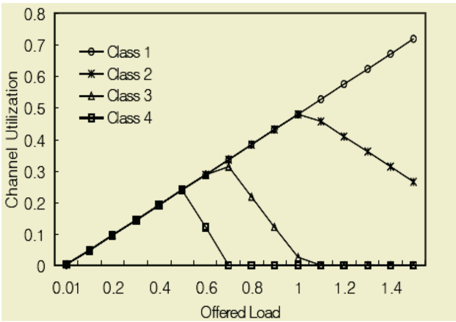
In this section, we evaluate performance of our scheme for IEEE 802.16. The system model for analysis consists of one base station and numbers of subscriber stations (SS). In addition, each SS is assumed to be a Poisson traffic source and the packet size (including overhead) is variable. The parameters used for performance evaluation are listed in Table 1.

Table 1. Simulation Parameter

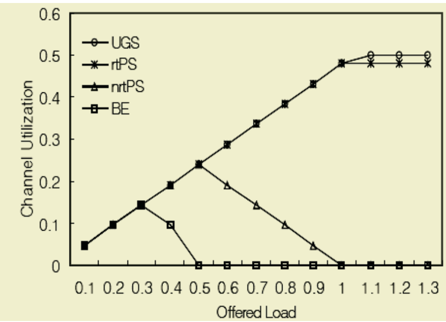
Meaning	Value(802.16)
Number of SS	20
Preamble(beacon)	3us
Each MAP	5us
Downlink DATA	8us
Register Contention(RC)	1us
Contention Period	Number of active station
Average data packet size	Each traffic 100~200byte
Fixed frame size	1ms
PHY rate	50Mbps

Figure 4 shows channel utilization obtained by simulation experiments and the analytical model given by Eq. (5a)~(5d). Fig. 5(a) shows channel utilization when we assume that there is no limit of the bandwidth that the highest priority traffic can take. On the other hand, Figure 4(b) shows channel utilizations when a fixed quota is allowed for the UGS flow and the remaining bandwidth is used for the other three flows. We can see that the UGS flows do not increase any more above some value because the BS provides fixed size data grants to the UGS flows at periodic intervals.

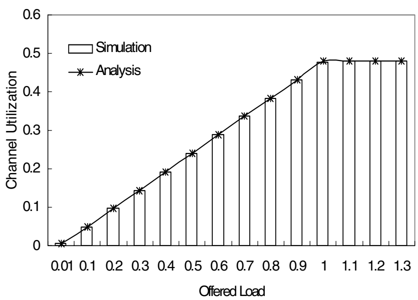
Fig. 4(c)~4(f) compares analytical and simulation results of channel utilization for four different priorities of packets. This figure indicates that our analytical model is simple, nevertheless accurate. The channel utilization of the high priority traffic increases linearly because it is not affected by the transmission of lower priority traffic.



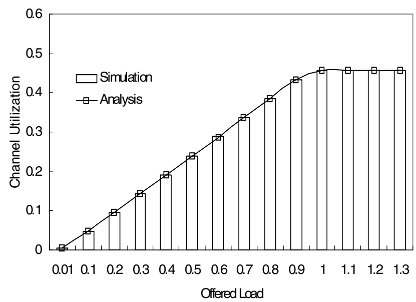
(a) Channel utilization when there is no limit of the bandwidth that the highest priority traffic can take



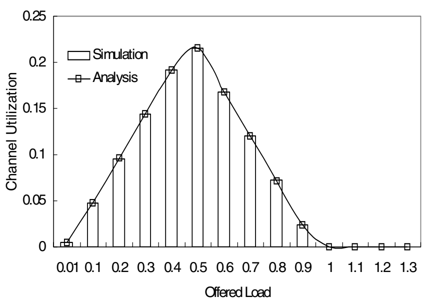
(b) Channel utilization when only a fixed quota is allowed for the UGS flow



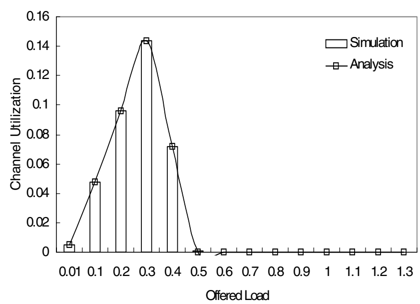
(c) Channel utilization of UGS flows (analytical and simulation results)



(d) (d) Channel utilization of rtPS flows (analytical and simulation results)



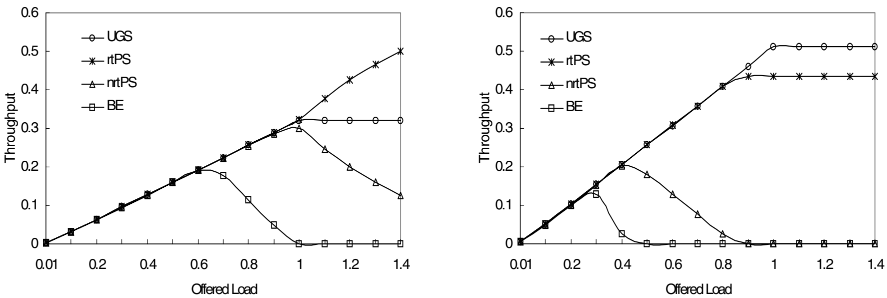
(e) Channel utilization of nrtPS flows (analytical and simulation results)



(f) Channel utilization of BE flows (analytical and simulation results)

Fig. 4. Channel Utilization

Fig. 5 shows throughput for four different types of packet with various packet sizes. In this figure, we can see that the same maximum throughput can be obtained by selecting proper packet size of UGS and rtPS flows. Because the nrtPS and BE flows are low priority classes of traffic, they are affected by the UGS and BE flows. The high priority traffic is constantly allocated uplink data slots granted by BS.



(a) When average packet size = 100byte **(b) When average packet size = 160byte**

Fig. 5. Throughput with different packet sizes

6 Conclusions

In this paper, we have proposed a new QoS architecture for IEEE 802.16 broadband wireless access MAC protocol. We also presented a bandwidth allocation and admission control policy for the architecture. The simulation and analytical results show that our architecture may provide QoS support in terms of bandwidth request and allocation for all type of traffic classes.

Acknowledgement. This research is supported by Program for the Training of Graduate Students for Regional Innovation.

References

[1] Kitti Wongthavarawat and Aura Ganz, "Packet scheduling for QoS support in IEEE 802.16 broadband wireless access systems" *Military Communications Conference, IEEE 2003*

[2] GuoSong Chu, Deng Wang and Shunliang Mei "A QoS architecture for the MAC protocol of IEEE 802.16 BWA system", *Communications, Circuits and Systems and West Sino Expositions, IEEE 2002*

[3] IEEE 802.16 Standard {Local and Metropolitan Area Networks} Part 16: Air Interface for Fixed Broadband Wireless Systems, *IEEE P802.16/D3-2001*

[4] Cao Y, Li VOK. "Scheduling algorithms in broad-band wireless networks" *Proceeding of the IEEE 2001*

[5] IEEE Standard for Local and metropolitan area networks-Part 16: Air Interface for Fixed Broadband Wireless Access Systems-Amendment 2:Medium Access Control Modification and Additional Physical Layer Specifications for 2-11GHz, *IEEE Standard 802.16a-2003*

A Pragmatic Methodology to Design 4G: From the User to the Technology

Simone Frattasi[1], Hanane Fathi[1], Frank Fitzek[1],
Marcos Katz[2], and Ramjee Prasad[1]

[1] Center for TeleInFrastruktur (CTIF), Aalborg University,
Niels Jernes Vej 12, 9220, Aalborg, Denmark
{sf, hf, ff, prasad}@kom.aau.dk
[2] Samsung Electronics, Co. LTD
marcos.katz@samsung.com

Abstract. The ever-increasing growth of user demands, the limitations of the *Third Generation of Mobile Communication Systems* (3G) and the emergence of new mobile broadband technologies on the market have brought researchers and industries to a throughout reflection on the *Fourth Generation* (4G). Many prophetic visions have appeared in literature presenting the future generation as the ultimate boundary of the wireless mobile communication without any limit in its potential, but practically not giving any designing rules and thus any definition of it. In this paper we hence propose a new user-oriented methodology that considers the user as "the angular stone in the design of 4G" and identifies his functional needs and expectations, reflecting and illustrating them in everyday life situations. In this way, we devise fundamental user scenarios where new services are significant assets for the user. The latter implicitly reveal the key features of 4G, which are then explicated in a new framework – the "user-centric" system – that, through a satellite hierarchical vision, describes the various level of interdependency among them. This approach consequently brings to the identification of the designing rules and therefore to a more pragmatic definition of 4G. Finally, an example of a new 4G application is also given in order to demonstrate the validity of the overall methodology.

1 Introduction

Following the paradigm of generational changes, it was originally expected that the *Fourth Generation* (4G) would follow sequentially after 3G and emerge between 2010 and 2015 as an ultra-high speed broadband wireless network [1]. In Asia, for example, the Japanese operator NTT DoCoMo introduced the concept of MAGIC for defining 4G [2], which mainly focuses on public systems and treats 4G as the extension of 3G cellular service. The latter is in general the main tendency also in China and South Korea. This view is hence referred to as the "linear 4G vision" and, in essence, is about a future 4G network that provides very high data rates (exceeding 100 Mbit/s), which will be deployed several

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 366–373, 2005.

years after 3G has become commercially available on a large scale. Additionally, it is expected that these 4G networks will enable seamless interoperability and interconnection with other mobile devices. This Asian vision assumes that network will generally have a cellular structure, which builds on the fundamental architecture of preceding generations of mobile technologies.

However, even if 4G is named as the successor of previous wireless communication generations, the future is not limited to cellular systems and thus 4G has not to be exclusively understood as a linear extension of 3G [3]. In Europe, for example, the *European Commission* (EC) envisions that 4G will ensure seamless service provisioning across a multitude of wireless systems and networks, from private to public, from indoor to wide area, and provide an optimum delivery via the most appropriate (i.e., efficient) network available. From the service point of view, it foresees that 4G will be mainly focused on personalized services [4]. Therefore, it emphasizes the heterogeneity and integration of networks and new service infrastructures, rather than increased bandwidth "per se". This view is referred to as the "concurrent 4G vision" and takes also into account technologies that are currently emerging and that may either complement or compete with 3G. While 2G was focused on full coverage for cellular systems offering only one technology and 3G provides its services only in dedicated areas and introduces the concept of vertical handover through the coupling with *Wireless Local Area Network* (WLAN) systems, 4G will be a *convergence platform* extended to all the network layers. Moreover, in order to boost innovation and define and solve relevant technical problems, the system level has to be envisioned and understood with a broader view, taking the user as the departing point. This *user-centric approach*, developed further in this paper, can result in a beneficial method for identifying innovation topics at all the different protocol layers and avoiding a potential mismatch in terms of service provisioning and user expectations.

There is clearly a need for a methodological change in the design of the next wireless communication generation. Therefore, in this paper we propose a methodology based on a top-down approach, which starts from the user and his functional needs reflected in everyday life situations[1]. As a consequence, new user scenarios have been identified in order to demonstrate that the new services are significant assets for the user. These will then reveal the key features of 4G, leading to the definition of a new framework – the "user-centric" system – that shows the direct correspondence relation among them. This is certainly a fundamental starting point in order to derive the designing rules and therefore a less prophetic and more pragmatic definition of 4G. Finally, a mapping to technical features and improvements is done to support the needed services.

The rest of the paper is organized as follows: Section 2 introduces the new methodological approach; Section 3 defines the envisioned user scenarios and

[1] Since each and every user is unique and has different needs depending on his profession, social condition, geographical location, habits, etc., it is difficult to define user needs in a generic fashion. Therefore, the issue has to be addressed for each group of users. For heuristic purposes, we focus on the average user in the western society.

the relevant services; Section 4 extrapolates, interrelates and describes the key features of 4G. Finally, the concluding remarks are given in Section 5.

2 From the User to the Technology

Instead of being only something that people use for task completion, communication technologies have become something that people live with, an integral part of everyone's life. In fact, their usage cannot be separated from the rest of peoples' lives and examined under a microscope as an isolated object. So far, the designers of the new technology have not enough considered the world for which they are designing. Indeed, in a broader context, developing technology for technology is meaningless even for the telecom industries, since they will most likely not get paid back for their initial investments. Therefore, it appears more logical and less risky to set a goal to develop technology in order to provide (and sell) new services to the user. From this point of view, the user or the "wireless person" is the main actor playing on the stage of the wireless world and he is unaware of and indifferent about the technology to use in order to get some desired service. Therefore, if we consider his requirements secondary with respect to the technological issues the risk is to face some incalculable failure (e.g., *Wireless Application Protocol* (WAP)). In fact, without a broad horizon obtained through an extended overview of the general problem and with just the limited and narrow point of view of the technology, no one is able to predict the level of acceptance and penetration in the market of a given technology or product. Needless to say, huge investments and enormous efforts by industry and academia may eventually be wasted. Thus, it becomes crucial to understand the user, his expectations and needs; and to consider him as the "angular stone" in the design of the new technology in order to turn 4G into a big success. Besides, it has also to be taken in consideration that novel technologies may have a significant (and unpredictable) impact on user's behaviour, and consequently their usage will then change the emerging products. So, understanding the user means understanding how he changes as the society around him changes in general, and specifically how he changes through the interaction with the products that are introduced. In particular, if technological developers start from understanding human needs, they are more likely to accelerate evolutionary development of useful technology. The pay-off from a technology innovation is that it supports some human needs while minimizing the down-side risks. Therefore, responsible analysis of technology opportunities will consider positive and negative outcomes, thus amplifying the potential benefits for society [5]. Clearly, there is a need for a new approach; there is a need for contextual understanding; there is a major methodological challenge in the design of the next generation of communication technologies.

The methodology we propose here is a top-down approach that focuses on a user-centric vision of the wireless world and consists in the following four steps:

1. It starts first from the user as a socio-cultural person with subjective preferences and motivations, cultural background and customs and habits. This leads to the identification of the user's functional needs and expectations in terms of services and products[2].
2. The functional needs are reflected and illustrated in everyday life situations, where the new services are significant assets for the user. This way, fundamental but exemplary user scenarios can be extrapolated from sketches of people's everyday life.
3. Key features can be extracted from the user scenarios assessed in the previous step. They are the basic pillars for a very relevant and pragmatic definition of the 4G technology.
4. The last step concerns the definition of the technical means related to the features outlined in Step 3. A mapping to technical features and improvements must be done to support the requirements of the different user scenarios defined in Step 2.

3 New 4G User Scenarios

3.1 Business On-the-Move

Even before leaving home to reach the place of a work appointment, the user would like to receive information about train/subway schedules, door-to-door delays, etc., and more personalized ones, such as knowing how long it takes walking to get on the first schedules, in order to eventually wait for the next train. According to the user's decisions, his time-plan must be consequently scheduled in the most efficient way. During his stay on the train, the user would like to download e-mails, listen to the radio, watch the TV, etc. (the environment also enforces the range of applications the user can exploit. For example, if we take into account the daily trip to work which is not longer than one hour – for instance, the distance to go from the suburbs of Paris to the center of the city itself – applications like movies on demand cannot be taken into consideration). Finally, before he will get off from the last planned train the most time-saving exit and the way to reach his final destination must be known and available in audio and/or video format.

3.2 Smart Shopping

The user would like to receive pop-ups informing him of some offer not only when passing by or through a shopping mall, but also anywhere else (e.g., in the relaxed home environment, or while on the bus/subway), where he can start to think about his spare time and maybe plan some fruitful shopping hours. With such service, the targeted advertisements become useful and even precious information for the user. These are not as annoying as massive ones because they

[2] However, to interrelate socio-cultural values and habits with functional needs is a sociological problem that is not described in this paper [6].

result from an user request and thus, they answer a real need. In particular, after giving some hints to the system about his preferences and hobbies, the user gets, without extra efforts, useful and needed information, well matched to his expectation. Then, he utilizes those inputs to get more detailed information regarding the route and the overall cost of the activity. Furthermore, the user would like to check whether in other shops, may be less distant, a similar offer is available. The previous considerations can be applied in case of restaurants and even for gasoline stations. Also exhibitions, cultural events, and concerts could be advertised according to the user's preferences.

3.3 Mobile Tourist Guide

A tourist walking in Paris can use his terminal to get instructions about the way to reach some sightseeing place, but also to interact with the environment in his surroundings warning him in case of some interesting detour route or giving information about something that is on the way to the final destination that he may most likely miss. Moreover, in a museum instead of buying the brochure or renting some electronic guides, all he needs is to download into his terminal a package in his language for a certain price and enjoy his tour listening to the audio guidance. For each work of art in the exhibition, he can automatically listen to the comments and explanations, without any effort of browsing through the guide. Also, by buying the ticket via his terminal or by signing up online on the waiting list which sends him back the approximate waiting time, the user avoids the problem of long queues of the famous museums. While he is on the virtual queue for the museum, he can go and enjoy another activity. The user terminal can also provide information about the culinary specialities of the city/region, where the nearest restaurant for getting a typical meal is situated.

3.4 Personalization Transfer

In a music festival or during a concert, the user wants to take pictures and record special moments with his friends and/or the entire event. He has a hand-held device – the most convenient to carry in a concert – that can support such demand. On the way back the pleasure of watching the pictures or videos is not limited on such device, since he can transfer the content to a publicly available larger screen – on the bus, at the train station, at the airport, etc. – and enjoy fully with his friends and the other people that were at the concert.

4 The "User-Centric" System

In this section we list and describe all the key features derived from the previous user scenarios. To do so, we show a framework illustrated in Figure 1 and referred to as the "user-centric" system, which we propose as the basis for the design of 4G systems [7].

Inspired by the Helios-centric Copernican theory, the user is located in the center of the system and the different key features defining 4G rotate around

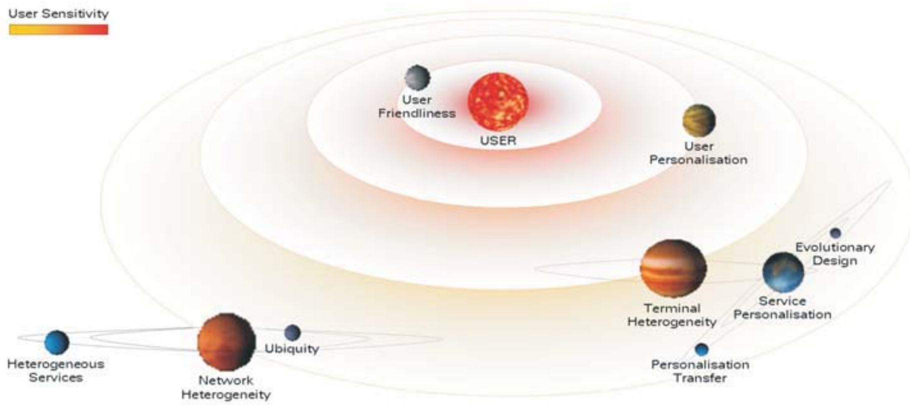


Fig. 1. The "User-Centric" System

him on orbits with a distance dependent on a user-sensitivity scale. Therefore, the further the planet is from the center of the system the less the user is sensitive to it. The decrease of the user-sensitivity leads to a translation towards the techno-centric system in which the network heterogeneity has a much stronger impact than the user friendliness. Furthermore, this kind of representation shows also the interdependency between key features and their relative technological developments: as shown in Figure 1, some of the planets have their own satellites.

The "user-centric" system demonstrates that it is mandatory in the design of 4G to focus on the upper layers (max user-sensitivity) before improving or developing the lower ones. Without user friendliness, for example, the user cannot exploit his device and access to other features, such as user personalization.

4.1 Key Features of 4G

User Friendliness and User Personalization. In order to encourage the people to move towards a new technology, which is a process that usually takes a long time and a great effort from the operators' side, the combination of user friendliness and user personalization appears to be as a the winning concept. User friendliness exemplifies and minimizes the interaction between applications and user thanks to a well designed transparency that allows the man and the machine to naturally interact (e.g., the integration of new speech interfaces is a great step on to achieve this goal). For instance, in Scenario A, the user can get information in text, audio, or video format so that the travelling information can be displayed in the most user-friendly way. User personalization refers to the way the user can configure the operational mode of his device and pre-select the content of the services chosen according to his preferences. Since every new technology is designed having in mind as the principal aim to penetrate the mass market and to strongly impact the people's lifestyle, the new concepts introduced by 4G are based on the assumption that the user wants to have the feeling that he is unique and thus he has exclusive needs. Therefore, in order to embrace a larger spectrum of customers, a high level of personalization must be provided, so that either the user terminal filters the huge amount of information delivered according to the user's flavors, or the operator sends only the information relevant to the user. This is illustrated in Scenario B where the user can receive targeted pop-up advertisements.

The combination between user personalization and user friendliness gives certainly to the user the idea of an easy management of the overall features of his device and the maximum exploitation of all the possible applications, conferring the right value to the user's expense.

Network and Terminal Heterogeneity. In order for 4G to be a step ahead of 3G, it must not only provide higher data rates but also some clear and evident advantage in people's everyday life. Therefore, the success of 4G consists in the combination of network and terminal heterogeneity. Network heterogeneity guarantees ubiquitous connection and provision of common services (e.g., voice telephony, etc.) to the user, ensuring at least the same level of *Quality of Service* (QoS) when passing from one network's support to another one. Moreover, due to the simultaneous availability of different networks, heterogeneous services are also provided to the user. For instance, in Scenario C the user can listen to a guided tour and he can purchase the entrance ticket for the museum as well. In contrast with 3G, 4G benefits from the terminal heterogeneity which is the support of different types of terminals in terms of display size, energy consumption, portability/weight, complexity, etc.

Since 4G will encompass various types of terminals that may have to provide common services independently of their capabilities, the tailoring of the content to the end-user device will be necessary to optimize the service presentation. Furthermore, as a result of the network heterogeneity, the upcoming new services will be accurately selected whether to be provisioned or not according to the capabilities of the terminal in use, in order to offer the best enjoyment to the user and to prevent a sensational flop of some service. This concept is referred to as service personalization (user personalisation works on top of it) and is clearly highlighted in Scenario D. It implicitly constrains the number of access technologies supportable by the user terminal. However, this limitation may be solved in the following two ways:

– By the development of devices with evolutionary design. A naive example can clarify this concept: in case the user has a watch-phone on which he would like to see a football match, just pressing a button on the watch's side a self-extracting monitor with a bigger screen can come out. Therefore, having the most adaptable device in terms of design can provide the user with the most complete application package, maximizing the number of services supported.
– By mean of a personalization transfer. An example extracted from Scenario D can clarify this concept: in case the user has a watch-phone on which he would like to see a video, he does not need to possess larger screen terminals as all the publicly available terminals can be borrowed by him for the displaying

time. Therefore, the advantage for the customer is to buy a terminal on which he has the potential to get the right presentation for each service, freeing it from its intrinsic restrictions. Furthermore, in a private environment the user can optimize the service presentation as he wishes exploiting the multiple terminals he has at disposal.

The several levels of dependency highlighted by the satellite hierarchical vision in the framework of the "user-centric" system definitely stress the fact that it is not feasible to design 4G starting from the access technology in order to satisfy the user's requirements.

5 Conclusions

In this paper, we have proposed a new top-down methodology composed by four different steps, ranging from the sociological perspective to the technical one. Starting from new 4G user scenarios we have then extrapolated a new framework – the "user-centric" system – that presents the key features of 4G: user friendliness and user personalization, network heterogeneity and terminal heterogeneity. Furthermore, its intrinsic satellite hierarchical structure shows the complex inter-dependencies among the various key features and outlines the real technical step up taken by 4G. The methodology proposed definitely demonstrates that it is mandatory in the design of 4G to focus on the user requirements before improving or developing the new technology.

Acknowledgements

This work has been supported by Samsung Electronics, Co., LTD, Korea.

References

1. E. Bohlin, S. Lindmark, J. Bjrkdahl, A. Weber, B. Wingert, P. Ballon, "The Future of Mobile Communications in the EU: Assessing the Potential of 4G", ESTO Publications, February, 2004.
2. K. Murota, NTT DoCoMo, "Mobile Communications Trends in Japan and DoCoMo's Activities Towards 21^{st} Century", in ACTS Mobile Summit99, Sorrento, Italy, June 1999.
3. F. Fitzek, "CTIF Definition of 4G in the Jade Project", February, 2004.
4. J. M. Pereira, "Fourth Generation: Now, it is Personal", in Proceedings of the 11^{th} IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), London, UK, September, 2000.
5. B. Shneiderman, "Leonardo's Laptop – Human Needs and the New Computing Technologies", The MIT Press, October 2002.
6. A. Gimmler, "D1.4: The Case of 4G – Social Aspects of the Next Generation of Communications Technologies", JADE Deliverable, July, 2004.
7. S. Frattasi, H. Fathi, F. Fitzek, Kiho Chung, R. Prasad, "4G: The User-Centric System", Mobile eConference 2004.

Integrating WMAN with WWAN for Seamless Services*

Jinsung Cho and Dae-Young Kim

Department of Computer Engineering,
Kyung Hee University, Yongin 449-701, Korea
chojs@khu.ac.kr

Abstract. Nowadays, wireless packet data services are provided over Wireless WAN (WWAN), i.e., cdma2000 1x/1xEV-DO mobile networks and Wireless MAN (WMAN) is being standardized for users to demand higher data rate services. WMAN can provide high data rate services, but its service coverage is relatively small. If WMAN may be integrated with WWAN, users are able to choose the optimal service according to service areas and get seamless services while they are moving around. At the same time, it is cost-effective for operators to construct and maintain the integrated network. In this paper, we propose an interworking scheme for the purpose of effectively integrating WMAN and WWAN. The proposed scheme adopts a tightly-coupled architecture for unified authentication/accounting and seamless services. In addition, we develop a performance model for handoffs between WMAN and WWAN. Through extensive simulations, it has been validated that the proposed scheme reduces packet losses dramatically compared with the loosely-coupled scheme.

1 Introduction

Recent advances in wireless communication technologies have provided driving forces behind the emergence of various wireless services. First of all, the mobile communication systems (WWAN) have been developed and evolved into the third generation. As a result of CDMA's enhanced capabilities and simplified migration path, the cdma2000 3GPP2 mobile communication system, one of the IMT-2000 standards, has been nation-widely deployed in Korea since the early of 2000, which is the world's first successful commercial deployment. Moreover, the number of Internet users has increased rapidly so that voice-centric services have changed into data-centric services. The cdma2000 mobile communication system has been evolved into 1xEV-DO and 1xEV-DV for high speed data services. The cdma2000 1xEV-DO services are also available in several countries including Korea.

As a result of the consecutive successful development of wireless networks, a new WMAN service, so-called WiBro (Wireless Broadband), has been defined

* This work was supported by SAMSUNG Electronics Co., LTD. in 2004.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 374–381, 2005.

in Korea for higher bandwidth with broader coverage. It has been developed to enable users to access the Internet anywhere anytime with high speed and good quality using portable equipments such as laptops, PDAs, and smart phones. WiBro network is based on IEEE 802.16 broadband wireless access [1]. It adopts OFDMA/TDD for multiple-access and duplex schemes, and aims to provide mobility rates up to 60km/h and data service rates up to 50Mbps. It has several additional functions to the IEEE 802.16 specification such as handoff, sleeping mode, periodic ranging, and bandwidth stealing [2].

The differences between WiBro and cdma2000 mobile communication systems can be considered in terms of cost, data rate, and coverage. That is, WiBro can provide high speed data communication with low costs but its service area is relatively small. If it effectively cooperates with existing cdma2000 mobile networks which are serviced in the whole country, customers are able to use enhanced and seamless services without interruption depending on service areas. Moreover, it will allow service providers to offer high speed data services with low costs through reducing expense for network construction and management. Considering handoffs between WiBro and cdma2000 services happen frequently, the technologies for seamless and continuous services should be carefully considered. In this paper, we propose an integration scheme between WiBro and cdma2000 mobile networks to provide seamless services. In addition, we develop a performance model for handoff between WiBro and cdma2000 mobile networks, and show the excellence of the proposed scheme through extensive simulations.

The remainder of the paper is organized as follows: Section 2 investigates various existing interworking schemes between 3G networks and WLANs as related work. In section 3, we propose our integration scheme in Section 3 and evaluate its performance through extensive simulations in Section 4. Section 5 presents some concluding remarks and future work.

2 Related Work

As there is no research effort on the integration of WiBro and 3G mobile networks to the best of our knowledge, we introduce the integration schemes between 3G networks and WLAN as related work. The early version of several research efforts on the integration of 3G mobile network and WLAN [3, 4, 5, 6, 7, 8] can be summarized as: (1) loosely-coupled and (2) tightly-coupled integration. In the loosely-coupled integration model, 3G network and WLAN exist independently and they also provide independent services. This integration model adds a gateway to support authentication and accounting for roaming services, and uses the mobile IP to provide mobility between WLAN and 3G network. Most existing research efforts on the integration of 3G networks and WLANs have focused on the loosely-coupled integration model [3, 4, 5, 6, 7] than the tightly-coupled integration approach, because of the service features of WLANs - that the mobility range of mobile nodes is very small. The advantage of the loosely-coupled integration is that it can be simply adapted to the existing communication systems and it thus can minimize the development efforts on making new standards.

On the other hand, in the tightly-coupled integration model, an AP in WLAN is connected to the SGSN or the PDSN in 3G networks, and thus it is possible to support integrated authentication, accounting, and network management. The tightly-coupled integration model requires further standardization work and, thus it may take far longer time to achieve the final step supporting seamless services and service continuity. Motorola laboratory proposed a tightly-coupled interworking method to integrate GPRS with WLAN [8]. Since the GPRS layer 1 and 2 in the proposed tightly-coupled integration model are simply substituted by the WLAN PHY and MAC, whereas the layer 3 of GPRS is used, the integration system requires additional non-trivial overhead on a mobile node and needs a gateway to support the functions of the GPRS layer 3. To the best of our knowledge, this is the unique research to present the tightly-coupled scheme for WLAN and GPRS.

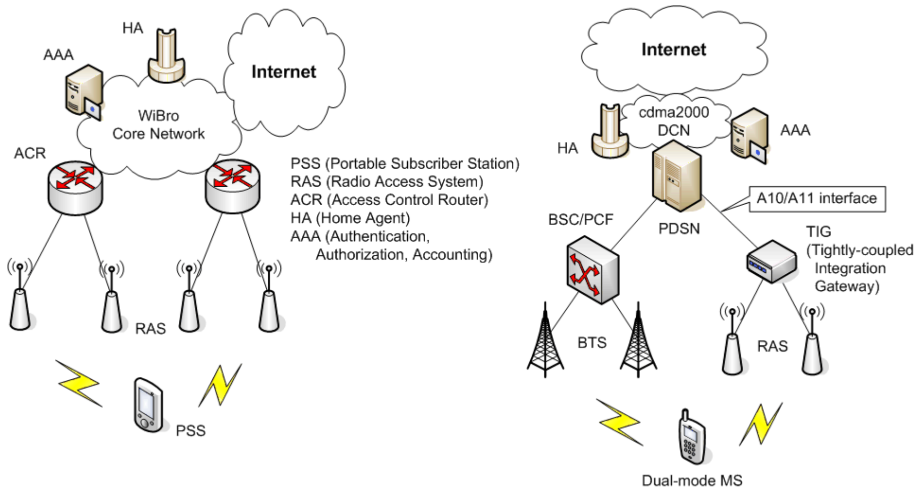
3 The Proposed Scheme

3.1 The Network Architecture

First of all, we introduce the current architecture of the WiBro network in Figure 1(a). There are four main components in the architecture: PSS, RAS, ACR, and WiBro Core Network. PSS communicates with RAS using WiBro wireless access technology. The PSS also provides the functions of MAC processing, mobile IP, authentication, packet retransmission, and handoff. The RAS provides wireless interfaces for the PSS and takes care of wireless resource management, QoS support, and handoff control. The ACR plays a key-role in IP-based data services including IP packet routing, security, QoS and handoff control, and foreign agent (FA) in the mobile IP. The ACR also interacts with AAA server for user authentication and billing. To provide mobility for PSS, the ACR supports handoff between the RASs while the mobile IP provides handoff between the ACRs.

In order to integrate the current WiBro architecture in Figure 1(a) with cdma2000 mobile network, the loosely-coupled integration model which was introduced in Section 2 may be considered. However, whereas WLAN provides services for fixed stations, WiBro can provide mobile services. As the service area of WiBro is smaller than that of cdma2000, vertical handoffs between WiBro and cdma2000 networks may happen frequently. Hence, the integration scheme for seamless services on handoff must be carefully considered.

Figure 1(b) depicts a tightly-coupled integration architecture. As shown in Figure 1(b), RAS in WiBro can be connected to PDSN in cdma2000 network through TIG which converts packets from RAS into ones conformed to A10/A11 interfaces [9], and vice versa. While adopting the tightly-coupled integration architecture, we should also take into account that cdma2000 networks have already been deployed and widely used. That is, the modification of existing nodes in cdma2000 networks should be minimized. To do that, we develop an efficient integration scheme in the next subsection.



(a) Current WiBro architecture (b) Proposed integration architecture

Fig. 1. Network architecture

3.2 The Integration Scheme

In cdma2000 packet data services, a mobile station first creates a PPP connection with PDSN. During that procedure, user authentication and IP address allocation are performed. On the other hand, in WiBro, EAP for user authentication and DHCP for IP address allocation are being considered. For the tightly-coupled architecture in Figure 1(b), however, it is efficient that WiBro adopts the mechanism of cdma2000 packet services. That is, a dual-mode MS handles PPP connections for WiBro as well. There are several advantages for doing that. First, the implementation of dual-mode MS is less complicated by adopting the same mechanism. Second, there is no modification of PDSN - this is the most important because cdma2000 networks have been deployed already. Third, the functionality of TIG is only to convert packet formats (i.e., not to process signaling messages and data traffic), and thus, TIG can be implemented with ease.

Figure 2 shows the proposed protocol architecture for our scheme. The dual-mode MS in Figure 2 shares IP and PPP layers between cdma2000 and WiBro

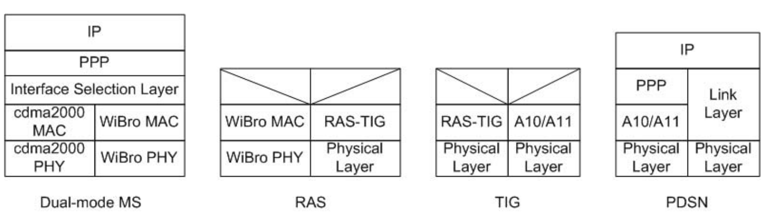


Fig. 2. Protocol architecture

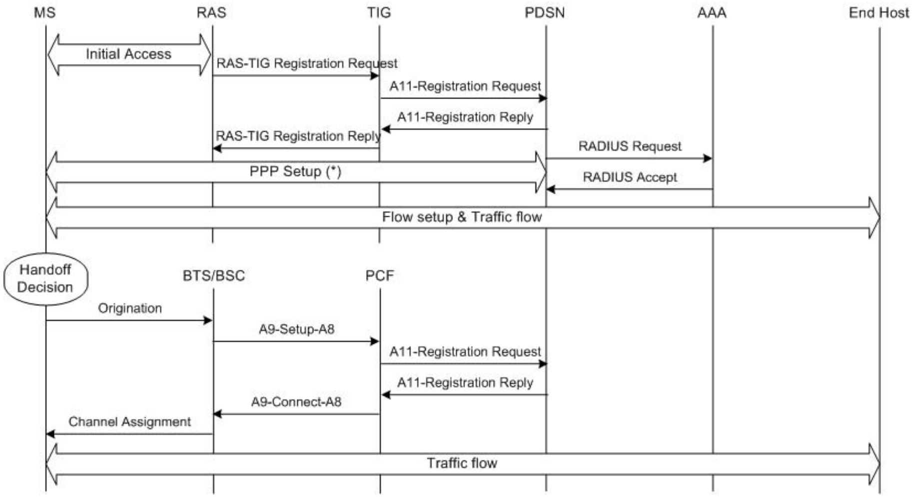


Fig. 3. Signaling and data flows

services. The ISL (Interface Selection Layer) selects the optimal interface according to link quality, signal strength, and so on. As TIG communicates PDSN with the standard A10/A11 interfaces [9], existing PDSN can be employed without any modification.

Figure 3 illustrates an example of the signal and data flows in the proposed scheme. The WiBro data service procedure is shown in the first part of Figure 3 as mentioned earlier. Once MS gets out of WiBro service area, MS performs a vertical handoff to cdma2000 network. In the meanwhile, PDSN buffer packets to the MS and send them after handoff completion. The buffering in PDSN reduces packet losses dramatically during handoff, which will be validated in the next section.

4 Performance Evaluation

The proposed scheme in Section 3 was intended for seamless services on vertical handoffs between WiBro and cdma2000 mobile networks. In this section, we evaluate the performance of our scheme through extensive simulations compared with the loosely-coupled integration which is based on mobile IP.

4.1 Simulation Model

The handoff delay of the proposed scheme ($D_{proposed}$) consists of $t_{release}$ (time to release the link in old network), t_{access} (time to create a wireless link in new network), and $t_{signaling}$ (time to deliver/process signaling messages in new network).

$$D_{proposed} = t_{release} + t_{access} + t_{signaling} \tag{1}$$

Table 1. Simulation parameters

Parameter	Value	Parameter	Value
t_{MS-BTS}	10ms	t_{MS-RAS}	8ms
$t_{BTS-BSC}$	5ms	$t_{RAS-TIG}$	5ms
$t_{BSC-PCF}$	1ms	$t_{TIG-PDSN}$	1ms
$t_{PCF-PDSN}$	1ms	$t_{RAS-ACR}$	5ms
$t_{PDSN-HA}$	1ms	t_{ACR-HA}	1ms

(a) Network model

Parameter	Value
T_{CDMA}	60s (exponential distribution)
T_{WiBro}	60s (exponential distribution)
p, q	0.2, 0.5, 0.8 (low, medium, high mobility)
r	100kbps $\sim$ 1Mbps (uniform distribution)
T_{think}	5s (exponential distribution)
M	60KB (exponential distribution)
s	100KB(streaming), 500KB(interactive)

(b) Mobile station and traffic model

In Eq. (1), $t_{release}$ and $t_{signaling}$ can be calculated as $t_{MS-BTS} + t_{BTS-BSC} + t_{BSC-PCF} + t_{PCF-PDSN}$ and $t_{RAS-TIG} + t_{TIG-PDSN}$, respectively, when MS moves from cdma2000 to WiBro network. They can be also calculated similarly on the reverse direction (i.e., from WiBro to cdma2000 network). After $t_{release}$, PDSN may be informed that a MS has moved to the other network, and hence, PDSN can buffer packets to the MS and can send them after handoff completion. So, the packet loss time is only $t_{release}$ in our scheme (i.e., $L_{proposed} = t_{release}$).

On the other hand, to calculate the handoff delay in the loosely-coupled integration scheme ($D_{loosely}$), the time for mobile IP registration ($t_{mobileIP}$) should be added to Eq. (1).

$$D_{loosely} = t_{release} + t_{access} + t_{signaling} + t_{mobileIP} \tag{2}$$

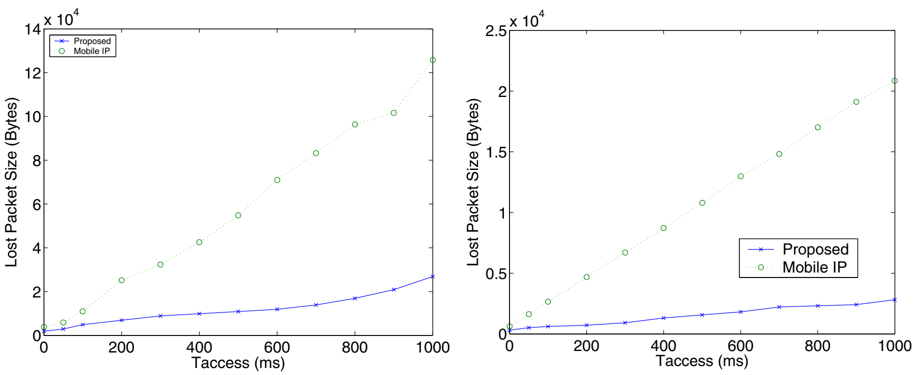
In Eq. (2), $t_{mobileIP}$ is calculated as $t_{MS-RAS} + t_{RAS-ACR} + t_{ACR-HA}$ when MS moves from cdma2000 to WiBro network. In loosely-coupled interworking, the packet loss time ($L_{loosely}$) is given $D_{loosely}$ because any node in new network is not aware of the handoff event before the mobile IP registration. In order to reduce the large delay on handoff based on mobile IP, several works are in progress including fast handoff in mobile IPv6. However, there is no scheme which is being standardized in WiBro and cdma2000 mobile network.

For the purpose of modeling the behavior of users, we assume the following scenario: a MS gets the cdma2000 service for T_{CDMA} seconds and moves to WiBro network with the probability of p. After T_{WiBro} seconds, the mobile station moves back to cdma2000 network with the probability of q. The large values of p and q indicate the high mobility.

As for the traffic model, we consider two types of services: real-time streaming and interactive. The interval to transmit packets in real-time streaming services is given $\tau_{streaming} = s/r$, where s is the packet size and r is the data rate of a stream. In interactive services, users request a web page of M bytes every T_{think} seconds. The packet interval $\tau_{interactive}$ is set considering the round-trip time between service end-points. Table 1 summarizes the simulation parameters used in this paper. Since our concern is centered on only the comparison of interworking architecture, we assume there is no packet loss in wireless links.

4.2 Simulation Result

Figure 4 shows the packet loss per handoff for 30 mobile stations. The x-axis of Figure 4 means t_{access} described in Eq. (1) and (2). The values of t_{access} are expected to be diverse according to the implementation details. Luo *et al.* measured the wireless link access time ($400 \sim 600ms$) from their WLAN and 3G inteworking prototype [5]. As shown in Figure 4, the packet loss in the proposed scheme is very small across all the values of t_{access}. This is due to buffering in PDSN as mentioned in Section 3.



(a) Real-time streaming service (b) Interactive service

Fig. 4. Simulation result: packet loss per handoff

More specifically, on $t_{access} = 500ms$ which is the average value of wireless link access time measured in [5], handoffs in the loosely-coupled integration scheme may cause to large loss of packets, resulting in the degradation of service quality. In addition, whereas our scheme does not require any re-authentication on handoff, user authentication in each network should be performed on handoff in loosely-coupled interworking. This will cause that far more packets may be lost in the loosely-coupled architecture. Therefore, the loosely-coupled interworking requires additional schemes to reduce packet losses. It may be achieved by terminal support like reducing t_{access} or by network support such as fast handoff.

5 Conclusion

In this paper we proposed an integration scheme between WiBro and cdma2000 networks to provide seamless services. We have designed a practical model considering the fact that cdma2000 mobile communication networks have already been implemented and widely used. We defined not only an efficient interworking model but also practical implementation methods in node operations, protocols, and interfaces between nodes, and so forth. Thus, this paper can give a theoretical and practical guideline to design WiBro which cooperates with current cdma2000 mobile networks.

In addition, we have developed a performance model for handoffs between WiBro and cdma2000 networks. From the model, it has been validated that the proposed scheme reduces packet losses compared with the loosely-coupled scheme. This is because PDSN can buffer packets during handoff period in our scheme. However, the increased round-trip time on handoff period may cause to degradation in TCP performance. We are currently tackling the problem.

References

1. I. Koffman and V. Roman, "Broadband wireless access solutions based on OFDM access in IEEE 802.16," *IEEE Communications*, Vol. 40, pp. 96–103, 2002.
2. TTA, `http://www.tta.or.kr/Home2003/committee/ommitToR.jsp?commit_code=PG05`
3. 3GPP, "3GPP system to WLAN interworking: Functional and architectural definition," *3GPP TR 23.934*, 2002.
4. M. M. Buddhikot, *et. al.*, "Design and implementation of a WLAN/CDMA2000 interworking architecture," *IEEE Communications*, Vol. 41, pp. 90–100, 2003.
5. H. Luo, *et. al.*, "Integrating wireless LAN and cellular data for the enterprise," *IEEE Internet Computing*, Vol. 7, pp. 25–33, 2003.
6. E. Gustafsson and A. Jonsson, "Always best connected," *IEEE Wireless Communications*, Vol. 10, pp. 49–55, 2003.
7. K. Ahmavaara, *et. al.*, "Interworking architecture between 3GPP and WLAN systems," *IEEE Communications*, Vol. 41, pp. 74–81, 2003.
8. A. K. Salkintzis, *et. al.*, "WLAN-GPRS integration for next-generation mobile data networks," *IEEE Wireless Communications*, Vol. 9, pp. 112–124, 2002.
9. 3GPP2, "Interoperability specification (IOS) for cdma2000 access network interfaces - part 1 overview," *3GPP2 A.S0011-A*, 2002.

Towards Mobile Broadband

J. Charles Francis and Johannes Schneider

Swisscom Innovations, Ostermundigenstrasse 93, 3050 Bern, Switzerland
JohnCharles.Francis@Swisscom.com

Abstract. This paper discusses the Open Access Network paradigm, whereby surplus capacity from residential DSL and fibre connections is made available for a public mobile service based on WLAN-technology. In this vision, a subscriber on-the-move can seamlessly traverse urban and suburban environments, while meeting needed QoS and security constraints, with connectivity based on residential WLANs. The approach represents a potentially cost-effective route towards 4G objectives.

1 Introduction

Cellular systems such as GSM and UMTS deploy radio antenna masts at locations selected according to site availability and network planning criteria. Base-station sites must be acquired by the mobile operator, a time-consuming and expensive process that is subject to planning permission by local authorities and objections from concerned members of the public. Looking to future, it is highly questionable whether a mobile broadband network can be economically deployed in this manner. Radio spectrum is shared finite resource, and delivering more bits-per-square-metre is accompanied by a reduction in radio range (Table 1). Geometrical considerations show that as the cell size is reduced, the number of base-stations needed to cover a given area increases at the square (Fig. 1). Moreover, operational and capital expenditure, being proportional to the number of cells, will rise in like manner. This motivates consideration of an alternative paradigm, the Open Access Network, in which surplus residential fixed-line capacity is leveraged for a public mobile broadband service.

Table 1. Relationship between bit-rate and radio range

Wireless Technology	*Bit-rate*	*Range*
GSM Voice (urban)	14 kb/s	**4 km**
UMTS 384 kb/ (urban)	384 kb/s	**1 km**
802.11b	6 Mb/s	**100 m**
802.11g	**20 Mb/**s	**50 m**

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 382 – 387, 2005.

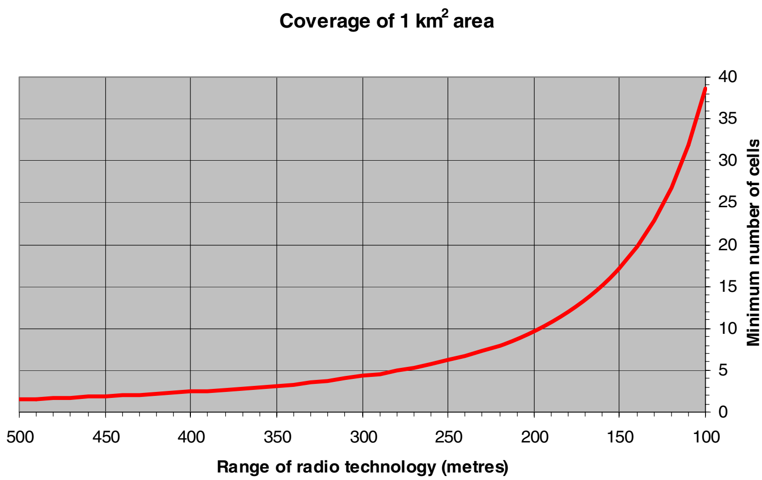


Fig. 1. As the radio range decreases, the minimum number of base-stations needed for contiguous coverage rises at the square

2 Open Access Networks

Following the unprecedented uptake of residential ADSL, considerable communication capacity has been made available in urban and suburban areas. Future capacity increases will result, moreover, from VDSL and FTTx deployments. However, since each residential high-speed link is typically devoted to one household, the connection is under-utilised. Typical residential usage patterns are intermittent: no activity for significant portions of day, with spikes at other times due to TV viewing, Web browsing, email, and file transfer. Better utilisation would be achieved if each fixed-line were harnessed as a public resource supporting many users, so allowing statistical factors to come into play. Another growth technology, Wireless-LAN, provides the means to share what has traditionally been a private resource.

Wireless Local Area Network (WLAN) technology has attracted considerable interest worldwide, with deployments in homes, offices, and public hotspots. Interworking with cellular systems has been addressed by the 3rd Generation Partnership Project (3GPP) [1, 2, 3, 4, 5], while specifications for integrating cellular and WLAN have been released by the Unlicensed Mobile Access (UMA) consortium [6, 7, 8]. Multimode phones supporting WLAN and cellular interfaces have arrived, allowing WLAN to be used for voice services. The convenience of using one phone for both cellular and fixed-line voice services, and the lower cost of fixed-network calls, will drive customer demand to be connected to the fixed-network wherever possible. This is also the case for Internet users.

The Open Access Network (OAN) concept [9, 10, 11, 12, 13, 14, 15, 16], releases surplus capacity from residential and corporate fixed-lines, including DSL and fibre, by means of residential WLAN. It foresees a broadband mobile network where the subscriber on the move is seamlessly connected to the fixed-network with needed quality of service and security requirements (Fig. 2). In the OAN approach, the radio signal that propagates outside the home is leveraged for a public broadband mobile service. Wireless technologies include IEEE 802.11 a / b / g variants, while newer technologies such as IEEE 802.11n and 802.16 may extend range and mitigate against interference. Coverage in the public environment is strongly influenced by antenna placement (e.g., use of an external antenna, antenna next to external wall, etc.) and by deployment of multiple antenna techniques including MIMO.

Mobility functionality is required to track the whereabouts of the mobile user for push-services such as incoming call set-up, and to ensure fast, seamless, handover. Mechanisms are also needed to govern the access of users to subscribed services and to guarantee the integrity and privacy of the household where the WLAN access point is located. Quality of Service (QoS) must be provided to guarantee the service for the residential user, while allowing surplus capacity to be utilised by the public. Conversational services such as VoIP need guaranteed bandwidth to avoid interruptions, while best-effort services may send and receive packets whenever surplus capacity becomes available due to the statistical fluctuations in residential traffic. Where the residential subscriber has signed-up for less capacity than the fixed-line can actually deliver, the surplus can be allocated to public users. Alternatively, subscribed capacity may be used intermittently (e.g. for Web browsing), and in this case the surplus can be offered to the public on a best effort basis.

There is a fundamental shift in paradigm from local-loop connections devoted to one household ("privately owned") and the local loop as a feed for a public wireless service. Residential customers may be reluctant to open their home network resources to public users without incentive. Business models involving revenue sharing may therefore be needed with associated network charging algorithms.

To illustrate the approach concretely, consider the Swiss town of Olten with some 20'000 inhabitants. Based on publicly available statistics, the town's non-wooded area is around 7 km^2, an approximate radius of 1.5 km, leading to minimum requirements for reasonably contiguous coverage of around 300 WLAN access points. So, for reasonable coverage some 5% of households need to offer public WLAN. The use of umbrella cells based on longer-range technologies such as 802.16 (WiMax) together with the 3G cellular network facilitates coverage with fewer WLAN Access Points. By equipping already cabled facilities such as phone booths and transmission cabinets with WLAN, coverage can be further improved.

2.1 Fixed-Network Coverage

Figure 3 depicts a high-level view of the fixed-network. The architecture includes copper local-loop components running DSL technology supported by an optical feeder. The fibre penetration depth towards the customer, leads to such notions as FTTCab (Fibre to the cabinet), FTTC (Fibre to the curb), FTTB (Fibre to the building), FTTO (Fibre to the office), FTTBus (Fibre to the business) and FTTH (Fibre to the home).

ADSL runs over copper twisted pairs and has been developed for asymmetric applications. The available bit-rate depends on the length of the copper line, and decreases with increasing distance due to attenuation and cross-talk. Estimates of reachablility versus bit-rate for ADSL customers are shown in Fig. 4. ADSL+ is variant that leaves the upstream channel more or less unchanged, but doubles the downstream bandwidth for a specified distance. Options for supporting symmetric services over copper include SDSL (ETSI-terminology) and SHDSL (ITU-terminology), which offer increased performance and compatibility compared to the older HDSL technology. VDSL provides higher bandwidth over copper twisted pairs and can be used for both for symmetrical and asymmetrical services. The higher bandwidth necessitates a higher frequency carrier, however, which limits the local-loop line length to around 300-1500 metres according to bit-rate. Laboratory measurements illustrating the trade-off between bit-rate and range are shown in Fig. 5.

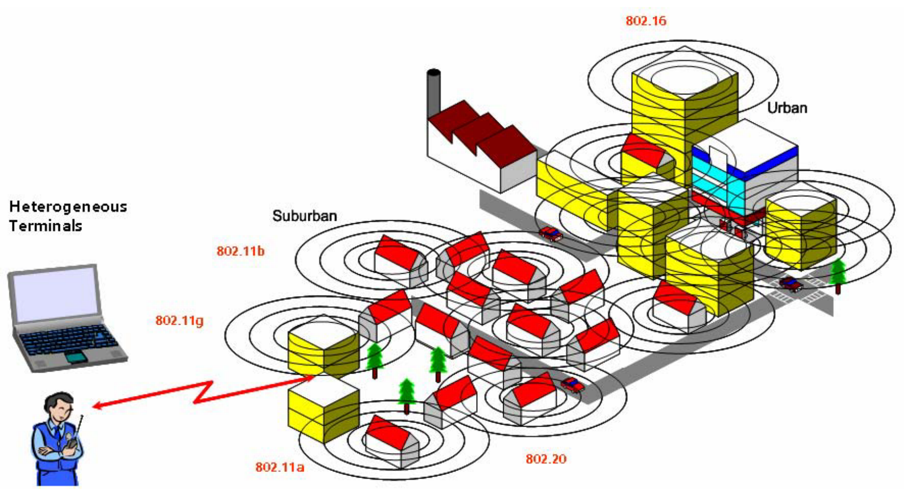


Fig. 2. An Open Access Network consists of many WLAN base-stations connected to the fixed-network and primarily located in private homes

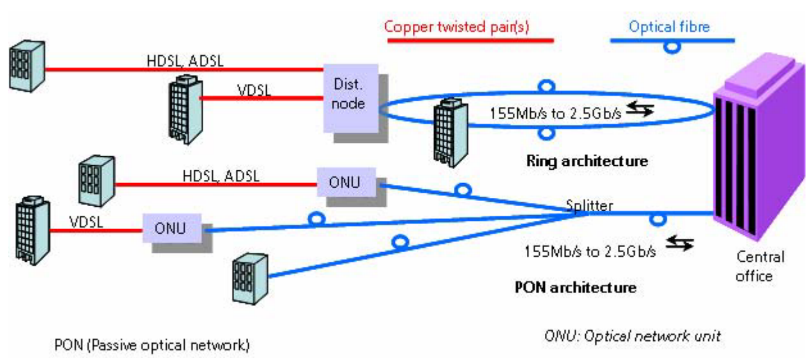


Fig. 3. Fixed-network topology for urban and suburban environments

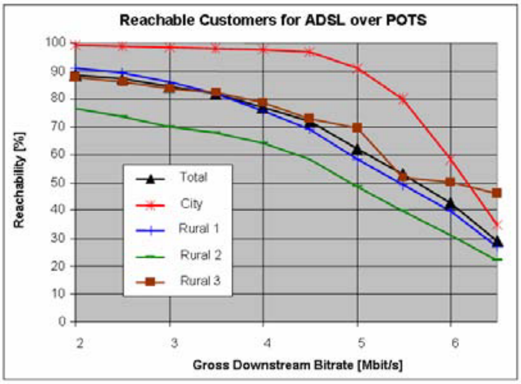


Fig. 4. Reachability versus bit-rate for ADSL residential customers

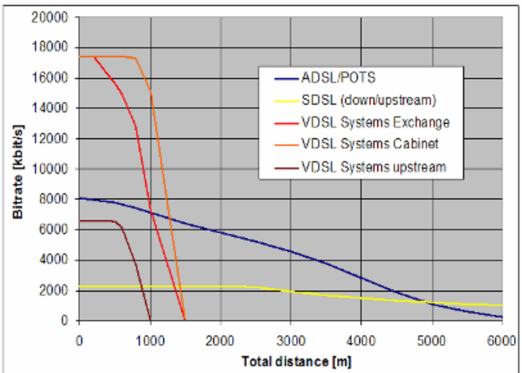


Fig. 5. Trade-off between bit-rate and range for ADSL, SDSL, and VDSL

3 Conclusions

In contrast to 4G approaches that attempt to “broaden” the narrowband mobile network, research on Open Access Networks seeks to demonstrate the feasibility of providing 4G wireless services by leveraging surplus capacity in the fixed-network via residential WLAN. To mobilise this capacity, a range of fundamental research issues must be addressed.

References

1. 3GPP TR 22.934 V6.2.0 (2003-09), 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Feasibility study on 3GPP system to Wireless Local Area Network (WLAN) interworking (Release 6)

2. 3GPP TS 23.234 V6.1.0 (2004-06), 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3GPP system to Wireless Local Area Network (WLAN) interworking; System description (Release 6)
3. 3GPP TS 24.234 V2.0.0 (2004-09), 3rd Generation Partnership Project; Technical Specification Group Core Network; 3GPP System to Wireless Local Area Network (WLAN) interworking; User Equipment (UE) to network protocols; Stage 3 (Release 6)
4. 3GPP TS 32.252 V.0.2.1 (2004-04), 3rd Generation Partnership Project; Technical Specification Group Service and System Aspects; Telecommunication management; Charging Management; Wireless Local Area Network (WLAN) charging; (Release 6)
5. 3GPP TS 33.234 V6.1.0 (2004-06), 3rd Generation Partnership Project; Technical Specification Group Service and System Aspects; 3G Security; Wireless Local Area Network (WLAN) interworking security (Release 6)
6. UMA User Perspective (Stage 1) R1.0.0 (2004-09-01) Technical Specification Unlicensed Mobile Access (UMA); User Perspective (Stage 1)
7. UMA Architecture (Stage 2) R1.0.0 (2004-09-01) Technical Specification Unlicensed Mobile Access (UMA); Architecture (Stage 2)
8. UMA Protocols (Stage 3) R1.0.0 (2004-09-01) Technical Specification, Unlicensed Mobile Access (UMA); Protocols (Stage 3)
9. Business modelling for systems B3G, P1203: The operator's vision on systems beyond 3G, Eurescom, Heidelberg, 2003
10. Francis J.C. and Fischer, C., .Mobile Networks Beyond 3G , Comtec Jour., Bern, 6/2003
11. Eskedal, T., Venturin R., Grgic, I., Andreassen, R., Francis, J.C., Fischer, C., Open Access Network Concept, a B3G Case Study, Proc. IST Mobile Communications Summit, Aveiro, 2003
12. Edvardsen E, Eskedal T. G., Arnes A., Open Access Networks, in “Converged Networking”, Ed. Chris McDonald, 2003
13. Francis, J. C., Open Broadband Access Networks, Comtec Jour., Bern, 5/2004
14. Francis, J. C., OBAN Scenarios”, Proc. EC SB3G Workshop, Brussels, 2004
15. Francis, J.C., A Mobile Paradigm for Fixed Networks, Proc. Interworking 2004, Ottatwa
16. Kuzminskiy A., Karimi H. A., Edvardsen E, Francis J. C., Interference Scenarios in Future Wireless Open Access Networks, Proc. WWRF11 WG5, Oslo, 2004

Emulation Based Performance Investigation of FTP File Downloads over UMTS Dedicated Channels

Oumer M. Teyeb[1], Malek Boussif[1], Troels B. Sørensen[1], Jeroen Wigard[2], and Preben E. Mogensen[2]

[1] Department of Communication Technology, Aalborg University, Fredrik Bajers Vej 7A, 9220 Aalborg East, Denmark
{oumer, mb, tbs}@kom.aau.dk

[2] Nokia Networks, Aalborg R&D, Niels Jernes Vej 10, 9220 Aalborg East, Denmark
{jeroen.wigard, preben.mogensen}@nokia.com

Abstract. The Radio Link Control (RLC) protocol of Universal Mobile Telecommunication System (UMTS) provides link layer reliability that could mitigate the effects of the hostile radio propagation channel on packet data transmission. In this paper, the impact of some of the RLC reliability mechanisms on the performance of File Transport Protocol (FTP) is investigated. The investigations are carried out using a real time emulation platform, which makes the results from this study more realistic than simulation or simplified analytical studies as the overall End-2-End performance is analyzed involving real world protocol implementations.

1 Introduction

Provision of data services is the main driving force behind the current standardization and deployment of 3G (Third Generation) networks. The fact that most of the packet services on the wired Internet today use Transmission Control Protocol (TCP) (TCP averages about 95% of the bytes, 90% of the packets, and 80% of the flows on the Internet [1]) is a clear indication that TCP is also going to be the transport protocol of choice for services running over 3G and beyond networks.

TCP is intended for use as a highly reliable host-to-host protocol in a packet switched computer communication networks[2]. The main features of TCP are its well-designed flow and congestion control mechanisms, which operate on top of its provision of reliability [3]. However, these TCP mechanisms were designed under the assumption that packet losses are only due to network congestion. Though this assumption holds for wired networks, it does not in wireless networks such as UMTS. These networks differ inherently from their wired counterparts, as they have higher error rates, higher latency, and lower yet highly variable bandwidth. As such, TCP performance over wireless networks is quite different from that in

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 388–396, 2005.

wired networks. In UMTS, link layer retransmission mechanisms already exist that can mitigate the influence of higher error rates on TCP performance.

In this paper, we discuss the effects of the settings of some of the UMTS RLC protocol reliability mechanisms on file downloads using FTP through emulation-based studies. Section 2 describes the RLC protocol. A description of the tool used to perform the investigations along with the different mechanisms that are under focus and the performance evaluation metrics is given in section 3. Section 4 discusses the performance evaluation results, and finally section 5 gives conclusions and some pointers to future work.

2 The RLC Protocol

In UMTS, reliable data transmission over the radio channel is provided through the RLC protocol[4]. RLC achieves this link layer reliability through Selective Repeat (SR) Automatic Repeat reQuest (ARQ) mechanism[1]. When the RLC receives a Service Data Unit (SDU) from upper layers, it segments it into RLC Protocol Data Units (PDUs), schedules the PDUs for transmission and then stores them into its (re)transmission buffer. Each PDU is given a unique Sequence Number (SN). Each Transmission Time Interval (TTI), the sender transmits a given number of PDUs depending on the instantaneous allocated bit rate of the air interface.

The receiver keeps the received PDUs in its reception buffer. When all the PDUs that comprise an SDU are received, the receiver assembles the SDU and sends it to upper layers. Depending on the setting of the parameter *in-sequence delivery*, SDUs can be sent to upper layers in- or out-of sequence. PDUs that are received properly are positively acknowledged (ACKed) by the receiver, and those that are not received properly are negatively acknowledged (NACKed). The sender removes the ACKed PDUs from its retransmission buffer, and retransmits the NACKed ones. The receiver sends the ACKs and NACKs using *status* PDUs that contain cumulative ACKs and bitmap fields. A value of n in the cumulative ACK field signifies the correct reception of all PDUs with $SN \leq n$. NACKs and non-cumulative ACKs are sent using bitmaps. For example, if the receiver has received PDUs $\#1, 2, 4, 6$ but not PDU $\#3$ and 5, it will put 2 in the cumulative ACK field and the values $[0, 1, 0, 1]$ in the bitmap.

Status reporting is triggered by the sender, the receiver or both. The sender triggers status reporting by setting the polling bit of some of the PDUs that it sends. The different mechanisms that control poll triggering are:

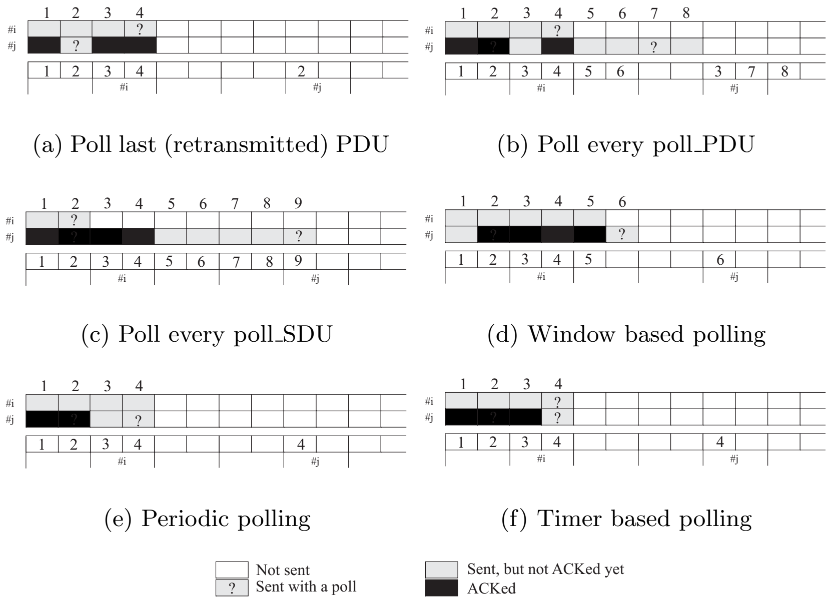
- **Poll last PDU and Poll last Retransmitted PDU:** If *Poll last (Retransmitted) PDU* is set, the last PDU in the transmission (retransmission) buffer that is sent will be polled. In fig. 1(a), PDU #4 will be polled at TTI

[1] The RLC protocol can operate in three modes, namely Transparent, Unacknowledged and Acknowledged. Only the Acknowledged mode supports retransmission mechanisms, and throughout this paper by RLC it is meant RLC Acknowledged mode.

$\#i$ if Poll last PDU is set and the retransmitted PDU #2 will be polled at TTI $\#j$ if Poll last Retransmitted PDU is set.
- **Poll every poll_PDU:** When this is configured, the polling bit is set on every poll_PDUth (re)transmitted PDU. For example, in fig. 1(b), the poll_PDU value was set to 4. Thus, at TTI $\#i$, when the 4^{th} PDU is sent, and when PDU #7 is sent at TTI $\#j$ the poll bits are set.
- **Poll every poll_SDU:** When this is configured, the polling bit is set on the last PDU of every poll_SDUth SDU. For example, in fig. 1(c), the poll_SDU is set to 1. At TTI $\#i$, when #2 is sent, and at TTI$\#j$, when #9 is sent, the polling bit is set, as they are the last PDUs of the corresponding SDUs.
- **Window based polling:** With *window based polling*, a poll is sent when the % of the occupied transmission window (i.e. the spacing between the first and last non-ACKed PDUs) reaches a certain threshold. In fig. 1(d), this threshold is set to 60%, and the window size is set to 10 PDUs. At TTI $\#j$, % reaches 60, triggering the sending of a poll along with PDU #6.
- **Periodic polling:** If this is configured, a poll is sent regularly at a specified period. In fig. 1(e), the period is set to ten TTIs. When the ten TTIs have elapsed (when we reach TTI $\#j$), a poll is triggered. As there are no PDUs scheduled to be sent, one of the PDUs that have not been acknowledged yet will be resent with the poll.
- **Timer based polling:**When this is configured, a timer is started whenever a poll is sent. When the timer expires, if the sender has not received a NACK or a cumulative ACK for the PDU with the highest SN that was waiting for an ACK when the poll was sent, polling will be triggered. In fig. 1(f), a poll was sent at TTI $\#i$ and when this poll was sent, the last PDU that was expecting an ACK was #4. When the timer expired at TTI $\#j$, neither a NACK nor a cumulative ACK for #4 has been received so a poll will be sent along with the retransmission of #4.

The sender sends a poll in order to receive the status of the PDUs that it has transmitted. The receiver responds to this request by sending a status PDU. In fig. 2(a), for example, the receiver gets a status request along with PDU #4 at TTI $\#i$. During the next TTI, the receiver sends a status report. In this case, as everything up to PDU #4 is received, the status contains a cumulative ACK for #4. There are two other mechanisms in RLC that enable the receiver to send a status report without being explicitly polled by the sender:

- **Periodic status reporting:** With this configured, a status is sent regularly at a specified period. In fig. 2(b), the status periodic is set to two TTIs. At TTI$\#i$ a poll is received so the receiver sends a status report (containing cumulative ACK for #2). At TTI $\#i + 2$, the periodic status fires for the first time. At TTI $\#i + 4$, the periodic status timer fires again and a new status report, containing cumulative ACK #4, is sent.
- **Missing PDU indication:** When this option is set, a status is sent when the receiver gets PDUs out of order, which is a likely indication that some PDUs may have been lost. In fig. 2(c), PDUs #3 and #4 are lost in the air interface. The receiver gets PDU #5 while it was expecting #3 at TTI



(a) Poll last (retransmitted) PDU

(b) Poll every poll_PDU

(c) Poll every poll_SDU

(d) Window based polling

(e) Periodic polling

(f) Timer based polling

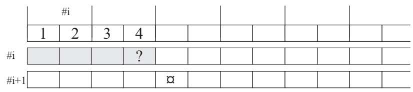
Fig. 1. RLC polling mechanisms. In the sub-figures, the 1^{st} and 2^{nd} rows represent the state of the transmission buffer at different TTIs, and the 3^{rd} row represents the SNs of the PDUs sent at different TTIs

$\#i + 2$. The receiver responds, if missing PDU indication is set, by sending a status PDU, containing cumulative ACK for #2, and a $[0, 0, 1, 1]$ bitmap.

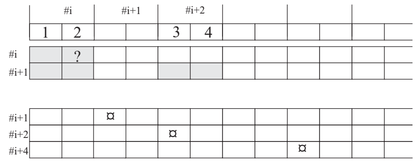
The minimum temporal spacing between consecutive polls and status reports can be controlled by using *Poll Prohibit* and *Status Prohibit* timers, respectively. When these are set, consecutive polls (status reports) will not be sent unless they are separated by a time greater than the configured timers. The number of times that a given PDU could be retransmitted can be specified using the *maxDAT* parameter. If a PDU has been retransmitted *maxDAT-1* number of times and still has not been received properly, the SDU that this PDU is part of will be discarded.

3 Emulation Tool and Investigated Scenarios

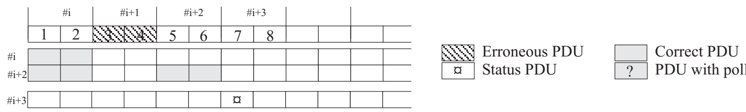
The investigations are carried out using RESPECT, a real-time emulation platform for UMTS[5]. RESPECT provides a link-level, real-time emulation of a UMTS network using off-the-shelf Linux operating system. Every Internet Protocol (IP) packet that is arriving to or departing from the machine where the emulator is running is diverted into the emulator, which passes it into an emulated UMTS protocol stack, making it experience the effects of a UMTS network.



(a) Status sent due to poll reception



(b) Periodic status reporting



(c) Missing PDU indication

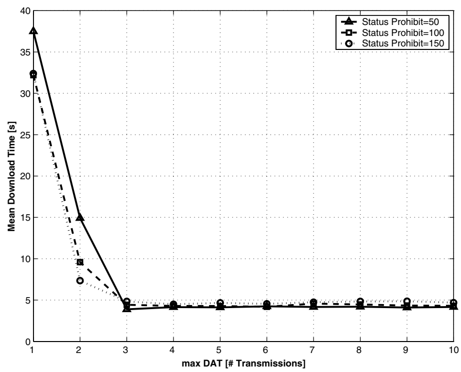
Fig. 2. RLC status reporting mechanisms. In the sub-figures, the 1^{st} row represents the PDUs that are being received during the indicated TTI, the 2^{nd} and 3^{rd} (set) of rows represent the states of the reception and transmission buffers, respectively, at the receiver side at different TTI instances

FTP sessions are started and during these sessions, it is assumed that a dedicated channel (DCH) is already setup. The DCH in the uplink (UL) is considered to be error free with a fixed bandwidth of $32kbps$, while the downlink (DL) has a fixed bandwidth of $384kbps$ with a constant, uncorrelated, frame erasure rate (FER) of 10%. In-sequence delivery is set both in the UL and DL.

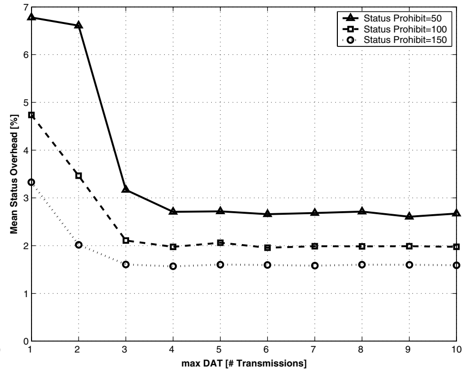
Based on the values given in [6] and [7], one way UMTS processing delay, without considering the transmission time in the air interface, is taken to be $57.5ms$. No limitations are put on RLC buffer and window sizes. *Poll last PDU*, *Poll last Retransmitted PDU* and *Missing PDU indication* are always set, as they help in avoiding many deadlock situations that may arise due to infrequent polling. The emulations were carried out on a machine with the linux 2.4 kernel TCP implementation. The download time, throughput, SDU delay (time taken from the arrival of an SDU till its complete reception and assembly), and status overhead (percentage of status PDUs that are sent as compared with the data PDUs) are used to evaluate the results.

4 Results and Discussion

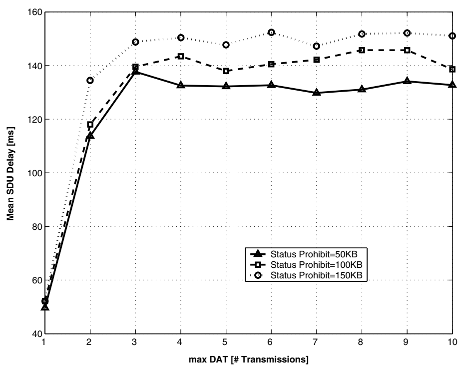
Fig. 3(a) shows the dependency of the download time on maxDAT and status prohibit, for a fixed timer poll value of $100ms$. The emulations were done for a $100KBytes$ file. As can be seen from the figure, for a given status prohibit value, the download time increases as the maxDAT value decreases. For the status prohibit values of $50ms$, $100ms$ and $150ms$, this increase in download



(a) Mean file download time



(b) Status overhead



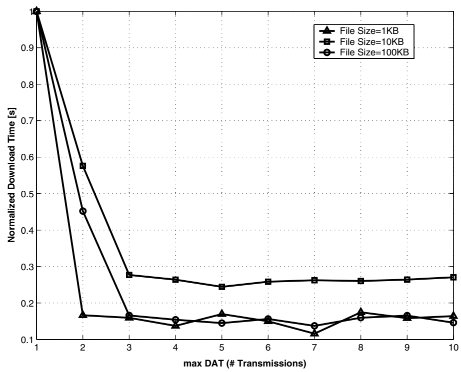
(c) Mean SDU delay

Fig. 3. Results for different maxDAT and Status Prohibit for a $100KBytes$ file download

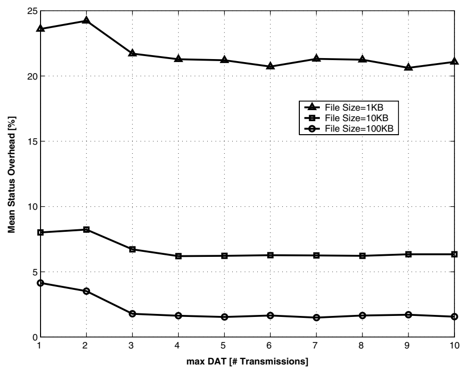
time is 800%, 647% and 589%, respectively, as maxDAT decreases from 10 to 1 [2]. This is an expected result as the main idea behind link layer retransmissions is to decrease the probability of TCP timeouts by retransmitting a subset of the packet, that is the RLC PDUs.

When it is large, maxDAT is the dominating factor and the effect of status prohibit is nullified. This is because even though status prohibit is small and leads to spurious retransmissions as too many status reports arrive due to missing PDU detection, maxDAT is high enough to prevent the untimely discard of the SDU. This conclusion will not hold true if the status prohibit is set to a very

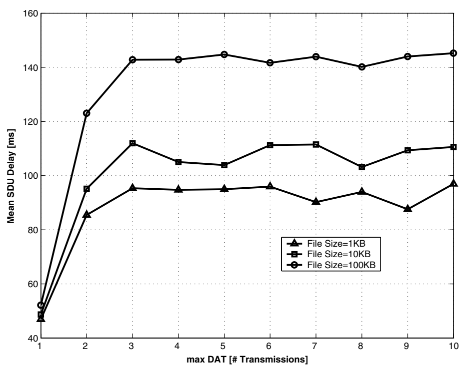
[2] Setting maxDAT to 1 is equivalent to disabling retransmission, and hence nullifying the RLC reliability mechanism



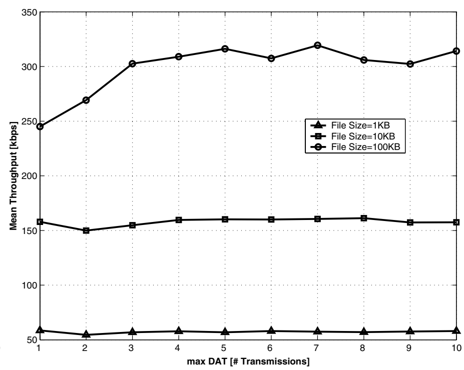
(a) Mean file download time



(b) Status overhead



(c) Mean SDU delay



(d) Mean throughput

Fig. 4. Results for different maxDAT and file sizes

high value, as that would have a considerable impact even if maxDAT is high, by causing status reporting to be delayed for unnecessarily longer periods. For maxDAT values below 3, status prohibit has a noticeable impact on the download time. When the status prohibit value is very low, the frequency of status reporting becomes high, increasing the probability that a PDU is retransmitted. An increase in the retransmission rate will make the retransmission count reach the maxDAT faster, leading to SDU discard. The SDU discard will lead to TCP timeouts, and hence a decrease in the link efficiency, i.e increase in download time.

In fig. 3(b) the effect of the same parameter combinations as the previous case are shown but for the status overhead. This shows that the lower the status prohibit, the higher the status overhead. As maxDAT increases, the status overhead also decreases, in a trend is similar to the download time. This is because for lower maxDAT values, there are a lot of status reports that are wasted completely because the SDUs are discarded anyway even though retransmission requests are coming.

Fig. 3(c) shows the effect of maxDAT and status prohibit on the SDU delay. The SDU delay values are very low for small values of maxDAT. This is because when the maxDAT is very low, there are a lot of SDUs that are discarded and the only SDUs that will account for the mean SDU delay value calculation are the ones that are completed with fewer PDU being retransmitted. For example, for the maxDAT value of 1, the only SDUs that are taken into account for the SDU delay calculation are the ones that are received without any of their PDUs being retransmitted. As the maxDAT value increases, the SDU delay increases, as more and more SDUs are being received properly, mostly with some of their PDUs being retransmitted. After maxDAT reaches 3 the SDU delay value remain almost the same. The effect of the status prohibit for this case is the reverse of what is seen for the download time and status overhead. With low status prohibit value, the retransmission requests come to the sender in a quick succession, increasing the rate of retransmission of PDUs, and hence decreasing the total time required to transmit a given SDU. Though this has a good effect from an SDU delay point of view, it can have a negative effect (and it has for the cases that are investigated here) on the overall file download session, as a most of the bandwidth will be used for retransmissions instead of for first time transmissions, and a lot of battery power will be spent by the mobile terminal through frequent status reporting.

Figures [4(a)-4(c)] also show the dependency of the download time, status overhead and SDU delay on maxDAT, for different file sizes, while the timer poll and the status prohibit are fixed at $300ms$ and $100ms$, respectively. The download times are normalized to the maximum for each file size. From the figures it can be seen that the trend is the same as in the previous cases, i.e. an increase in maxDAT leads to a decrease in the download time, a decrease in the status overhead and an increase in the SDU delay, for a given file size.

For small file sizes, as can be seen in fig. 4(b), the status overhead is the greatest as the chances of cumulatively acknowledging a lot of PDUs at once is very low. However, the SDU delay is the lowest for small file sizes as the probability of an SDU being queued in the RLC transmission buffer before we can start sending it for the very first time is very low.

Fig. 4(d) shows the evolution of the mean throughput as a function of maxDAT for different file sizes. It can be seen that the file size greatly affects the bandwidth utilization, the utilization factor ranging from 15% for the $1KBytes$ case up to 81% for the $100KBytes$ case. The main reason behind this is that for small file sizes, the file download is completed before the TCP connection is able to get out of the initial cycles of TCP slow start.

5 Conclusion

In this paper, the performance of FTP file download over a UMTS dedicated channel, under the assumption of constant bit rate and uncorrelated errors has been investigated. It is found that the main determining factor is the maxDAT value, while the status prohibit value plays a minor role when the maxDAT is not high enough. No disadvantage of setting maxDAT to a higher value is found for the investigated cases. However, a definite conclusion can not be given unless extensive investigations are carried out considering several issues such as correlated errors and advanced TCP retransmission mechanisms such as Selective Acknowledgments (SACK). Such interactions may lead to redundant simultaneous retransmissions at the RLC and TCP layer, therefore diminishing the advantages of high maxDAT values, or even turning it into a disadvantage. Also, the performance may be different if other type of services such as streaming are considered. In the future, we want to consider the aforementioned factors to arrive at a definite conclusion on the effects of the RLC mechanisms and their parameters settings.

References

1. Greg Miller and Kevin Thompson. The Nature of the Beast: recent Traffic Measurements from an Internet Backbone. http://www.caida.org/outreach/papers/1998/Inet98/Inet98.html.
2. Jon Postel. *RFC 793: Transmission Control Protocol*, September 1981.
3. M. Allman and V. Paxson and W. Stevens. *RFC 2581:TCP Congestion Control*, April 1999.
4. 3GGP TS 25.322 v5.4.0. *RLC Protocol Specification*, March 2003.
5. Malek Boussif, Oumer M. Teyeb, Troels Sørensen, Jeroen Wigard, and Preben M. Mogensen. RESPECT: A Real-time Emulator for Service Performance Evaluation in Cellular neTworks. In *Vehicular Technology Conference, Fall*, 2005. submitted for publication.
6. Harri Holma and Antti Toskala. *WCDMA for UMTS, Radio Access for Third Generation Mobile Communications*. John Wiley & Sons, 2004.
7. 3GGP TS 25.853 v4.0.0. *Delay Budget within the Access Stratum*, March 2001.

Uni-source and Multi-source m-Ary Tree Algorithms for Best Effort Service in Wireless MAN

Jin Kyung Park, Woo Cheol Shin, Jun Ha, and Cheon Won Choi

Dankook University, Seoul, Korea
cchoi@dku.edu

Abstract. IEEE 802.16 WirelessMAN standard specifies the air interface of broadband wireless access systems providing multiple services. In the wireless MAN, the best effort service class is ranked on the lowest position in priority and is assisted by a MAC scheme based on reservation ALOHA. In such a MAC scheme, a collision of resource requests is unavoidable so that wireless MAN standard adopted a truncated binary exponential backoff algorithm to arbitrate request attempts. However, it was revealed that a truncated binary exponential backoff algorithm may deteriorate delay and throughput performance due to its capture or starvation effect. Aiming at improving such performance, we propose uni-source and multi-source m-ary tree algorithms as alternatives to resolve request collisions in a wireless MAN. For the uni-source m-ary tree algorithm, we first develop an analytical method to calculate the maximum throughput. Secondly, using the analytical method as well as simulation method, we evaluate maximum throughput, mean and variance of MAC PDU delay. From numerical results, we confirm that proposed algorithms invoke superior delay and throughput performance to a truncated binary exponential backoff algorithm.

1 Introduction

IEEE 802.16 WirelessMAN standard specifies the air interface of fixed point-to-multipoint broadband wireless access systems providing multiple services in a wireless metropolitan area network [3][5]. Between a base station (BS) and subscriber stations (SS's), the wireless MAN supports four service classes identified as unsolicited grant service, real-time polling service, non-real-time polling service, and best effort service. Among the service classes supported by the wireless MAN, the best effort service class is ranked on the lowest position in priority and is usually assisted by a medium access control (MAC) scheme based on reservation ALOHA.

In the wireless MAN operating in time division duplexing (TDD) mode, time is divided into frames and each frame consists of uplink and downlink subframes. In such a time structure, an SS using the best effort service sends a request message during a part of a uplink subframe (identified as request opportunity) and informs the BS of its demand for resource to send MAC protocol data units

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 397–405, 2005.

(PDU's). If two or more SS's attempt requests on a same request opportunity, a collision occurs among the requests and the SS's that involved in the collision attempt requests again later. To suppress repeated collisions, a collision resolution algorithm is needed to arbitrate request attempts and IEEE 802.16 Wireless-MAN standard adopted a truncated binary exponential backoff algorithm [5]. However, it was revealed that a truncated binary exponential backoff algorithm inherently causes capture or starvation effect, which in turn deteriorates delay and throughput performance [8]. Thus, aiming at improving delay and throughput performance, we present two algorithms as alternatives to resolve request collisions in the wireless MAN.

Concerning a collision resolution algorithm for the best effort service in the wireless MAN, we note three points: First, it is desirable that an algorithm is cooperative with the feedback information for truncated binary exponential backoff algorithm. Secondly, an algorithm should be able to support any number of request opportunities per uplink subframe. Finally, the delay and throughput performance is a critical factor in designing a collision resolution algorithm since scarce resource may be available for the best effort service after resource is preferrably allocated to other service users.

An m-ary tree algorithm is a collision resolution algorithm, where the users involved in a collision is randomly partitioned into m groups and the users belonging to the first group exclusively take the incoming chance of transmitting data [7]. Since an m-ary tree algorithm was introduced [1], a number of variants have been reported [7]. However, most of these algorithms are only applicable to the frame structure in which a single request opportunity is provided per frame. IEEE 802.14 HFC standard also adopted a ternary tree algorithm [2]. The algorithm, however, requires centralized control information.

In this paper, with accomodating the three points, we propose two collision resolution algorithms identified as uni-source and multi-source m-ary tree algorithms. First, we develop an analytical method to exactly calculate the maximum throughput induced by the uni-source m-ary tree algorithm. Secondly, using the analytical method as well as simulation method, we investigate maximum throughput, mean of MAC PDU delay and variance of MAC PDU delay exhibited by each proposed algorithm in various environments.

In section 2, we describe a MAC scheme for the best effort service in the wireless MAN. In section 3, we present two algorithms (uni-source and multi-source m-ary tree algorithms) for arbitrating request attempts and resolving request collisions. In section 4, for the uni-source m-ary tree algorithm, we present an analytical method to calculate the maximum throughput. In section 5, we evaluate the delay and throughput performance of the proposed algorithms in comparison with a truncated binary exponential backoff algorithm.

2 MAC Scheme for Best Effort Service

In a wireless MAN, the best effort service is usually supported by a MAC scheme based on reservation ALOHA. Such a MAC scheme must include a number of

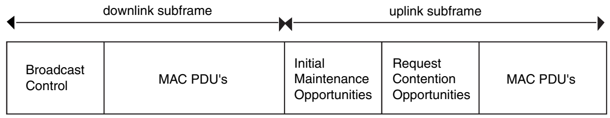


Fig. 1. Frame structure in wireless MAN

details, while many of them are not specified [3]. Recently, for the best effort service in a wireless MAN, candidate MAC schemes were proposed in [6]. In this section, referring to the details in [6], we construct a MAC scheme for the best effort service.

In the wireless MAN operating in TDD mode, time is divided into frames and each frame consists of uplink and downlink subframes. (Figure 1 shows a simplified frame structure in the wireless MAN.)

A request contention field consists of a number of request opportunities. Prior to sending MAC PDU's, an SS chooses a request opportunity and attempts to send a request for resource (to transmit MAC PDU's) using the selected opportunity. In our MAC scheme, an SS is only allowed to make a new request attempt after the previous attempt is either positively or negatively acknowledged. Also, whenever an SS attempts a request, the SS is allowed to demand a limited amount of resource. The maximal amount of resource is prescribed to be the amount of resource for transmitting a single MAC PDU. If the request fails (due to collision), the SS re-attempts a request later according to a collision resolution algorithm. Otherwise, the request is stored at a buffer residing in the BS and is positively acknowledged through a broadcast control field. Following the FCFS service discipline, the BS selects a request from the buffer and grants resource to the request. In our MAC scheme, a request is granted as much resource as it demands. However, if the available resource in an uplink subframe is insufficient, the BS grants resource in the next uplink subframe to the request. Upon reception of resource grant information via broadcast control field, the SS transmits PDU's using the allocated resource.

3 Collision Resolution Algorithms

In this section, we describe the uni-source and multi-source m-ary tree algorithms. In these algorithms, there are a number of groups (including idle group) and each SS belongs to a certain group at any time. The BS releases the result of request attempt in each request opportunity as request success, request collision or no request. According to the result of request attempt, SS's are re-partitioned into a number of groups. For the description of the algorithms, we suppose that K SS's use the best effort service and J request opportunities are provided at each uplink subframe in the wireless MAN.

3.1 Uni-source m-Ary Tree Algorithm

Let $T^{(n)}$ denote the start time of request opportunities in the nth uplink subframe for $n \in \{1, 2, \cdots\}$. Let $V^{(n)}$ denote the number of groups (excluding the idle group) and $G_i^{(n)}$ be the ith group at time $T^{(n)}-$ for $i \in \{1, \cdots, V^{(n)}\}$. Also, set $G_0^{(n)}$ to be the idle group at time $T^{(n)}-$.

1. Suppose that $V^{(n)} \in \{1, 2, \cdots\}$ at time $T^{(n)}-$ for $n \in \{1, 2, \cdots\}$. Then, each SS belonging to group $G_1^{(n)}$ independently chooses a request opportunity in the nth uplink subframe with probability $1/J$, and attempts a request using the selected opportunity.

Suppose that $V^{(n)} = 0$ at time $T^{(n)}-$. Then, all K SS's belong to the idle group $G_0^{(n)}$. If there are some SS's which are loaded with MAC PDU's (for which they have to attempt resource requests) in group $G_0^{(n)}$ at time $T^{(n)}-$, each of these SS's chooses a request opportunity in the nth uplink subframe with probability $1/J$, and attempts a request using the selected request opportunity.

2. Suppose that at least one collision occurred in the nth uplink subframe. Then, each of the SS's that failed in request chooses a number in $\{1, \cdots, m\}$ with probability $1/m$. If an SS chooses $i \in \{1, \cdots, m\}$, the SS joins group $G_i^{(n+1)}$. On the other hand, the SS's that succeeded in request return to the idle group $G_0^{(n+1)}$. If $V^{(n)} \in \{2, 3, \cdots\}$, then each SS belonging to group $G_i^{(n)}$ moves to group $G_{i+m-1}^{(n+1)}$ for $i \in \{2, \cdots, V^{(n)}\}$. Thus, $V^{(n+1)} = V^{(n)} + m - 1$.

Suppose that no collision occurred in the nth uplink subframe. Then, the SS's that succeeded in request return to the idle group $G_0^{(n+1)}$. If $V^{(n)} \in \{2, 3, \cdots\}$, then each SS belonging to group $G_i^{(n)}$ moves to group $G_{i-1}^{(n+1)}$ for $i \in \{2, \cdots, V^{(n)}\}$. Thus, $V^{(n+1)} = \max\{0, V^{(n)} - 1\}$.

3.2 Multi-source m-Ary Tree Algorithm

In the multi-source m-ary tree algorithm, there are a number of groups associated with each request opportunity. Recall that $T^{(n)}$ is the start time of request opportunities in the nth uplink subframe. Let $V_j^{(n)}$ denote the number of groups associated with the jth request opportunity and $G_{j,i}^{(n)}$ be the ith group associated with the jth request opporutnity at time $T^{(n)}-$. Also, set $G_0^{(n)}$ to be the idle group at time $T^{(n)}-$ and $U^{(n)} = \sum_{j=1}^{J} I_{\{V_j^{(n)}=0\}}$ for $n \in \{1, 2, \cdots\}$.

1. Suppose that there is a request opportunity $j \in \{1, \cdots, J\}$ such that $V_j^{(n)} \in \{1, 2, \cdots\}$. Then, each SS belonging to group $G_{j,1}^{(n)}$ attempts a request at the jth request opportunity in the nth uplink subframe.

Suppose that in the idle group $G_0^{(n)}$, there are some SS's which are loaded with MAC PDU's (for which they have to attempt requests) at time $T^{(n)}-$. If $U^{(n)} \in \{1, \cdots, J\}$, each of the SS's independently chooses one among the request opportunities such that $V_j^{(n)} = 0$ with probability $1/U^{(n)}$, and attempts a request using the selected request opportunity.

2. Suppose that a collision occurred at the jth request opportunity in the nth uplink subframe. Then, each of the SS's that involved in the collision independently chooses a number in $\{1, \cdots, m\}$ with probability $1/m$. If an SS chooses $i \in \{1, \cdots, m\}$, the SS joins $G_{j,i}^{(n+1)}$. If $V_j^{(n)} \in \{2, 3, \cdots\}$, each SS belonging to group $G_{j,i}^{(n)}$ moves to group $G_{j,i+m-1}^{(n+1)}$ for $i \in \{2, \cdots, V_j^{(n)}\}$. Thus, $V_j^{(n+1)} = V_j^{(n+1)} + m - 1$.

Suppose that no collision occured at the jth request opportunity in the nth uplink subframe. Then, the SS which attempted a request using the jth request opportunity in the nth uplink subframe, if any, returns to the idle group $G_0^{(n+1)}$. If $V_j^{(n)} \in \{2, 3, \cdots\}$, each SS belonging to group $G_{j,i}^{(n)}$ is transfered to group $G_{j,i-1}^{(n+1)}$ for $i \in \{2, \cdots, V_j^{(n)}\}$. Thus, $V_j^{(n+1)} = \max\{V_j^{(n)} - 1, 0\}$.

4 Maximum Throughput Calculation

In this section, we present an analytical method to exactly calculate the maximum throughput induced by the uni-source m-ary tree algorithm. A request is said to depart from the SS if the request does not collide and hence succeeds. Also, the PDU service rate at the BS is defined as the average number of PDU's that can be transmitted by use of the available resource for the best effort service in an uplink subframe. Note that the aggregated rate of request departures from all SS's is equal to the request arrival rate at the BS. Thus, the maximum throughput is yielded by taking the minimum of the maximum aggregated rate of request departures and PDU service rate. For the calculation of maximum throughput, we assume that K SS's use the best effort service and each of the SS's is saturated, i.e., an SS has infinite number of MAC PDU's to send at any time. We also assume that J request opportunities are provided in each uplink frame.

In the uni-source m-ary algorithm, all K SS's attempt requests together in some uplink subframes. Let C_k denote the index of the uplink subframe in which all K SS's attempt requests together in the kth time. Recall that $T^{(n)}$ indicates the start time of the request opportunities in the nth uplink subframe and $V^{(n)}$ is the number of groups at time $T^{(n)}-$. Then, $C_0 \triangleq 1$ and

$$C_k = \min\{n \in \{C_{k-1} + 1, C_{k-1} + 2, \cdots\} : V^{(n)} = 0\} \tag{1}$$

for $k \in \{1, 2, \cdots\}$. Note that an SS which attempted a request in the C_kth uplink subframe is not allowed to make a new request attempt until the $(C_{k+1} - 1)$st uplink subframe. (Once an SS succeeds in request, the SS returns to the idle group and remains at the idle group until $T^{(C_{k+1})}-$.) Thus, every SS succeeds in request exactly one time in the interval $[T^{(C_k)}, T^{(C_{k+1})})$. Define $B_K^{(k)} \triangleq C_k - C_{k-1}$ for $k \in \{1, 2, \cdots\}$. Then, for given K, the sequence $\{B_K^{(k)},\ k = 1, 2, \cdots\}$

is independent and identically distributed (i.i.d.). Let B_K be a random variable such that $B_K^{(k)} \stackrel{d}{=} B_K$ for all $k \in \{1, 2, \cdots\}$. Set $\beta_K \stackrel{\triangle}{=} E(B_K)$. From the above arguement, the maximum aggregated rate of request departures, denoted by δ is then expressed as

$$\delta = \lim_{n\to\infty} \frac{nK}{\sum\limits_{k=1}^{n} \left[T^{(C_k)} - T^{(C_{k-1})}\right]} = \frac{K}{\tau\beta_K} \tag{2}$$

where τ is the frame duration time.

Suppose that collisions occurred in the C_kth uplink subframe. Then, the SS's that involved in the collisions are partitioned into m groups denoted by $G_1^{(C_k+1)}, \cdots, G_m^{(C_k+1)}$. Let X be the number of SS's that involved in the collisions and Y_i be the number of SS's which belong to group $G_i^{(C_k+1)}$ for $i \in \{1, \cdots, m\}$. Consider the number of uplink subframes which are passed until all Y_i SS's (belonging to group $G_i^{(C_k+1)}$) ultimately succeed in request. Then, it has the same distribution as B_{Y_i} for all $i \in \{1, \cdots, m\}$. Thus, we have the following relation of $\{B_K,\ K = 1, 2, \cdots\}$:

$$B_K \stackrel{d}{=} 1 + \sum_{i=1}^{m} B_{Y_i} \cdot I_{\{X\in\{1,\cdots,K\}\}}\ . \tag{3}$$

Let $h(K, J, q)$ denote the probability that q requests succeed when K request attempts are made on J request opportunities in a same uplink subframe. Then, the random variable X has the mass such that $P(X = r) = h(K, J, K - r)$ for $r \in \{0, \cdots, K\}$. Note that $h(K, J, q)$ is equal to the probability that the number of boxes containing exactly one ball is q when K balls are put into J boxes. From [4], we have

$$h(K, J, q) = \frac{(-1)^q J! K!}{q! J^K} \sum_{l=q}^{\min\{K,J\}} \frac{(-1)^l (J - l)^{K-l}}{(l - q)!(J - l)!(K - l)!} \tag{4}$$

for $q \in \{0, \cdots, \min\{K, J\}\}$. Since each SS that involved in a collision independently chooses one of the m groups with probability $1/m$, the random vector $(Y_1, \cdots, Y_m)$ has the conditional mass as

$$P((Y_1, \cdots, Y_m) = (q_1, \cdots, q_m)\ |\ X = r) = \binom{r}{q_1 \cdots q_m} (\frac{1}{m})^r \tag{5}$$

for $r \in \{0, 1, \cdots\}$ and $(q_1, \cdots, q_m) \in S_r$, where

$$S_r \stackrel{\triangle}{=} \{(j_1, \cdots, j_m) \in \{0, \cdots, r\}^m : j_1 + \cdots + j_m = r\}\ . \tag{6}$$

Set $\beta_0 \stackrel{\triangle}{=} 1$ and $\beta_1 \stackrel{\triangle}{=} 1$. Then, from (3), (4) and (5), we have the following recursive relation of $\{\beta_K,\ K = 2, 3, \cdots\}$:

$$\beta_K = \frac{1}{1 - h(K,J,0)(\frac{1}{m})^{K-1}} \Bigg[h(K,J,K) + h(K,J,0)(\frac{1}{m})^{K-2}$$
$$+ \sum_{r=1}^{K-1} h(K,J,r) \sum_{(q_1,\cdots,q_m)\in S_{K-r}} \left[1 + \sum_{i=1}^{m} \beta_{q_i}\right] \binom{K-r}{q_1 \cdots q_m} (\frac{1}{m})^{K-r}$$
$$+ \ h(K,J,0) \sum_{(q_1,\cdots,q_m)\in S_K^*} \left[1 + \sum_{i=1}^{m} \beta_{q_i}\right] \binom{K}{q_1 \cdots q_m} (\frac{1}{m})^{K} \Bigg] \tag{7}$$

for all $K \in \{2,3,\cdots\}$, where $S_r^* \triangleq \{(j_1,\cdots,j_m) \in \{0,\cdots,r-1\}^m : j_1+\cdots+j_m = r\}$. Let γ be the MAC PDU service rate at the BS. Then, we finally have the maximum throughput induced by the uni-source m-ary tree algorithm, denoted by η as

$$\eta = \min\{\delta,\gamma\} = \min\{\frac{K}{\tau\beta_K},\gamma\} \tag{8}$$

from (2).

5 Performance Evaluation

In this section, using the analytical method as well as simulation method, we evaluate the delay and throughput performance exhibited by each of the three collision resolution algorithms: uni-source m-ary tree, multi-source m-ary tree and truncated binary exponential backoff algorithms. The environment assumed in this section is as follows: In the wireless MAN, 10 SS's use the best effort service. The duration times of downlink and uplink subframes are equal to 1000 minislots. (Thus, the frame duration time is equal to 2000 minislots.) In an uplink subframe, the duration time of a request opportunity is 8 minislots and 16 minislots are assigned for initial maintenance opportunities. Also, the amount

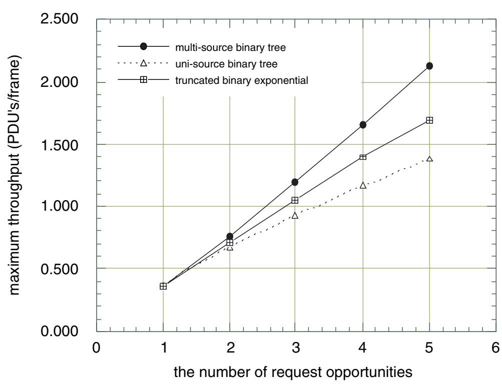


Fig. 2. Maximum throughput vs. the number of request opportunities

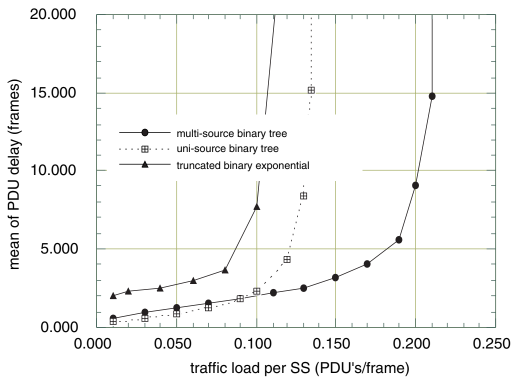


Fig. 3. Mean of PDU delay time vs. traffic load per SS

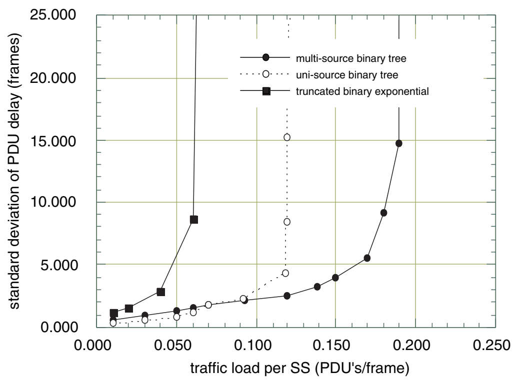


Fig. 4. Standard deviation of PDU delay time vs. traffic load per SS

of resource allocated for other services except best effort service has the uniform distribution, where the mean is equal to the 10% of the total amount of resource in an uplink subframe and the standard deviation is 2%. At each SS, the transmission time of each MAC PDU is fixed to 48 minislots and the sequence of MAC PDU arrival times is modeled as a mutually independent Bernoulli point process. In addition, we assume a truncated binary exponential backoff algorithm in which the number of request opportunities that an SS must deny prior to making the nth attempt of a same request has the uniform distribution in $\{0, \cdots, 8 \cdot 2^n\}$, and an SS renounces a request if the SS fails in the 16th attempt of the request.

In figure 2, we show the maximum throughput with respect to the number of request opportunities. In this figure, we compares the uni-source binary tree, multi-source binary tree and truncated binary exponential backoff algorithms, and observe that the multi-source binary tree algorithm invokes the highest maximum throughput. In figures 3 and 4, we compare the uni-source binary tree, multi-source binary tree and truncated binary exponential backoff algorithms in delay performance. In these figures, we observe that the multi-source binary tree algorithm invokes superior delay performance to other algorithms. Note that the maximum throughput of the uni-source binary tree algorithm was shown to be lower than the maximum throughput of the truncated binary exponential backoff algorithm in figure 2. However, we notice that the uni-source binary tree algorithm exhibits better delay performance than the truncated binary exponential backoff algorithm.

6 Conclusions

In provisioning best effort service at a wireless MAN, request collisions are unavoidable so that a collision resolution algorithm is needed to suppress repeated collisions. In this paper, aiming at improving delay and throughput performance, we proposed the uni-source and multi-source m-ary tree algorithms as alternatives to truncated binary exponential backoff algorithm. For the uni-source m-ary tree algorithm, we first developed an analytical method to eactly calculate the maximum throughput. Secondly, using the analytical method as well as simulation method, we evaluated each proposed algorithm in maximum throughput, mean of PDU delay time and variance of PDU delay time. From numerical examples, we observed that the multi-source binary tree algorithm produces higher maximum throughput than a truncated binary exponential backoff algorithm. Moreover, both of proposed algorithms were shown to invoke superior delay performance than a truncated binary exponential backoff algorithm.

References

1. J. Capetanakis, "Tree Algorithm for Packet Broadcast Channels," *IEEE Transactions on Information Theory*, vol. 25, no. 5, pp. 505-515, September 1979.
2. M. Corner, N. Golmie, J. Liebeherr, and D. Su, "A Priority Scheme for the IEEE 802.14 MAC Protocol for Hybrid Fiber-Coax Networks," *IEEE/ACM Transactions on Networking*, vol. 8, no. 2, pp. 200-211, April 2000.
3. C. Eklund, R. Marks, K. Standwood, and S. Wang, "IEEE Standard 802.16: A Technical overview of the WirelessMAN Air Interface for Broadband Wireless Access," *IEEE Communications Magazine*, vol. 40, no. 6, pp. 98-107, June 2002.
4. W. Feller, *An Introduction to Probability Theory and Its Applications*. John Wiley & Sons, 1968.
5. IEEE 802.16c-2002, "IEEE Standard for Local and metropolitan area networks-Part 16: Air Interface for Fixed Broadband Wireless Access Systems-Amendment 1: Detailed System Profiles for 10-66 GHz," January 2003.
6. J. Park, W. Shin, J. Ha, and C. Choi, "Maximal Grant for Resource Request Supporting Best Effort Service in Wireless MAN," *Proceedings of International Conference on Networking*, pp. 356-362, 2004.
7. R. Rom and M. Sidi, *Multiple Access Protocols Performance and Analysis*. Springer-Verlag, 1990.
8. B. Whetten, S. Steinberg, and D. Ferrari, "The Packet Starvation Effect in CSMA/CD LANs and a Solution," *Proceedings of the 19th Conference of Local Computer Networks*, pp. 206-217, 1994.

High Rate UWB-LDPC Code and Its Soft Initialization

Jia Hou and Moon Ho Lee

Institute of Information & Communication, Chonbuk National University,
Chonju, 561-756, Korea
{jiahou, moonho}@chonbuk.ac.kr

Abstract. In this paper, we describe a kind of high rate low density parity check (LDPC) code and its decoding scheme for ultra wideband (UWB) communication. Particularly, the proposed scheme uses a hybrid soft normalization algorithm from the autocorrelations of UWB signals to initialize the LDPC decoder, and it integrates several normalized soft values of UWB signals as sine (BPSK) or cosine (PPM) elements by using the window concept.

1 Introduction

LDPC codes with large block length and low rate have been shown to have record breaking performance for low signal-to-noise applications. The high rate LDPC codes are also excellent, outperforming comparable BCH and RS codes even at short block length. For certain applications such as magnetic recording, high rate LDPC codes at short block length are of particular interest [1,2]. In future, these LDPC codes will be applied to provide high speed and high quality communication. Otherwise, UWB technique which has lower power and higher rate is attracted much attention for short range networking [4,5]. Recent results indicate that UWB radio is viable candidate for short range multiple access communications in dense multi-path environments [8], exploiting the advantages of the UWB's fine time resolution properties [9]. It is therefore desirable to find a high rate channel coding scheme with short block length. In this paper, we propose a simple high rate and short block length regular LDPC codes which is suitable for high speed packet transmission. In particular, this paper looks in more detail at a random and systematic method to construct high rate regular LDPC code, addressing the following issues. First, we specify the parity check matrix of a randomly systematically constructed LDPC (SC-LDPC) code [2]. Based on several sub-matrices from shift registers, we present a regular method and a simple extension way to obtain the high performance parity check matrix without four cycles. Next, we investigate the combination of UWB signals with binary LDPC codes by using a simple normalized soft initialization on sine or cosine values from the UWB decorrelator. The proposed scheme can accurately represent the UWB transmitted signals as a suitable soft value for LDPC iterative decoder. Finally, a conclusion is drawn.

2 Description of High Rate LDPC Construction

The recent papers [1] reported high rate binary LDPC codes are shown a near approach to Shannon limit in AWGN channel. In order to design a high rate code for

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 406 – 412, 2005.

Regular Method | Simple Extension Method

$$
\left[\begin{array}{cccc:cccc:cccc:cccc}
1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&1&1&1&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&1&1&1&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1\\
\hline
1&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0\\
0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&0\\
0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0\\
0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1\\
\hline
1&0&0&0&0&1&0&0&0&0&1&0&0&0&0&1\\
0&1&0&0&1&0&0&0&0&0&0&1&1&0&0&0\\
0&0&1&0&0&0&0&1&0&1&0&0&0&1&0&0\\
0&0&0&1&0&0&1&0&1&0&0&0&0&0&1&0
\end{array}\right]
\begin{array}{cccc}
1&0&0&0\\
1&0&0&0\\
1&0&0&0\\
0&1&0&0\\
0&1&0&0\\
0&1&0&0\\
0&0&1&0\\
0&0&1&0\\
0&0&1&0\\
0&0&0&1\\
0&0&0&1\\
0&0&0&1
\end{array}
$$

Fig. 1. Regular construction of SC-LDPC codes and simple extension

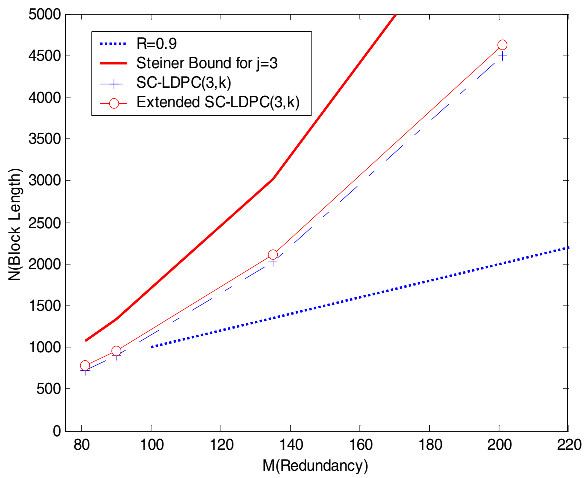


Fig. 2. High rate SC-LDPC codes, block length and its relation Steiner bound, with $R > 0.9, N < 5000$, $j = 3$

UWB transmission, we now investigate a column weight j, row weight k, redundancy M and short block length N regular LDPC code construction. Particularly, the systematically LDPC, we confine ourselves hereafter to $j = 3$, because in the context of Steiner systems [1,3], these codes are much interesting. The small cycles, i.e. four cycles, prevent binary codes from achieving maximum likelihood performance with iterative decoding algorithm, therefore, the LDPC code's criterion is closely related to the Steiner bound. As illustrated in Fig.1, the simple systematically constructed LDPC codes are composed of j horizontal sub-matrices of size $(M / j) \times N$ which can be easily implemented by using the shift registers. The first sub-matrix consists of the k squared ones matrix, the second is identity matrix and the third consists of the identity matrix of size $k \times k$ and its $(k-1)$ cyclically shifted versions. Thus the SC-LDPC of size $M \times N = jk \times k^2$ can be derived as [3], according to the Steiner bound,

$$Ns = M(M-1)/j(j-1) = (jk)(jk-1)/j(j-1), \tag{1}$$

where the code rate $R = 1 - \frac{M}{N} = \frac{k-j}{k}$. It is suitable to design such code with high rate and short length ($R > 0.9, N < 6000$) to approach the Steiner bound, as shown in Fig.2. A simple extension can be applied to obtain the lower density and higher rate, as illustrated in Fig.1. The simulation shows that the extension method could take a tighter approach to the Steiner bound. Moreover, in short range and high speed networking system, UWB signals always are transmitted as short packet and higher rate. Therefore, we now design SC-LDPC codes with length $N < 6000$, $R > 0.9$ for UWB systems.

3 Soft Normalization for UWB Transmission

Recently, UWB system is described in which the transmitted signal occupies an extremely large bandwidth even in the absence of data modulation [4,5,6]. In this case, a signal is transmitted with a bandwidth much larger than the modulation bandwidth and thus with a reduced power spectral density. This approach has the potential to produce a signal that is more covert, has higher immunity to interference effects, and has improved time of arrival resolution [7,8,9]. These systems use pulse amplitude or pulse position modulation (PPM), and different pulse generation methods, pulse rate and shape, center frequency and bandwidth. Most of these systems generate and radiate the impulse response of a wideband microwave antenna and use that reponse as their basic pulse shape by exploiting the BPSK or PPM. Assuming the UWB signal is a modulated train $p(t)$ of Gaussian pulses $s(t)$, spaced in time,

$$p(t) = \sum_{n=0}^{N-1} s(t - nT) e^{-j\frac{2\pi C(n)t}{Tc}}, \tag{2}$$

where T is the repetition period of the pulse train $p(t)$, $C(n)$ is a permutation of integers $\{0,1,...,N-1\}$ for time hopping, and $Tc = \frac{T}{N}$. A typical Gaussian monocycle signal is given as [7]

$$s(t) = \left[1 - \left(\frac{t}{\sigma}\right)^2\right] \exp\left[-\frac{t^2}{2\sigma^2}\right], \tag{3}$$

where σ is a pulse width parameter. Further, we denote a UWB BPSK signal as

$$x(k) = b_k p(t), \tag{4}$$

where b_k is k th transmitted BPSK symbols chosen from $\{\pm 1\}$. Thus the decision of received UWB signals is from the autocorrelations of the pulse generator function as

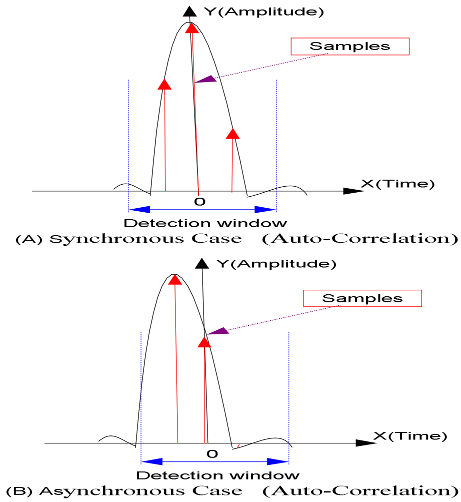


Fig. 3. Soft normalization sine sampling algorithm

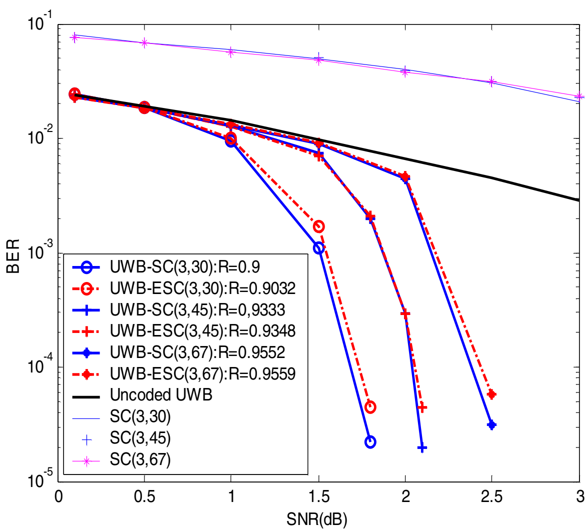


Fig. 4. Performance of UWB-SC-LDPC codes and UWB-ESC-LDPC codes in AWGN channel

$$Z = \begin{cases} \int_{-\infty}^{\infty} b_i p(t-\delta) p^*(t) dt > 0, if \quad b_i = +1 \\ \int_{-\infty}^{\infty} b_i p(t-\delta) p^*(t) dt < 0, if \quad b_i = -1 \end{cases} . \tag{5}$$

In the case of high speed UWB transmission, a good correlation on asynchronous should be obtained. Besides on designing a Gaussian monocycle signal for good correlations, we need an enlarged soft decision value to detect UWB signals and initialize the iterative decoder. We write the soft value of UWB signal detector as

$$\int_{-\infty}^{\infty} b_i p(t-\delta)p^*(t)dt = \sum_{n=sample} b_i p(n-\delta)p^*(n)\,. \tag{6}$$

Assuming the Y axis part of $b_i p(n-\delta)p^*(n)$ is Y_n and the X axis part is X_n, as shown in Fig.3. We present a decision rule by using sine normalization for BPSK UWB signals,

$$\begin{cases} \sum\limits_{n=sample} \dfrac{Y_n}{\sqrt{X_n^2+Y_n^2}} > 0, for \quad b_i = +1; \\ \sum\limits_{n=sample} \dfrac{Y_n}{\sqrt{X_n^2+Y_n^2}} < 0, for \quad b_i = -1. \end{cases} \tag{7}$$

Clearly, the normalization is united by one. The soft normalization rule in this paper has robustness for asynchronous autocorrelations, because that the window detection and sampling are used for protecting signal estimation. In addition, the effective time duration of the window is set as [7],

$$T_w = 7\sigma = 0.5ns\,. \tag{8}$$

In this paper, the decision is according to the BPSK symbols by using the phases $\{\pm\}$. When the correlation is existed in perfect synchronous, we can get the largest amplitude at time $"0"$. Otherwise, in the case of asynchronous, we use sampling from the window detection to remain the largest amplitude. If PPM is used, the cosine normalization may be exploited to decision the estimation values. In the case of BPSK symbols, the error bias function in conventional normalization can be shown as

$$\varepsilon_n = \left|1 - E[Y]\right|^2, \tag{9}$$

and the proposed normalization has

$$\varepsilon_c = \left|1 - E\left[\frac{Y}{\sqrt{X^2+Y^2}}\right]\right|^2. \tag{10}$$

Easily, we prove that

$$\varepsilon_c = \left|1 - E\left[\frac{Y}{\sqrt{X^2+Y^2}}\right]\right|^2 = \left|\frac{E\left[\sqrt{X^2+Y^2}\right] - E[Y]}{E\left[\sqrt{X^2+Y^2}\right]}\right|^2$$

$$\leq \frac{\left|E[X]\right|^2}{\left|E\left[\sqrt{X^2+Y^2}\right]\right|^2} \approx \frac{\left|E[X]\right|^2}{\left|E\left[\sqrt{X^2+Y^2}\right]\right|^2} \approx \left.\frac{\varepsilon}{\left|E[Y]\right|^2}\right|_{Y>>X}, \tag{11}$$

where ε is a value near to zero. Let (9) set $\varepsilon_n = \varepsilon$ as the optimal result and the energy of the transmission $\left|E[Y]\right|^2 \geq 1$, it is clearly that the proposed normalization has lower error bias. Further, we define the intial value for LDPC iterative decoder as

$$L(p) = \frac{2}{\delta^2} \sum_{n=sample} \frac{Y_n}{\sqrt{X_n^2 + Y_n^2}}. \tag{12}$$

As illustrated in Fig.4, the numerical results show that UWB and UWB-SC-LDPC systems achieve much enhancement from conventional SC-LDPC codes without UWB transmission. Especially, over about 1.5dB, the UWB-SC-LDPC efficiently combats noise, and UWB extended SC-LDPC codes (ESC-LDPC) have similar performances, but with higher rates. Additionally, IEEE P802.15.03 presented the UWB transmission are between the 1.5dB to 5dB, it is significant approached our numerical results on UWB-SC-LDPC codes. At lower SNR, different high rate cases have similar performances; at higher SNR, the proposed codes show a sharply decreasing before the error limit. As shown in the simulations, the UWB-SC-LDPC and UWB-ESC-LDPC codes ($R = 0.9 \sim 0.96$) have good results without error floor, after 1.5dB. Further, by using different window sizes to detect the UWB signals, the numerical results demonstrate that $T_w = 7\sigma = 0.5ns$ is an efficient parameters for BPSK UWB-SC-LDPC codes.

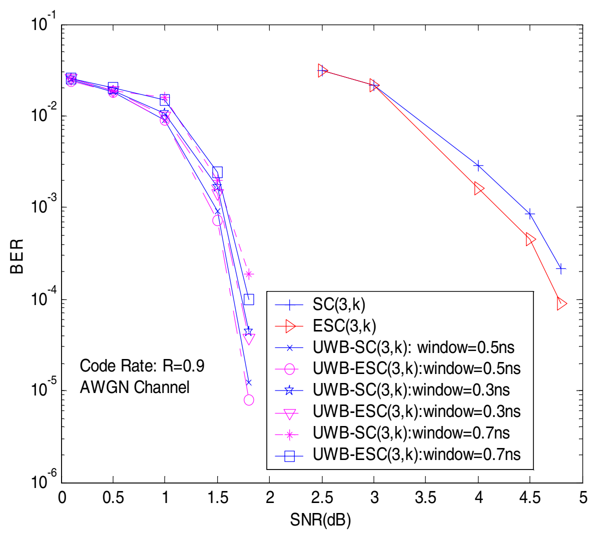


Fig. 5. Performance of UWB-SC-LDPC codes and UWB-ESC-LDPC codes by using different window sizes in AWGN channel

4 Conclusion

We presented a simple high rate LDPC codes for UWB transmission. The distinct advantage of the proposed LDPC codes lies in the fact that the key parameters of the obtained codes, d, j, k, N, M , are easily known. By extensive computer simulations using UWB transmission, it has been observed that no significant difference in the BER between the different rates UWB-SC-LDPC codes and UWB standard at lower SNR ($SNR < 1dB$). However, we achieve about 2dB improvement from UWB after $SNR = 1.5dB$ (IEEE 802.15 transmission model $SNR = 1.5 \sim 5dB$). Generally, high rate UWB-SC-LDPC codes can efficiently gain the best BER with short block length.

Acknowledgement. This work was supposed by university IT research center projects, Ministry of information & Communication, Ministry of Commerce industry & energy, Korea.

References

1. 1. D.J.C Mackay, M.C. Davey: Evaluation of Gallager Codes for Short Block Length and High Rate Applications. Proceedings of IMA workshop on codes, systems and graphical models, (1999) 108–112
2. D. Hosli, E. Svensson, and D. Arnold: High Rate Low Density parity Check Codes: Construction and Application. ISTC 2000, Brest, France, (2000) 111-115
3. D.J.C Mackay: Good Error Correcting Codes Based on Very Sparse Matrices," IEEE Trans. on Information Theory, vol.45 no.3, (1999) 399-431
4. M.Z. Win, and R.A. Scholts: Ultra-Wide Bandwidth Time Hopping Spread Spectrum Impulse Radio for Wireless Multiple Access Communications. IEEE Trans. on Communication, vol.48, no.4, (2000) 679-689
5. M.Z. Win, and R.A. Scholts: Impulse Radio: How It Works. IEEE Comm. Letters. Vol.2, no.2 (1998) 36-38
6. F.R. Mireles: On the Performance of Ultra Wideband Signals in Gaussian Noise and Dense Multipath. IEEE Trans. on Vehicular Technology. Vol.50, no.1, (2001) 244-249
7. L. Zhao, and A.M. Haimovich: Interference Suppression in Ultra Wideband Communications. Proc. of International workshop on UWB, (2001) 759-763
8. H. Luediger, and S. Zeisberg: User and Business Perspective on An Open Mobile Access Standards. IEEE Comm. Magazine, (2000) 160-163
9. M.Z. Win, and R.A. Scholts: On the Robustness of Ultra Wide Bandwidth Signals in Denth Multipath Enviroments. IEEE Comm. Letters. Vol.2, no.2 (1998) 36-38

Cube Connected Cycles Based Bluetooth Scatternet Formation

Marcin Bienkowski[1,⋆], André Brinkmann[2], Miroslaw Korzeniowski[1,⋆], and Orhan Orhan[1]

[1] International Graduate School of Dynamic Intelligent Systems, University of Paderborn, Germany
{young, rudy}@upb.de, orhan@hni.upb.de
[2] Heinz Nixdorf Institute, University of Paderborn, Germany
brinkman@hni.upb.de

Abstract. Bluetooth is a wireless communication standard developed for personal area networks (PAN) that gained popularity in the last years. It was designed to connect a few devices together, however nowadays there is a need to build larger networks. Construction and maintenance algorithms have great effect on performance of the network. We present an algorithm based on Cube Connected Cycles (CCC) topology and show how to maintain the network so that it is easily scalable. Our design guarantees good properties such as constant degree and logarithmic dilation. Besides, the construction costs are proven to be at most constant times larger than any other algorithm would need.

1 Introduction

In this paper we address the problem of network topology construction and maintenance for a wide variety of networks. We require any two nodes to be able to build a bidirectional communication link; for radio networks this can be achieved by placing all the nodes within the communication range. Our topology has a very low requirement for the maximum degree of a node. It is sufficient if the node is capable of communicating with 7 neighbors simultaneously.

The requirements above make the Bluetooth protocol [1] a perfect candidate for our network design. Bluetooth is one of the most recent wireless communication standards developed for Personal Area Networking. Its specification assigns roles of *masters* and *slaves* to nodes. The structure consisting of one master and up to 7 active slaves connected to it is called a *piconet*. Each piconet has a specific frequency-hopping channel which is controlled by its master. For efficiency reasons it is profitable to minimize the number of masters (and thus the number of piconets) and connect two masters not directly, but through a slave, to which we refer later as a *bridge*. Such connection of piconets by bridges can

⋆ Partially supported by DFG-Sonderforschungsbereich 376 "Massive Parallelität: Algorithmen Entwurfsmethoden Anwendungen".

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 413–420, 2005.

establish a large network structure called *scatternet*. Furthermore, the frequency hopping mechanism used by Bluetooth makes the situation, in which a bridge participates in more than two piconets, very undesirable, since the probability of collision between its masters grows very quickly.

An important property of a network is the possibility to maintain a simple routing scheme in it. Neither large routing tables nor long lasting path-finding routines should be used due to bounded network bandwidth and memory of the devices. Last but not least, dynamic scalability of the network should be taken into consideration. This means that nodes can join and leave the network at their convenience without loosing the mentioned characteristics.

In this paper we present a topology which has all the properties mentioned above. We start from the theoretical Cube-Connected-Cycles structure (CCC) [2] and we model it using Bluetooth devices. Each node in the theoretical structure is simulated by a Bluetooth master. Further, if we have a communication link between two nodes in the theoretical structure, we join the two corresponding masters by a bridge. Since in CCC each node has a degree of 3, each master will have 4 spare links which can be used for connecting additional slave nodes.

Among the networks with constant degree, our structure has asymptotically the best possible dilation of $\mathcal{O}(\log n)$; the constant hidden in the $\mathcal{O}$ notation is small. The scalability limits are set by the frequency hopping scheme used by Bluetooth protocol rather than by our topology. The maintenance cost is also optimal. We prove that for any sequence of nodes joining and leaving our network, the cost of our algorithm is at most 18 times larger than the cost of the optimal offline algorithm for the same sequence.

2 Scatternets: Related Work

The problem of scatternet formation for Bluetooth has been intensively studied in the last few years. The proposed algorithms can be categorized into two broad clases. The first group includes those that assume that all devices are in communication range of each other. The algorithms from the second group form a connected network also when this condition is not fulfilled. Due to space limitations we do not go into details for the second group. Check [3, 4, 5, 6] for more information.

Formations for Devices In Range. One of the earliest scatternet formation algorithm studied by Salonidis et al [7] is the Bluetooth Topology Construction Protocol (BTCP), which works only for at most 36 nodes. For a larger number of nodes it proposes a scheme that does not build a fully connected scatternet. Ramachandran et al [8] give two distributed algorithms (one randomized and one deterministic) which build optimal topologies consisting of stars. The issue of choosing bridges to connect the stars is left open. Baatz et al [9] propose a scheme based on composing the topology of k 1-factors (a 1-factor is a graph of maximum degree at most 1, i.e. consisting of independent edges). In each 1-factor one node of an edge is treated as a master and the other as a slave. This topology has an advantage of having multiple active piconets at the same time even if there is overlap between them. However, the roles of masters and slaves are distributed equally which is not desirable for scatternets. The tree scatternet formation (TSF) [10] is a self repairing structure, which organizes nodes into a tree. It allows nodes to arrive and leave arbitrarily. The tree structure guarantees that there are no loops in the network and thus that routing between any pair of nodes is unique. It succeeds in minimizing the number of piconets in the network but is not suitable for larger networks due to high delays in communication. Wang et al [11] define an algorithm called *Bluenet* which aims at constructing a random connected graph. The main disadvantage of this topology is that it lacks any structure which would enable simple routing. Lin et al introduce BlueRing[12] in which the scatternet is based on a ring structure. The architecture has a simple routing and is easy to maintain, however it is only scalable up to a medium sized networks (50-70 nodes). It is unusable for larger networks because of an average dilation and congestion being linear in the size of the network. One of the most advanced approaches in the design of scatternets is BlueCube[13] which proposes a d-dimensional hypercube as a theoretical basis of the network formation. It has a logarithmic dilation, but is only defined for a certain number of nodes. Since the degree of a Bluetooth node is limited to 7, this places also an upper bound on d, limiting the number of nodes in the network to approximately 2^7. The only truly scalable solution we are aware of is proposed in [14], where a network of constant degree and polylogarithmic diameter is constructed. The network is based on a backbone that enables routing based on virtual labeling of nodes without large routing tables or complicated path-discovery methods. The scheme is fully distributed and dynamic in the sense that nodes can join and leave the network at any time.

3 Building and Maintaining Large Scale Bluetooth Scatternets

Our approach is based on a network topology called Cube Connected Cycles. We consider this topology on the basis of the graph theory and adjust it to the Bluetooth specification. First we give a theoretical definition of the topology and show how it can be implemented using Bluetooth devices. Then we present a maintenance algorithm for Bluetooth scatternet based on CCC topology. The algorithm changes the structure instantly when nodes are joining or leaving the system and assures that the number of changes is constant in each step. In the full version of the paper we present another algorithm, which tries not to change the topology as long as possible; the resulting topology updates are large but happen very rarely. The amortized number of changes is even lower than in the case of the smooth maintenance scheme.

Cube Connected Cycles Topology.

Definition 1. *The d-dimensional Cube Connected Cycles network has $d \cdot 2^d$ nodes. The nodes are represented by two indices (i, j), where $0 \leq i < d$ and $0 \leq j < 2^d$. The connectivity is:*

$$(i,j) \rightarrow \begin{cases} (i, j \oplus 2^i) & 0 \le i < d, 0 \le j < 2^d \\ ((i \pm 1) \mod d, j) & 0 \le i < d, 0 \le j < 2^d \end{cases}$$

where $\oplus$ represents the bitwise xor operation. The first set of edges are the cube edges; the second set of edges are the cycle edges.

Observation 1. *The d-dimensional Cube Connected Cycles network has the following properties:*

1. *The number of nodes is $n = d \cdot 2^d$.*
2. *The degree of each node is 3 (or smaller for $d \le 2$).*
3. *The number of edges is $m = \frac{3}{2} \cdot n = 3 \cdot d \cdot 2^{d-1}$ (or smaller for $d \le 2$).*
4. *For any two nodes a and b we can compute a path from a to b of length at most $3 \cdot d$.*

The proof of this observation can be found for example in [2].

If we want to use the CCC topology as a basic interconnection network for the Bluetooth Scatternet formation, we have to be careful and consider the roles for masters and slaves. We propose that nodes in the CCC network are represented by masters in the Scatternet network. Each link from the CCC network will be implemented by a slave (called also a bridge) belonging to two masters and no slave slave will be connected to more than two masters. We can observe that for simulating d-dimensional CCC we need to have at least $5 \cdot d \cdot 2^{d-1}$ nodes ($d \cdot 2^d$ masters and $3 \cdot d \cdot 2^{d-1}$ slaves). It is possible for each master to have 4 additional slaves, thus the upper bound on the number of nodes in d-dimensional network is $13 \cdot d \cdot 2^{d-1}$.

When the number of devices participating in the network exceeds this number, we have to start a process which will rebuild the network. The easiest way would be just to increase d by 1. However, this solution would not work due to the lower bound on the required number of nodes in a $d+1$-dimensional CCC network.

Therefore we introduce an intermediate network topology between the d-dimensional CCC and the $(d+1)$-dimensional CCC. The d-dimensional intermediate CCC network, or in short d-dimensional iCCC network, is defined as follows:

Definition 2. *The d-dimensional iCCC network has $(d+1) \cdot 2^d$ nodes. The nodes are represented by two indices (i,j), where $0 \le i \le d$ and $0 \le j < 2^d$. The connectivity is:*

$$(i,j) \rightarrow \begin{cases} (i, j \oplus 2^i) & 0 \le i < d, 0 \le j < 2^d \\ ((i \pm 1) \mod (d+1), j) & 0 \le i \le d, 0 \le j < 2^d \end{cases}$$

The first set of edges are the cube edges; the second set of edges are the cycle edges.

Compared to the standard CCC definition, the iCCC topology contains an additional ring node (d,j) for each ring of the CCC. This additional ring node is connected to the nodes $(d-1, j)$ and $(0, j)$. As node $(d, j + 2^{(d+1)})$ does not exist, node (d, j) does not have a cube edge.

The properties of the iCCC network are very similar to the properties of the CCC network topology:

Observation 2. *The d-dimensional iCCC network has the following properties:*

1. *The number of nodes is $n = (d+1) \cdot 2^d$.*
2. *The degree of each node is 3 (or smaller for each ring node (d, j) or in case where $d \leq 2$).*
3. *The number of edges is $m = (3 \cdot d + 2) \cdot 2^{d-1}$ (or smaller for $d \leq 2$).*
4. *For any two nodes a and b we can compute a path from a to b of length at most $4 \cdot d$.*

The properties 1 to 3 directly follow from the definition of the d-dimensional iCCC network. Observation 4 can be derived from the properties of a d-dimensional CCC network.

To get from a node (i, j) to a node (u, v), the following path selection strategy can be used. The first part of the path is to get from node (i, j) to node $(0, j)$, which takes at most $d/2$ steps. Then a standard routing scheme for the CCC network, which does not consider iCCC specific nodes (d, j), can be used. To achieve this, almost any dimension-order routing scheme can be used, involving not more than $3 \cdot d$ steps. The last part of the path selection, incurring at most than $d/2$ steps, is to get from node (x, v) to node (u, v). This finishes the proof of Observation 2.

For ease of explanation, we assume that the *CCC* and *iCCC* networks have got a dimension of at least 3. Similar to the lower and upper bounds for Scatternets using the CCC as network topology, the upper and lower bounds for an *iCCC* network are as follows:

$$min_d^{iCCC} = (5 \cdot d + 4) \cdot 2^{d-1} \qquad max_d^{iCCC} = (13 \cdot d + 14) \cdot 2^{d-1}$$

A Smooth Way to Maintain the CCC Topology. Below we introduce a maintenance scheme that will involve a smooth transition from a d-dimensional CCC network over a d-dimensional iCCC network to a $(d+1)$-dimensional CCC topology or vice versa. The different steps of this scheme are displayed in Fig. 1. During the transition, for some time each master will have to simulate the behavior of two nodes in the CCC network. Therefore the degree of a master can grow up to 6. This does not cause any problems, since the Bluetooth specification allows a degree of 7.

The transition from a d dimensional network to a $d+1$ dimensional network involves several steps:

At first we only extend each cycle by an additional master numbered d and transform the network into an iCCC network. Therefore, each time if a new node enters the system and cannot become a loose slave, it extends one of the rings by an additional master d (see Fig. 1.b). To connect to the master nodes 0 and $d-1$, two bridge nodes are required. As the first bridge we can use the slave

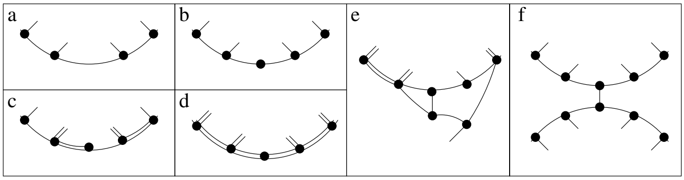


Fig. 1. Transition from a d-dimensional CCC to a $d+1$-dimensional CCC

node that has formerly connected the nodes 0 and $d-1$. As the second bridge we have to take one of the slave nodes of this ring. This transition can be done locally inside each ring. After this step has been made for all rings, the transition to a d-dimensional iCCC is finished.

After the length of each ring has been increased by one, each master acts as two nodes of the $d+1$-dimensional CCC but still has degree 3. From now on, each master wants all of its connections to be doubled. This can also be done gradually as new nodes come and join the network as loose slaves (see Fig. 1.c and 1.d). When a master has doubled all of its connections, it wants to split itself into two nodes, each of them taking over one of the connections from each pair. At this point we distinguish between two types of masters.

A master (d, j) splits itself as soon as it has two loose slaves and both of its edges are doubled. One of its slaves becomes master $(d, j+2^d)$ and the other becomes a bridge between (d, j) and $(d, j+2^d)$. Both pairs of cycle edges are treated in the same way. We describe the procedure for the edges which were both originally connected to $(0, j)$. If the node $(0, j)$ has not split yet, we simply use the edges to connect (d, j) to $(0, j)$ and $(d, j+2^d)$ to $(0, j)$. If it has, we connect (d, j) to $(0, j)$ and $(d, j+2^d)$ to $(0, j+2^d)$ (see Fig. 1.e).

For $i \neq d$, a master (i, j) splits itself as soon as it has a loose slave and all three of its edges (two cycle edges and one cube edge) are doubled. It uses the slave to create master $(i, j+2^d)$ (there will be no connection between (i, j) and $(i, j+2^d)$). One edge from each pair of edges stays connected to (i, j) and the other is connected to $(i, j+2^d)$. To decide which edge is connected to which master, we use the same procedure as for master (d, j). If a node on the other side of the edges has not yet split, we do it arbitrarily. If it has, we do it so that we achieve the following connections: (i, j) with $(i, j \oplus 2^i)$, $((i+1) \mod d, j)$, $((i-1) \mod d, j)$; and $(i, j+2^d)$ with $(i, j \oplus 2^i + 2^d)$, $((i+1) \mod d, j+2^d)$, $((i-1) \mod d, j+2^d)$ (see Fig. 1.e and 1.f).

After all the masters have split, we increase the dimension d by 1.

If a node wants to leave the network, our algorithm works in general inversely to the situation when a node joins the network. The main assumption is that we can exchange the leaving node with any other node in the network. Thus, we can decide which node actually leaves.

Reduction of the network proceeds in three phases. If there are any loose slaves, they are removed in the first place. If there are none, we try to find such $0 \leq i \leq d$ and $0 \leq j < 2^{d-1}$ that node $(i, j+2^{d-1})$ still exists and is independent from node (i, j). We remove the node $(i, j+2^{d-1})$ and attach all of its slaves to the node (i, j). It will now perform the roles of both these nodes. We were allowed to attach all the slaves from one node to the other due to the fact that there were no loose slaves at any of those nodes, so they both had at most 3 slaves each.

If we cannot find either loose slaves or independent masters numbered $(i, j + 2^{d-1})$, we remove double connections, i.e. if we are able to find a pair of masters, that have two bridges between them, we remove one of the bridges. Last of all we can remove nodes $(d-1, j)$ for $0 \le j < 2^d - 1$ one by one, finally decreasing the dimension from d to $d-1$. When we remove such a node, we use one of the slaves that connected it to $(d-2, j)$ and $(0, j)$ to connect $(d-2, j)$ and $(0, j)$ and the other slave can become a loose one of $(0, j)$.

At the same time as removing the double edges, i.e. after all the masters have been merged in pairs, we decrease the dimension d by 1.

4 Comparison of the Maintenance Scheme with the Best Possible Strategy

If a node enters or leaves the network, the topology of the network changes. Each change of the topology causes costs in terms of interruptig the current communication traffic. To compare our strategy with the best possible strategy, we introduce the following, simple cost model:

Definition 3. *Each insertion or removal of a connection costs one cost unit.*

In the following theorem, we assume that the best possible strategy has only to change one connection for each insertion or removal of a node.

It is possible to show that even in this cost model the additional costs induced by our smooth strategy compared to the best possible strategy can be bounded by a constant factor.

Theorem 3. *The smooth maintenance scheme for the CCC scatternet construction is 6-competitive for the insertion and 20-competitive for the removal of nodes compared with a best possible strategy.*

The proof is available in the full version of the paper.

References

[1] Bluetooth Special Interest Group: Bluetooth SIG, Specification of the Bluetooth System, Ver. 1.2. (2003)
[2] Leighton, F.T.: Introduction to parallel algorithms and architectures: array, trees, hypercubes. Morgan Kaufmann Publishers (1992)
[3] Stojmenovic, I.: Dominating set based bluetooth scatternet formation with localized maintenance. In: Proc. of the 16th IEEE International Parallel and Distributed Processing Symposium (IPDPS). (2002)

[4] Zaruba, G., Basagni, S., Chlamtac, I.: Bluetrees – scatternet formation to enable bluetooth based ad hoc networks. In: Proc. of the IEEE International Conference on Communications (ICC). (2001)
[5] Petrioli, C., Basagni, S., Chlamtac, I.: Bluemesh: degree-constrained multi-hop scatternet formation for bluetooth networks. Volume 9. (2004) 33–47
[6] Petrioli, C., Basagni, S., Chlamtac, M.: Configuring bluestars: multihop scatternet formation for bluetooth networks. IEEE Transactions on Computers, Special issue on Wireless Internet (2003) 779–790
[7] Salonidis, T., Bhagwat, P., Tassiulas, L., LaMaire, R.: Distributed topology construction of bluetooth personal area networks. In: Proc. of the 20th IEEE Infocom. Volume 3. (2001) 1577–1586
[8] Ramachandran, L., Kapoor, M., Sarkar, A., Aggarwal, A.: Clustering algorithms for wireless ad hoc networks. In: Proc. of the 4th international workshop on Discrete algorithms and methods for mobile computing and communications. (2000) 54–63
[9] Baatz, S., Bieschke, C., Frank, M., Khl, C., Martini, P., Scholz, C.: Building efficient bluetooth scatternet topologies from 1-factors. In: Proc. of the IASTED International Conference on Wireless and Optical Communications (WOC). (2002) 300–305
[10] Tan, G., Miu, A., Guttag, J., Balakrishnan, H.: An efficient scatternet formation algorithm for dynamic environments. In: Proc. of the IASTED Communications and Computer Networks (CNN). (2002) 68–74
[11] Wang, Z., Thomas, R., Haas, Z.: Bluenet – a new scatternet formation scheme. In: Proc. of the 35th Annual Hawaii International Conference on System Sciences (HICSS). Volume 2. (2002) 61–69
[12] Lin, T.Y., Tseng, Y.C., Chang, K.M.: A new bluering scatternet topology for bluetooth with its formation, routing, and maintenance protocols. Wireless Communications and Mobile Computing **3** (2003) 517–537
[13] Chang, C.T., Chang, C.Y., Sheu, J.P.: Constructing a hypercube parallel computing and communication environment over bluetooth radio system. In: Proc. of the IEEE International Conference on Parallel Processing. (2003) 447–454
[14] Barriere, L., Fraigniaud, P., Narayanan, L., Opatrny, J.: Dynamic construction of bluetooth scatternets of fixed degree and low diameter. In: Proc. of the fourteenth ACM-SIAM Symp. on Discrete Algorithms (SODA). (2003) 781–790

Design of UWB Transmitter and a New Multiple-Access Method for Home Network Environment in UWB Systems

Byung-Lok Cho[1], Young-Kyu Ahn[1], Seok-Hoon Hong[1], Mike Myung-Ok Lee[2], Hui-Myung Oh[3], Kwan-Ho Kim[3], and Sarm-Goo Cho[4]

[1] Dept. of Electronics Engineering, Sunchon National University, Sunchon, Korea
blcho@sunchon.ac.kr, sage@web.sunchon.ac.kr, seokhoon@comsys.sunchon.ac.kr

[2] Dept. of Information and Communication Engineering, Dongshin University, Naju, Korea
mikelee@dsu.ac.kr

[3] Korea Electrotechnology Research Institute

[4] Korea Electronics Technology Institute
chosg@keti.re.kr

Abstract. This paper suggest a new multiple-access method for UWB system. The EVOMA (EigenVector-Orthogonal Multiple Access) made up for the whole weakness of implementation of MB-OFDM and frequency sharing of single-band and, which were relative. Moreover, there is no problem using devices at the same time without interferences in spite of using the single-band because of the frequency characteristics of the pulse itself, and orthogonal characteristics and this has a strong point to implement easily without further need.

1 Introduction

Recently, the UWB system which make it possible to transmit and receive information than 100Mbps in short distance has caught attention. In 2002, FCC in the United States of America posted the standards for use of UWB and suggested imaging systems, indoor and handheld UWB systems, car radar systems and so on as a main application field of UWB. Also, the importance of UWB has been realized in Korea, appointed to power of development in the field of home network as main technology to consist of environment of ubiquitous.

Therefore, this paper will introduce a new multiple access method of UWB which will be used in home network and show the result of implementation of transmitter through the method.

In 2002, FCC in USA defined signal of UWB as weak signal which has bandwidth of 10dB more than 500MHz in the frequency domain and standardization of UWB for local wideband communication has been discussed at IEEE802.15.3a. At present, the system of UWB has been suggested single-band and multi-band like MB-OFDM. However, MB-OFDM has been pretended because of the problem in under circumstance of interferences of devices with limited frequencies in the field of home network which will be used UWB.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 421 – 429, 2005.

This paper will explain the weakness of interferences of single-band and a new multiple access method to make up for the weakness of implementation of MB-OFDM.

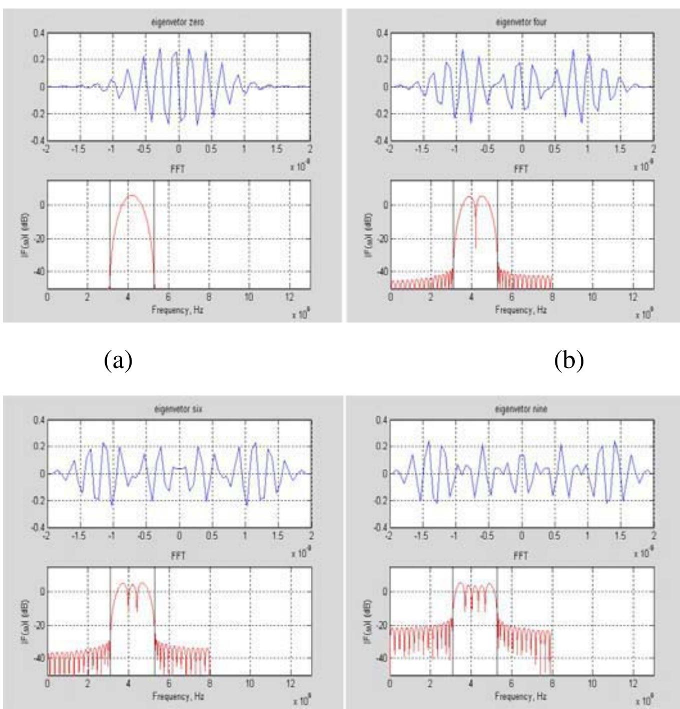
In addition to , the pulse of UWB which will be used in a new multiple access and the multiple access method suggested above will be introduced in the Chapter2. In chapter3, the result of the matlab of transmitter and receiver which acquired from the method suggested will be explain. In Chapter4, the implementation of FPGA and ASIC will be represent and the conclusion will be made in chapter5.

2 UWB Pulse and EVOMA

2.1 Introduction of UWB Pulse Used

This research used the pulses which are satisfied the FCC frequency spectrum mask and orthogonal characteristics of the pulses for a new multiple access method.

The pulses which shifted the frequency band are satisfied with the FCC Frequency mask. however, there are weak points that this makes the structures of the transmitter and receivers complicated and the volumes enlarge. Because of the reason that it sues carrier frequency even though it has frequency spectrum characteristics satisfied with the FCC frequency spectrum mask.



(a) (b)

(c) (d)

Fig. 1. Used UWB Pulse. (a) 1st pulse, (b) 5th pulse, (c) 7th pulse, (d) 10th pulse

Therefore, the pulses that we got from pulse design algorithm by prolate spheroidal wave function[3][4] were used to make up for the weakness.[1][2] The four of 10 UWB pulses which was ejected from the pulse design algorithm are figured on the Fig 1.

The pulses on the fig 1. have a frequency band from 3.1GHz to 5.3GHz and the pulse width is 4ns. Also, these have 64 coefficients. It is possible to eject the pulses which have the frequency band and pulse width suitable for environment to try to applying to when the above mentioned pulse design algorithm is used.

2.2 EVOMA (EigenVector Orthogonal Multiple Access)

The pulses introduced in the former paragraph are made by ejecting the eigenvectors of toeplitz matrix with structure of hermitian and each of them has orthogonal characteristics. A new multiple access method will be suggested, which makes it possible to transmit and receive without interference by using orthogonal characteristics of the pulses.

First of all, block diagram of transmitter and receiver of EVOMA to suggest figured in Fig 2. and Fig 3.

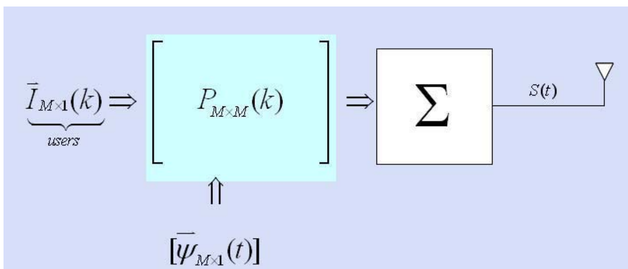


Fig. 2. Block diagram of EVOMA transmitter

In Fig 2.

$$S(t) = \vec{\psi}^{\,T}_{M\times1}(t) \bullet P_{M\times M}(k) \bullet \vec{I}(k) \tag{1}$$

$P_{M\times M}$(k) appeared in formal 1. is permutation matrix and $\Psi_{M\times1}$(k) are pulses produced. Positive pulse transmits when user's input data is ‘1’ and negative pulse transmits when ‘-1’. No any other pulse transmits when ‘0’.

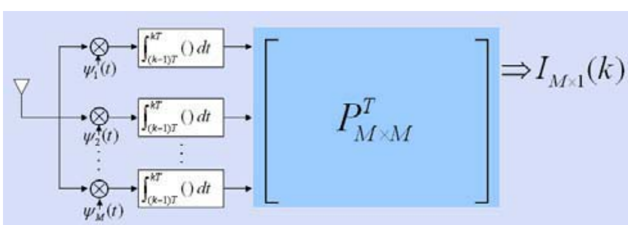


Fig. 3. Block diagram of EVOMA receiver

The pulses produced from pulse generation are assigned to user's input data signal in permutation matrix appeared in fig 2 and 3. At the same time, the pulses are modulated to the sort of BPSK. It is shown in fig 4.

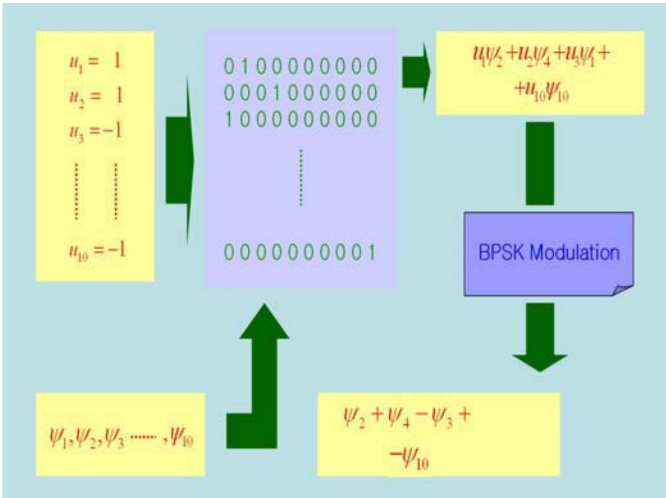


Fig. 4. Block diagram of EVOMA-BPSK

It is made multiple access and the modulation of sort of BPSK when user input data are assigned to the pulses. In the environment of home network, the Modulation is made only with EVOMA without any further process of modulation and it is possible to recover exactly user's input data signal only with correlation process during the receiving process.

3 Simulation

3.1 Simulation Using Matlab 6.5

Used permutation matrix in this research is the formal (2).

$$P_{M\times M}(k) = \begin{bmatrix} 0&1&0&0&0&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0&0&0 \\ 1&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&1&0&0 \\ 0&0&1&0&0&0&0&0&0&0 \\ 0&0&0&0&1&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&1&0 \\ 0&0&0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&1 \end{bmatrix} \tag{2}$$

In the system of fig2, Supposing the transmitting signal is consist '0, 0, 1, 0, -1, 0, 1, 0, 0, 0' when user number is 10, and supposing 3rd user transmit '1', 6th user transmit '-1' and 8th user transmit '1'.

These data transmit as a form of the next fig 5 through the progress of fig 4 in the system of fig 2.

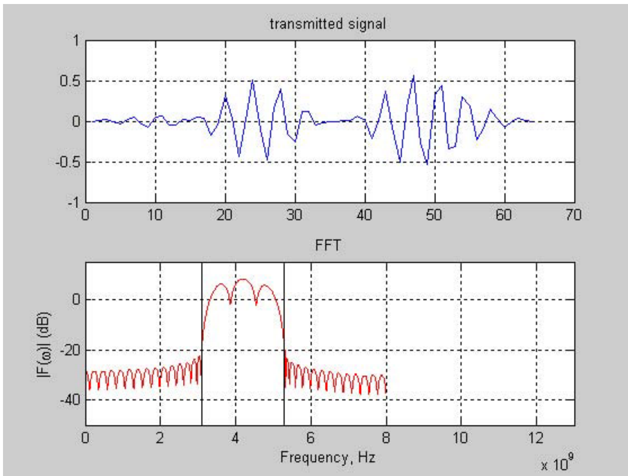


Fig. 5. Transmitted signal S(t)

In the receiver, receives the S(t) and pass through the correlation progress which is applicable to each pulse. The result after the process of correlation is shown in fig 6, and after this progress, the first input signal $I_{M \times 1(k)}$ which is recovered after through the permutation matrix again is shown fig 7.

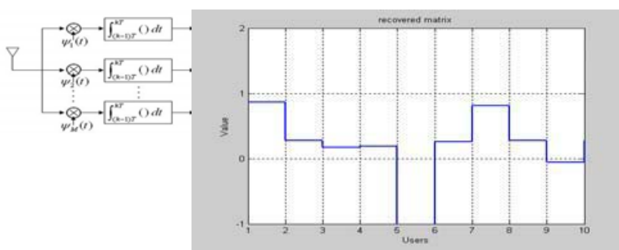


Fig. 6. The result after correlation

The recovered data of each user '0, 0, 1, 0, 0, -1, 0, 1, 0, 0' can be confirmed in fig 7. and Fig 8. These are agree with the input user information at first time, and it is not a result through the final decision process. SNR is simulated in 15dB in AWGN channel.

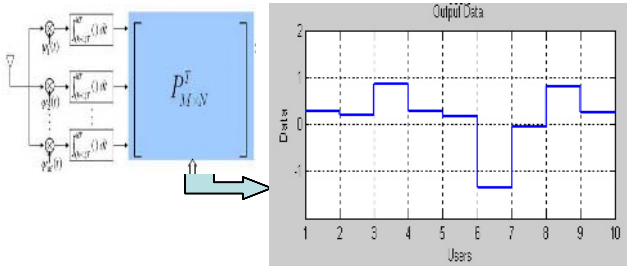


Fig. 7. Recovered each user input data after through the permutation matrix

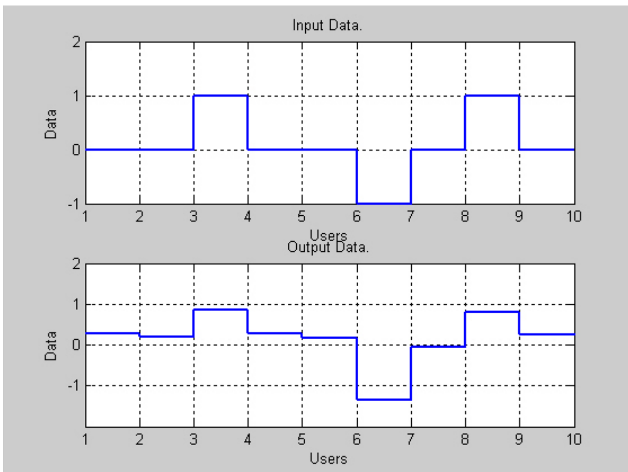


Fig. 8. User input data and recovered data

4 Implementation

4.1 FPGA

To use the target device "Altera Device(Stratix EP1S25F672C6)" and to apply EVOMA, Transmitter is implemented. UWB pulse that will be used, is sampled 64 coefficients, then it is implemented on the Altera Device and designed a permutation, input/output section, pulse generation section and clock divide section using VHDL.

Fig 9. Fig 10. is the first form each of which is from Modelsim and FPGA device(Stratix EP1S25F672C6). When 2nd user transmit the data ‘1’ and 3rd user transmit the data ‘-1’, the expected wave form of S(t) is shown at fig 11. And it is the output from FPGA device.

The simply example of the generated pulse and transmission signal s(t) is shown such above.

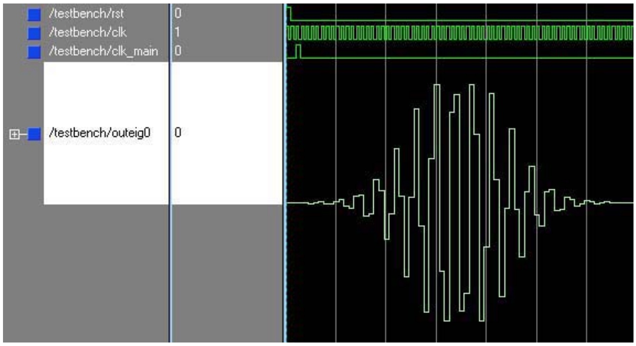


Fig. 9. Block diagram of EVOMA transmitter

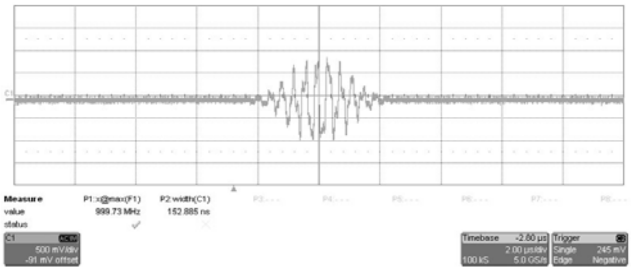


Fig. 10. The first pulse form is confirmed by oscilloscope from the output of FPGA Device

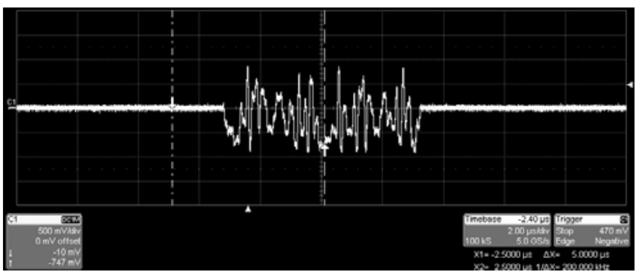


Fig. 11. The transmission signal S(t) is confirmed by oscilloscope form the output of FPGA Device

4.2 ASIC

It confirm the result of the system which was implemented with FPGA, and then implemented to ASIC. The system which was designed with VHDL simulated in the level of the function with ModelSim. The system was synthesized by Hynix0.35um 1-poly 4-metal Phantom Cell Library in the DesignCompiler from SYNOPSYS after that there was no problem in the former simulation. The system was finished through the final layout in the Hynix design house after implementation of Place&Route by

Apollo. the chip is in the process of production and the final test will be scheduled after finishment of the production.

The layout plot of the chip in the process of the production is figured in Fig.12

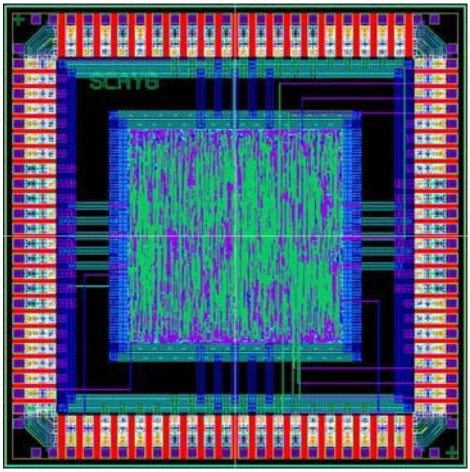


Fig. 12. Chip layout plot. (Hynix0.35um 1-poly 4-metal)

5 Conclusion

In this research, we propose the new multiple access method of UWB for the Home Network. And we have been simulate using matlab and implement using FPGA. Though this method is a single band method, we can show the possibility of transmission and receiving plural information at the same time without interference using the proposal EVOMA method, and it has a more simple structure than MB-OFDM which is a present leading method. Judging from this, it is possible that UWB module will be able to be implemented the miniaturization, high security, low power consumption and low-cost. And it has many possibility at the point of frequency sharing view, for it is able to transmit and receive plural information without interference in the single band.

Here and now, we are evaluating the performance of AWGN in multiplex channel model, and have a plan which is ASIC implementation of receiver and transceiver using EVOMA.

Acknowledgement

This research is partially supported by research grants in Engineering Research Foundation, Sunchon National University in 2004, IDEC(Intergrated circuit Design Education Center), KETI(Korea Electronics Technology Institute), KERI(Korea Electrotechnology Research Institute) and partially supported by the Dongshin University research grants in 2004.

References

1. Brent Parr, ByungLok Cho, Kenneth Wallace, and Zhi Ding, "Spectrally compliant ultra-wideband pulse design with performance analysis", IEEE Tansaction on Wireless Communications, accepted (not published yet). 2004.
2. Brent Parr, ByungLok Cho, Kenneth Wallace, and Zhi Ding, "A Novel UWB Pulse Design Algorithm", IEEE Commun. Lett., 2003.
3. D. Slepian and H.O. Pollak, "Prolate Spheroidal Wave Functions", *Fourier Analysis, and Uncertainty-I, B.S.T.J.*, 40, No. 1, pp. 43-46, 1961.
4. D. Slepian , "Prolate Spheroidal Wave Functions", *Fourier Analysis, and Uncertainty-V:The Discrete Case, B.S.T.J.*, 57, No. 5, pp. 1371-1430, 1978.

Bluetooth Device Manager Connecting a Large Number of Resource-Constraint Devices in a Service-Oriented Bluetooth Network

Hendrik Bohn, Andreas Bobek, and Frank Golatowski

University of Rostock,
Institute for Applied Microelectronics and Computer Science,
Richard-Wagner-Strasse 31,
18119 Rostock-Warnemünde, Germany
{hendrik.bohn, andreas.bobek, frank.golatowski}@etechnik.uni-rostock.de

Abstract. The unique advantages of Bluetooth such as low power consumption capability, cheap hardware interfaces and easy set-up offer new application areas. This is a reason why Bluetooth is even considered for a Service Oriented Architectures (SOAs) consisting of a large number of resource-constraint devices. Although several proposals are available for reducing power consumption for intra-piconet communication, none of them addresses the utilization of the Park mode to reduce power consumption together with the support of a large number of accessible devices. This ongoing research work bridges that gap by proposing polling based on the probability of service access for a centralized Bluetooth network of resource-constraint devices. Services of connected devices (slaves) are offered to the central device (master) which manages all communications in the network. The slaves remain in Park mode unless their services are accessed by the master. This research work shows the feasibility of our proposal.

1 Introduction

Although Bluetooth [1] was originally designed as a cable replacement technology, its unique advances such as low power consumption, cheap hardware interfaces and easy set-up offer further application areas. Bluetooth can be used for a *Service Oriented Architecture* (SOA) which in our case connects a large number of resource-constraint devices (slaves) to a central device (master). SOAs are network architectures in which devices offer services to each other. *Services* are entities which provide information, perform an action or control resources on the behalf of other entities.

The application of Bluetooth in a service-oriented network of resource-constraint devices leads to two main problems. Firstly, a Bluetooth piconet supports only up to 8 devices in an active connection with a maximum bandwidth of 1 Mbps. Secondly, many services are only accessed once in a while and stay idle for most of the time although their Bluetooth interface are still in an active mode.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 430–437, 2005.

The simplest configuration of a Bluetooth networks is a *piconet*, where up to 7 devices (slaves) are actively connected to a device (master) which manages all connections. Actively connected slaves are polled by the master and may send their data. The polling scheme in a piconet is not specified by the Bluetooth specification. Beside active connections the master can manage up to 255 slaves in *Park mode*. Parked slaves listen to the master in a certain interval and remain inactive on low power in between.

The probability of service access ranges from rarely (e.g. outside temperature service) to continuously (e.g. multimedia services). These different requirements for the connection provide new opportunities for power saving approaches. Devices may remain in low power mode unless their services are accessed. After service access associated devices switch to low power mode again.

This paper describes a device manager on a central device (master) for a service-oriented Bluetooth network. Although every device in a piconet can be the master according to the Bluetooth specification, in our network a central device is chosen to be always the master. Our network connects a large number of devices while guaranteeing service access. Therefore, devices send the maximum access intervals for their services (in the Bluetooth service descriptions) to the master when entering the network. The master collects the initial data (e.g. temperature for temperature service) from the services and stores it in a cache. Afterwards, the device is put into Park mode and only reactivated when exact service data is demanded or the service access interval is elapsed.

The remainder of this paper is organized as follows. Section 2 describes related research works. Needed background information is provided in section 3. Section 4 describes the Device Manager and its operations in detail. A conclusion is given and future research is specified in section 5.

2 Related Works

Low power approaches for wireless networks are addressed by numerous research works. On the hardware level, power consumption can be reduced by adjusting the power level on the wireless transmitter during active connections [2]. On the software level, the basic idea is to estimate when a device will transmit data and to suspend it for the time it is not used [3].

Bluetooth devices stay in five different modes or states, respectively, regarding low power approaches: Standby, Active, Sniff, Hold and Park mode. The *Standby mode* marks the unconnected state. The other modes are managed by the master. Active devices listen to all communication whereas the Sniff, Hold and Park modes are low power modes (suspend modes). Devices in *Sniff mode* frequently listen for a certain time quantum to the communication and are suspended otherwise. Devices in *Hold mode* are suspended for a certain time and automatically reactivated afterwards. Parked devices are not connected to the piconet but synchronize to the master in a certain interval. They have to be explicitly reactivated by the master.

Most of the research on intra-piconet communication focuses on optimizing the polling scheme depending on traffic estimations. They either utilize the Sniff mode or the Hold mode. We are not aware of any research which considers the Bluetooth Park mode for reducing power consumption.

Subsequent *polling schemes* utilize the Sniff mode of Bluetooth: Garg et al. proposed several polling schemes varying sniff interval and serving time in a sniff interval [4]. Chakraborty et al. proposed the Adaptive Probability based Polling Interval (APPI) [5]. APPI was developed for bursty traffic and adapts the serving time in a sniff interval to a probable and frequent burst of traffic. Yaiz et al. developed the polling scheme called Predictive Fair Polling (PFP) [6]. PFP uses a urgency metric for each slave predicting if data is available and keeping track of the fairness. The slave with the highest urgency metric is polled.

Following polling scheme make use of the Hold mode: Adaptive Power Conserving service discipline for Bluetooth (APCB), a polling algorithm proposed by Zhu et al. that utilizes the Hold mode [7]. Like the Adaptive Share Polling (ASP) from Perillo and Heinzelman [8]. APCB observes the traffic and estimates the traffic rates. The Hold interval is adapted accordingly. Adjusting power in APCB is done by altering a value which determines the change of the flow rate. In comparison to that, the range between the necessary amount and the actual amount of polls tunes the power consumption in ASP. The reason is the non-predictability of succeeding traffic while polling less or as necessary.

Another polling scheme which should be mentioned in this context is Deficit Round Robin (DRR), proposed by Shreedhar et al. [9]. It works similar to the simple Round Robin (RR) polling scheme, where all slaves are always polled by the master in a certain order and send all their data when polled. DRR limits the transmission of each node to a certain time quantum. If the transmission time of a node exceeds the corresponding quantum the transmission is stopped and the next node is served. The remaining transmission time is added to quantum for the next round. The BlueHoc software uses DRR as the scheduler [10].

The article of Lee at al. examines the affect of the amount of slaves on the throughput and latency time in a Bluetooth network [11]. That research also considers the Park mode. The Park mode is not used to reduce power consumption rather to extend the number of possible slaves and delivers results on throughput and latency time for all (parked and active) slaves. Slave are put to Park mode and reactivated in a RR manner.

We are not aware of any research work addressing the Park mode in low-power approaches although it can be additionally used for extending the number of connected slaves.

3 Background

3.1 Bluetooth

Bluetooth is radio based communication technology with a transmission range of 10 to 100 metres using the 2.4 GHz Industrial, Scientific and Medical (ISM) band. A spread spectrum is used to avoid interferences and noise of other devices

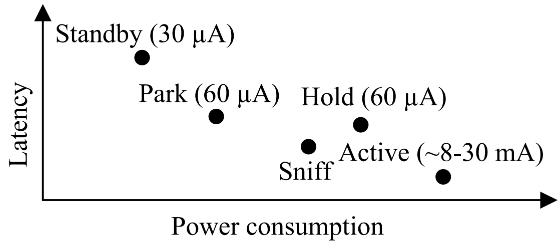


Fig. 1. Latency and power consumption of Bluetooth modes

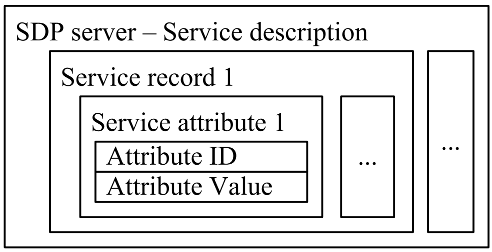


Fig. 2. Overview of the Bluetooth service descriptions

by frequency hopping (1600 hoppings per second). Signals are modulated using a Gaussian Frequency Shift Keying (GFSK) modulation scheme and utilizes slotted Time Division Duplex (TDD) with a slot interval of 625 μsec. The master manages the whole communication in a piconet. Slaves are only allowed to send data when polled by the master.

Figure 1 compares the Bluetooth low-power modes with the Active mode regarding latency and power consumption (adapted from Milios) [12]. The Park mode saves most power of the connection modes but has the longest latency time. The Hold mode uses more power than the Park mode in average because the device switches automatically to Active mode when the Hold interval is elapsed.

3.2 Service Orientation in Bluetooth

Bluetooth offers a minimal service-oriented functionality. The *Bluetooth Service Discovery Protocol* (SDP) offers searching and browsing for Bluetooth services based on service descriptions. Searching for service means that a SDP client (service user) queries available SDP servers (service providers) for desired services. Browsing for services is the searching without prior information about the services. A device can be both, SDP client and server. Only one SDP server per device is allowed. Bluetooth does neither provide any kind of notification mechanisms (e.g. when a SDP server enters or leaves the network, when a service description changes) nor methods to access the services.

A SDP server maintains a list of *service records* (as shown in Figure 2). Each service record represents a service and consists of a list of *service attributes*. Service attributes consist of an *attribute identifier* (ID) and corresponding *attribute value*. Attribute IDs are 16-bit unsigned integers and reflect the semantics of a service. Some attribute IDs and related value data types (e.g. Service Name as a string value) are predefined by the Bluetooth Special Interest Group (SIG). Each service instance belongs to a *service class* that specifies the meaning, prescribed services attributes and data types. New service classes are defined as a subclass of an existing one extended by new attributes. Service classes are represented by a 128-bit Universally Unique ID (UUID). UUID guarantee to be unique across all space and all time.

4 Device Manager for Bluetooth

This paper addresses a single centralized piconet consisting of an always available device (always functioning as Bluetooth master) and several resource-constraint devices (Bluetooth slaves) which may enter and leave the network dynamically. We build on the assumptions that Bluetooth slaves work as SDP servers only. They can not be SDP clients due to there limitations. The Bluetooth master primarily works as a SDP client.

The Bluetooth master contains the *Device Manager* (DM). The DM is permanently available and controls the entire network, all connections and is responsible for reducing the power consumption of the slaves utilizing the Park mode. It involves following tasks: providing scheduling for connected devices, establishing a connection, accessing services, reacting on unnotified device leaving and changes of the service descriptions as well as refusing a device. The task are described in subsequent sub-sections.

4.1 Overview of Device Manager

The Device Manager offers access to available Bluetooth services and hides the actual Bluetooth devices. It accepts service requests and processes them by accessing the Bluetooth services. The DM manages an additional cache which works as a service directory for available services and their states.

We distinguish between three devices regarding Park mode: Cached, on-demand and always-active devices. *Cached devices* update their attribute values in a certain interval. The master requests the values and caches them. *On-demand devices* are woken up by the master when new attributes values are requested. *Always-active devices* deliver accurate values and can not be parked as the name suggests. The type of the device is defined by the semantics of their services. In case that embedded services require different type following rules apply: If a service is always-active the device is always active. Cached and on-demand services are accessed according to their requirements and work parallel.

In case that there are more active devices needed than supported by piconets, the scheduler of Lee et al. [11] can be applied.

4.2 Scheduling

Time controlled operations in conjunction with time constraints belong to classical scheduler problems that are solved by scheduler algorithms normally. Such algorithm is not part of this paper. The DM's scheduler is requested at time of establishing a connection to a new device with required park interval to assure it will not disorganize other devices with required park intervals. Furthermore, the scheduler is responsible for initiating service access to services with park interval attributes.

4.3 Service Description for Connected Devices

The service management of the Device Manager requires two additional attributes which have to be defined by the service provider: MaximumParkInterval and AlwaysActive.

The *MaximumParkInterval* attribute (Unsigned Integer) determines the maximum time a service may be parked (in ms). It is used for cached services to define the maximum interval in which they have to be updated. The Park interval for the device results from the minimal MaximumParkInterval of all services (for devices providing more than one service). The attribute value 0 stands for on-demand services.

The *AlwaysActive* attribute (Boolean) identifies if a device may be parked. If the value is 0 the device may be parked. The value 1 stands for always active devices. Devices are always active devices by default.

4.4 Establishing a Connection

Before an SDP connection can be established the new device has to build up a Bluetooth connection to the master. This may initiated in two ways: The master searches for new devices or the new device runs an inquiry and finds the master.

When the master finds a device it will be automatically connected as an active slave. When a new device starts *paging* (establishing a piconet) it normally functions as a master of the connection (according to the Bluetooth specification). Our described network has a predefined master. Therefore, the Bluetooth role switch procedure [1] has to be applied that the new device becomes a slave.

When the Bluetooth connection is established the master sends an SDP request to the new slave browsing for services. The SDP response of the new slave includes the service descriptions which are put into the service cache of the master. If the slave is an on-demand or cached device it is put into Park mode.

4.5 Accessing a Service

Time of accessing a service by the master depends on service management attributes of the service. Services with a specified value for park interval are woken up by the scheduler accordingly. The new state values are determined by regular Bluetooth SDP operations (request and response) and are stored in the master's cache until they are accessed from outside the master. On-demand services are accessed only if service access is requested from outside the master. Therefore the master asked the scheduler to get an appropriate moment and puts the device into active mode. After using the service the master puts the device back into Park mode. Services requiring always active connections are already in active mode. Service access can happen permanently.

4.6 Disconnection of a Device

Since communication between master and slave is always initiated by the master, there is no way for slaves to inform the master about intended leaving the network. Disconnected devices are recognized at next service access as described above. After realizing such breakdown, the master informs the scheduler and deletes its cache for each service the device was offering.

4.7 Refusing a Device

In certain situations the master has to refuse a device if it tries to establish a new connection. The decision whether to refuse or not is made by the scheduler in dependence of the quality of service constraints given by the device (e.g. asking for large bandwidth). In all cases the master will send a Disconnection Command to the device.

4.8 Changing Service Description While Connected

Changing a service description means adding or removing one or more service attributes in the service record. As said before, communication is initiated by the master only. Therefore changed descriptions are recognized at next service access. For each added attribute the master allocates new fields in the cache, for each missing attribute the master removes according fields, respectively.

5 Conclusion and Future Work

We have described how the Bluetooth Park mode can be utilized to connect a large number of resource-constraint devices while reducing power consumption in a service-oriented Bluetooth network. The whole communication is managed by a predefined master device which contains a Device Manager for the management including a cache for available service descriptions and related service values. The Device Manager can be used to access Bluetooth services from outside the Bluetooth network. This network set-up addresses Bluetooth slaves which only offer services and do not make use of other services. This paper showed the concept of such a network, described the needed algorithms and procedures and illustrated the feasibility of our approach. Currently, we are implementing the concept which will be used for in-car networks.

Future work will be done on evaluations of the implementation and improvements concerning larger bandwidth and shorter latency times. Furthermore, future proposals will adapt this concept to Bluetooth scatternets.

Acknowledgement

This work was supported by the SIRENA project in the framework of the European premier cooperative R&D program "ITEA".

References

1. The Bluetooth Special Interest Group: Specification of the Bluetooth System 1.2 (2004)
2. Monks, J., Bharghavan, V., Hwu, W.-M.: A Power Controlled Multiple Access Protocol for Wireless Packet Networks. In Proceedings of IEEE INFOCOM 2001, Anchorage, Alaska, USA (2001)
3. Kravets, R., Krishnan, P.: Power Management Techniques for Mobile Communication. In Proceedings of MobiCOM, Dallas, Texas, USA 1998 (1998)
4. Garg, S., Kalia, M., Shorey, R.: MAC Scheduling Policies for Power Optimization in Bluetooth: A Master Driven TDD Wireless System. In Proceedings of IEEE Vehicular Technology Conference 2000, Tokyo, Japan (2000)
5. Chakraborty, I., Kashyasp, A., Kumar, A., Rastogi, A., Saran, H., Shorey, R.: MAC Scheduling Policies with Reduced Power Consumption and Bounded Packet Delays for Central Controlled TDD Wireless Networks. In Proceedings of IEEE International Conference on Communications 2001, Helsinki, Finland (2001)
6. Yaiz, R., Heijenk, G.: Polling Best Effort Traffic in Bluetooth. In Proceedings of The Fourth International Symposium on Wireless Personal Multimedia Communications 2001, Aalborg, Denmark (2001)
7. Zhu, H., Cao, G., Kesidis, G., Das, C.: An Adaptive Power-Conserving Service Discipline for Bluetooth. In Proceedings of IEEE International Conference on Communications 2002, New York, USA (2002)
8. Perillo, M., Heinzelman, W.: ASP: An Adaptive Energy-Efficient Polling Algorithm for Bluetooth Piconets. In Proceedings of IEEE Hawaii International Conference on System Sciences 2003, Big Island, Hawaii, USA (2003)
9. Shreedhar, M., Varghese, G.: Efficient Fair Queueing using Deficit Round Robin. In Proceedings of ACM SIGCOMM 1995 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, Cambridge, Massachusetts, USA (1995)
10. BlueHoc: Bluetooth performance evaluation tool. https://oss.software.ibm.com/developerworks/opensource/bluehoc/ (2002)
11. Lee, T.-J., Jang, K., Kang, H., Park, J.: Model and Performance Evaluation of a Piconet for Point-to-Multipoint Communications in Bluetooth. In Proceedings of IEEE Vehicular Technology Conference 2001, Rhodes, Greece (2001)
12. Milios, J.: Baseband Methods for Power Saving. Presentation at Bluetooth Developers Conference, San Jose, California, USA (2000)

ESCORT: Energy-Efficient Sensor Network Communal Routing Topology Using Signal Quality Metrics

Joel W. Branch, Gilbert G. Chen, and Boleslaw K. Szymanski

Rensselaer Polytechnic Institute,
Department of Computer Science, Troy, New York, U.S.A.
{brancj, cheng3, szymansk}@cs.rpi.edu

Abstract. ESCORT aims at decreasing the energy cost of communication in dense sensor networks. We employ radio frequency (RF) signal quality assessment in forming communities of redundant nodes. These communities avoid spanning regions of environmental interference to preserve the routing fidelity of the network. ESCORT is routing protocol-independent and conserves energy by alternating redundant nodes' radio duty cycles. Simulation demonstrates that ESCORT enables nodes to deactivate their radios more than 60% of the time while sustaining acceptable communication performance.

1 Introduction

Advances in hardware and communications technologies, coupled with the increased need for on-demand mobile computing, are fueling the advances in pervasive computing. An essential component of this new computing paradigm is wireless sensor networks (WSNs), which collect and analyze information describing environmental phenomena. Some interesting examples of WSN applications are described in [7], [8].

Unfortunately, WSNs come with inherent challenges. One is that tiny sensor nodes are resource-constrained devices, providing limited storage, processor, and battery capacity. Another one is costly transceiver operation that strains nodes' batteries. Finally, transient wireless links threaten an application's integrity. Therefore, WSN algorithms should promote energy-efficiency while sustaining application quality. The above challenges serve as motivation for our work. Further motivation arises from an observation that each WSN is tuned to a very specific problem, and thus, no individual WSN protocol will be applicable in all scenarios; this includes routing protocols. Therefore, methods for enhancing energy-efficiency of multiple routing protocols must be adopted.

A well-accepted method of energy conservation in WSNs, and one which we follow, is the selective deactivation of nodes' radios. Generally, radio operation uses huge amounts of energy, as represented by the *transmit/receive/sense/idle* ratio for a Crossbow MICA2DOT sensor mote [4]: *75mW/24mW/15mW/81μW* (assuming a 3V power source). As this ratio demonstrates, radio operation is generally the most costly activity of sensor nodes.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 438–448, 2005.

This paper presents ESCORT, which represents our research on the novel use of RF signal quality assessment (SQA) to cluster wireless sensor nodes based on connectivity and spatial separation. This allows a community of redundant nodes to function as a single *virtual* routing entity and thus, operate transparently under most routing protocols. Establishing sleep schedules within the communities then saves energy. Our contribution regards the use of SQA, which helps ESCORT mitigate the effect of packet loss and control the extent of community formation, preserving the connectivity of the overall network.

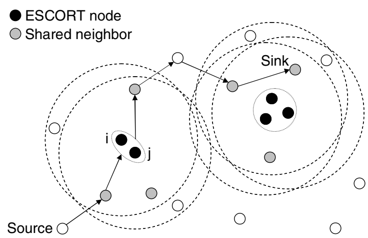


Fig. 1. ESCORT topology applied to a small wireless sensor network

Fig. 1 shows ESCORT's effect on a WSN in which communities of redundant nodes are formed (indicated by the dotted circles), and shared neighbors, residing within the communitys' intersected transmission regions (indicated by dashed circles), are established to ensure that communities maintain only bi-directional links with neighboring nodes. The example path is routed through the community on the left, in which either node *i* or *j* may forward the traffic depending on their transceiver states.

Before continuing, we state some fundamental assumptions concerning ESCORT. First, sensor board components operate independently from transceivers and remain powered over all states. Second, all inter-node links are bi-directional. Third, nodes exhibit no mobility. Additional assumptions are stated as needed.

The remainder of this paper starts with a description of the ESCORT approach in Section 2 and a detailed description of the actual algorithm in Section 3. Section 4 presents a performance evaluation. Section 5 provides a discussion of related work and Section 6 concludes the paper.

2 The ESCORT Approach

2.1 RF Signal Quality Assessment

We define RF signal quality as a combination of two separate metrics: *link quality* and *signal strength*. We describe their uses below. Both factors help to determine the selection of redundant sensor nodes while sustaining acceptable application-level performance.

Link Quality. We use link quality assessment to form communities of nodes with equivalent routing functionality for two main reasons. One, healthy links promote energy-efficient packet delivery. Lal et al. support this assertion by demonstrating that extensive retransmissions over faulty links waste transmission energy [6]. Two, healthy intra-community links provide robust intra-group coordination that preserves the layer of transparency under the selected routing algorithm, ensure proper determination of sleep schedules and help to reliably share routing state information.

Fig. 2 shows the result of considering intra-community link quality in constructing node communities. The obstructions cause poor link quality between two groups of nodes. As a result, separate communities have been formed on either side of the obstructions. Without link quality assessment, one community might have formed, exhibiting poor coordination due to loss of ESCORT control packets. Later, we describe where link quality assessment fits into the ESCORT algorithm.

Designing link quality assessment algorithms is beyond the scope of this paper. We assume use of a technique proposed in [6], in which link quality is graded via packet delivery rate and signal-to-noise ratio measurements. The authors observe that in energy-constrained networks, where nodes save energy via radio deactivation, the quality of the wireless links are not known a priori to packet transmission. Thus, a low-cost initialization phase is used during which nodes periodically wake up to measure link quality.

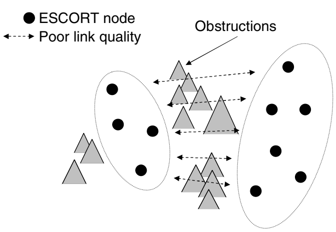


Fig. 2. Effect of link quality assessment in community formation

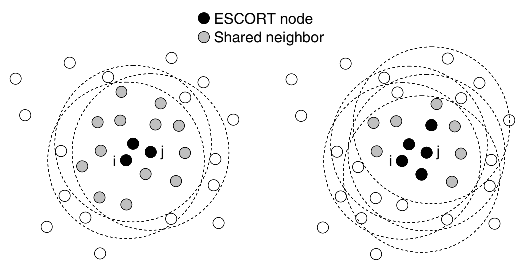


Fig. 3. Comparative effect of using different signal strength thresholds on community formation

Signal Strength. Another important metric for communal topology formation is the spatial separation between nodes. It can be measured in various ways, perhaps the most intuitive being the use of GPS coordinates. However, even though GPS may be used by the application layer, because of its energy requirements and cost, we decided not to require its use. Instead, we use the received signal strength (RSS) for distance estimation and thus the second metric of signal quality.

Other distance measurement methods are available (e.g., time difference of arrival and angle of arrival), but we adopt RSS for its relatively low implementation overhead. We do not convert the actual signal strength into distance, since we need a comparative measure and not absolute distance itself.

Signal strength enables us to control the tradeoff between energy savings and the network connectivity. Adding nodes to community decreases its reach-ability. Fig. 3 illustrates the difference between using smaller and larger RSS thresholds to form communities. In the left community, nodes *i* and *j* cooperate in soliciting nearby nodes to join their community. We assume that all candidates exhibit acceptable link quality with both nodes *i* and *j*. With large RSS thresholds (such that nodes registering RSS measurements above a pre-defined value may join the community) only the closest neighbors gain membership. However, on the right, nodes *i* and *j* relax the RSS threshold, allowing the community to grow larger. As a result, the number of shared neighbors eligible for synchronous communication decreases. Since communal nodes' sleep time increases with community size, the tradeoff between potential energy savings and network connectivity is obvious from the figure.

One might assume that signal strength may also be used to predict packet loss behavior, thus eliminating the need link quality assessment. However, in [11], experiments disprove the perceived strong correlation between signal strength and packet loss, finding that not all links with high RSS exhibit low packet loss. Thus, we do not rely on the use of RSS for measuring link quality.

3 The ESCORT Algorithm

3.1 Initialization

RF Signal Quality Assessment. Initialization begins with each node assessing the quality of its wireless links. As previously stated, we expect to use a method such as that described in [6] to assess link quality. We propose parallelizing RSS assessment with link quality assessment since it would be beneficial to gauge links' signal strengths over multiple samples to obtain average values for each link, $\overline{RSS}$. We also assume that this sub-phase will account for variations in the environment. Neighbors exhibiting intolerable signal quality are excluded as neighbors all together.

Topology Establishment. In this sub-phase, each node identifies an initial partner. Since multiple neighbors will probably exhibit healthy link quality, the neighbor displaying the highest $\overline{RSS}$ value is selected as the initial partner. We also do this in the interest of maintaining network connectivity in the unlikely case that the initial community can not further expand. A `JOIN_REQUEST` packet is sent to the potential partner and pairing is established when two nodes select each other as partners.

The next step involves community expansion. For each initial node pair, *i* and *j*, the node with the highest ID is designated as the *coordinator* (assume this to be *i*) and thus takes responsibility for coordinating the community's expansion. Included with each `JOIN_REQUEST` is the respective node's inner-neighbor set, *A*. Given *i*'s full neighbor set, B_i, and the global link quality threshold, *LQ_THRESH*, A_i is defined as:

$$A_i = \{x : x \in B_i \text{ and } LQ_{ix} > LQ\_THRESH\} \quad (1)$$

where LQ_{ix} is the link quality rating between i and x. Also included in A_i is each neighbor's $\overline{RSS}$ value. Given a predefined global threshold, *RSS_THRESH*, i selects neighbors from both sets, A_i and A_j, to form a potential community set, *PC*, defined as:

$$PC = \{x : x \in A_i, A_j \text{ and } \overline{RSS_{ix}}, \overline{RSS_{jx}} > RSS\_THRESH\} \quad (2)$$

where $\overline{RSS_{ix}}$ and $\overline{RSS_{jx}}$ are the $\overline{RSS}$ values of node x registered at i and j respectively. *PC* represents the set of candidates used for community expansion.

Continuing, the coordinator node solicits each node in the set *PC* to join its community by transmitting `JOIN_REQUEST2` packets. In the event that multiple `JOIN_REQUEST2` packets are received, the $\overline{RSS}$ values of the requesting nodes are used as tie-breakers with the highest $\overline{RSS}$ value winning. This helps optimize the inter-node distance within the formed communities. Solicited nodes then send `JOIN_REPLY2` packets to their chosen coordinator and all nodes belonging to a particular community adopt a pseudo ID matching that of the coordinator node (the original ID is not discarded). This formation of one semantic node allows transparent interaction between the routing and ESCORT protocol layers since all nodes in a community are now receptive to a shared identity.

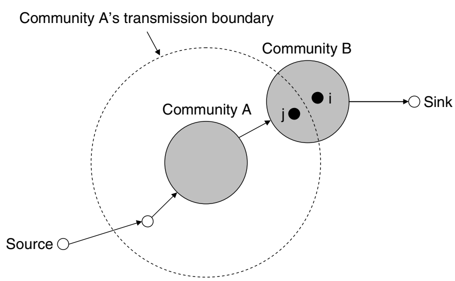


Fig. 4. An ill-conditioned ESCORT community causing an inconsistent routing state

Routing protocol faults may occur if ESCORT produces inconsistent states. This is illustrated by the ill-conditioned community *B* in Fig. 4, two of whose members *i* and *j* lie on either side of community *A*'s communication boundary. Suppose during the routing protocol's path discovery phase, *j*'s radio is active, while the rest of community *B*'s radios are deactivated. Subsequently, *A*, perceived as one node to the network layer, arbitrarily selects *B* as its "next hop" on some constructed path, due to *j*'s active state. However, at a later time, *i*, which lies outside of *A*'s communication range, will become active and the path segment *A*-*B* will cause significant packet loss.

We prevent this scenario by utilizing the community members' individual neighbor sets constructed in the initialization phase. Proceeding community establishment, the intersection of all of the members' neighbor sets is calculated by each member. Afterwards, each node knows the resultant reach-ability of the entire community, and precautions are taken to prohibit communication with excluded neighboring nodes or communities via transmission of `IGNORE` packets. At this point, communities can effectively communicate as singular entities without concerns of asynchrony.

3.2 Runtime

ESCORT's runtime behavior is driven by *leader election* and *state-sharing*. The *leader* node is the one that handles routing for its community during a given duty cycle. Routing protocol state-sharing is conducted between duty cycles to ensure that new leaders route packets appropriately using updated information. To avoid packet loss, ESCORT control messages should use a channel separate from those used by the routing layer for communication. Such separation makes the updated route information delivery rate independent of the application traffic loads that may introduce channel contention. If state-sharing were disrupted due to this condition, the application packet delivery rate would decrease because of possibly incomplete routing information.

Leader Election. Leader election is designed to be efficient, fair, and fault-tolerant, promoting graceful network degradation. To balance the community's workload, the node with the most residual energy at the end of a duty cycle is chosen as the active node for the next cycle. The original coordinator node is the only node known to share high quality links with the rest of its community. Thus, it conducts leader election and state-sharing. This duty, coupled with normal routing duties would place an unfair burden on the coordinator nodes, so they are currently exempt from routing duties.

After the duty cycle time, T_{dc}, expires, all nodes enter the *election* state and send a `VOTE` packet, containing the node's residual energy, to the coordinator. The last active node also includes its energy dissipation rate over the last duty cycle. The coordinator then selects the node with the highest residual energy as the new leader and broadcasts its ID to the community.

Fairness is further achieved by the calculation of T_{dc}. So that all nodes dissipate energy at approximately the same rate, an exponential average is used to predict the dissipation rate in the next duty cycle based on past rates. Thus, if a node j wins the election, its predicted dissipation rate, DR_j, is calculated by the coordinator as:

$$DR_j = \alpha(DR_i) + (1-\alpha)(\tau) \tag{3}$$

where DR_i is the dissipation rate of the last active node during the last cycle, τ stores the average over the history of operation, and α controls the responsiveness to recent history. T_{dc} is then calculated by:

$$T_{dc} = \frac{p \times IE_j}{DR_j} \tag{4}$$

where IE_j is the initial energy of j and p is the percentage of IE_j to be expended in the next duty cycle. T_{dc} is then broadcast to the entire community.

State Sharing. In the state-sharing phase, the last active node sends the new leader its routing layer state information (e.g., forwarding tables, counters, etc.) to maintain routing fidelity. We note that the last active node continues to forward packets on behalf of the community until the end of this phase. Any changes to the state between the time state information is transmitted and received at the new leader should be negligible to performance. If necessary, additional interaction between the routing and ESCORT layers may handle significant route updates, but we leave this for future research. After the routing state is transferred, the new leader remains active while all other nodes sleep for the calculated duty cycle time.

4 Performance Evaluation

In this section, we present an analysis of ESCORT's performance using the SENSE simulator [16]. Our main intent was to compare various performance metrics of a wireless multi-hop sensor network with and without ESCORT applied. The following performance metrics, which we examine, have evolved as standard ratings in the literature for benchmarking WSNs:

- **Packet delivery rate:** Percent of total end-to-end `DATA` packets successfully delivered
- **Packet delay:** End-to-end time incurred for `DATA` packet delivery
- **Sleep rate:** Percent of total time a node spends sleeping
- **Energy consumption:** Energy consumed by a node throughout the simulation
- **Network lifetime:** Time needed for 70% of the path nodes[1] to drain their batteries

4.1 Simulation Framework and Environment

We used four components to model the WSN protocol stack: *application*, *network*, *MAC*, and *physical*. The application component implemented a bursty traffic model. The network component defined the routing protocol. The MAC component provided an implementation of the IEEE 802.11 wireless protocol standard. The physical component simulated the radio. SENSE also has a *channel* component, which simulates propagation effects in the wireless communication medium, and a *battery* component, that models energy consumption. Finally, we designed the *ESCORT* component, placing it between the network and MAC components.

Our energy consumption model is derived from the power specifications for the Crossbow MICA2DOT MPR510CA sensor mote [4]. For most experiments, nodes start with an initial energy of $1*10^4$J. For the experiments in which we measure network lifetime, we change the initial energy to $1*10^3$J in order to decrease lengthy simulation times.

All experiments were executed on a virtual test-bed of size 800m*400m, on which nodes were randomly placed. The population ranged from 30 to 80 nodes; we tested in increments of 10 nodes. We altered the SENSE channel component to model

[1] We only consider nodes lying on route paths because performance of those is the most effected by ESCORT.

obstructions. At this point, obstructions are represented by rectangular entities with a thickness in the range of 5-50m. Obstructions were placed randomly near the center of the test-bed so as to maintain a variety of routes between the sources and sink, which are positioned on opposite sides of the test-bed. For this initial study, we assumed that any significant line-of-site obstructions rendered the signal unsuitable for communication, eliminating its use by ESCORT.

The free space propagation model was used throughout all simulations. All sets of simulations were executed twice: once with nodes using a transmission range of 250m and once using a range of 300m. The size of the DATA packet's payload was 512b. Routes were established using the AODV routing protocol [9]; ESCORT was then applied to AODV for performance testing. For simplicity, traffic (source and sink) nodes were not clustered by ESCORT.

For each simulation type, ten trials were executed for each population. Each simulation ran for 50,000 units of simulated time. In referring to Equations 3 and 4, we set the values of α and p to 0.9 and 0.001 respectively

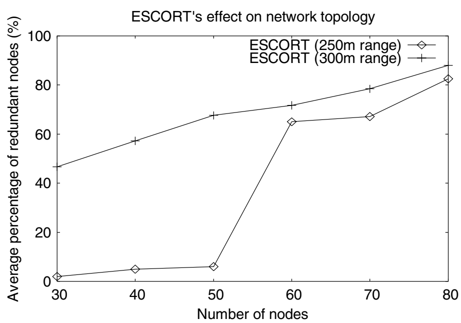


Fig. 5. Transmission range and node density impact on ESCORT's energy savings

4.2 Topology Characteristics

Fig. 5 illustrates how ESCORT's effect on network topology increases along with network density. Fig. 5 specifically shows that for a larger transmission range, ESCORT's effect on the network is more persistent as the network density increases, solidifying the potential for the network to save energy. For a smaller transmission range, ESCORT displays a similar advantage only after a particular threshold, lying somewhere between a population of 50 and 60 nodes. These results lend to the idea that ESCORT is especially beneficial for those networks with larger transmission ranges. We note that the *RSS_THRESH* values were adjusted so as to allow communities to grow larger along with the population size. This is because at smaller populations, larger communities risk having little or no neighbors to route to. Dynamically selecting the *RSS_THRESH* is a subject of future research.

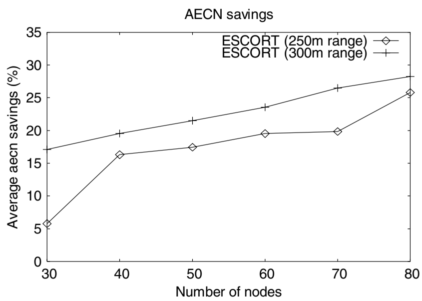


Fig. 6. Average AECN Savings

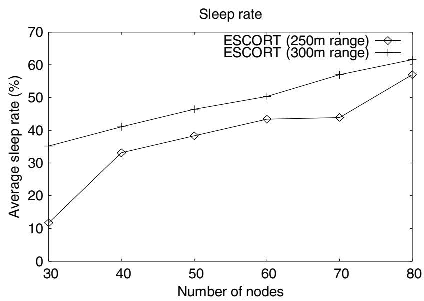


Fig. 7. Average sleep time percentage

4.3 Energy Savings

We assess energy savings by two measurements. First, we inspect the *average energy consumption per node (aecn)*. aecn, adopted from [18], is defined as follows:

$$aecn = \frac{E_0 - E_t}{n \times t} \tag{5}$$

where E_0 and E_t is the initial energy and the residual energy at time t (end of simulation), respectively, for n total nodes. Fig. 6 illustrates the average aecn savings achievable with ESCORT, showing up to 25% reduction for the 250m case and 28% reduction for the 300m case. The second metric we use is sleep rate. Fig. 7 shows that ESCORT allows nodes to sleep up to more than 55% of the time for the 250m case and more than 60% of the time for the 300m case. These two metrics together show that while ESCORT allows a significant amount of sleep time, factors influencing energy savings depend *also* on the nodes' transmission range and other radio power specifications.

ESCORT also contributed a significant improvement to *network lifetime*. For the 250m case, ESCORT achieved up to an approximate 38% (at 80 nodes) increase in network lifetime over the case using just AODV. For the 300m case, similar increase was achieved: 36% with 80 nodes. These results highlight the significance of ESCORT's energy savings that are essential for prolonged WSN operation.

4.4 Packet Delivery Performance

ESCORT's energy-efficiency is achieved without impacting the *packet delivery* rate, which held at about 99% across all experiments. ESCORT's impact on another packet delivery metric, *delay*, shown in Fig. 8 and Fig. 9, is also rather insignificant. Additional delay is no more than approximately 1.3 seconds for the cases of 250m and 300m transmission ranges. This additional delay grows slowly with the increase of node population. Overall, these results show ESCORT's ability to sustain application performance even for large node densities.

Many other attempts at energy savings showed that packet delivery performance usually decreases as a result of increased energy savings. Our results show that

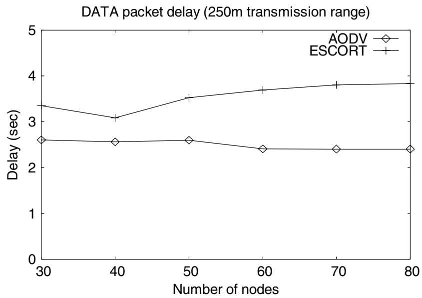


Fig. 8. Average DATA packet delivery delay for the 250m case

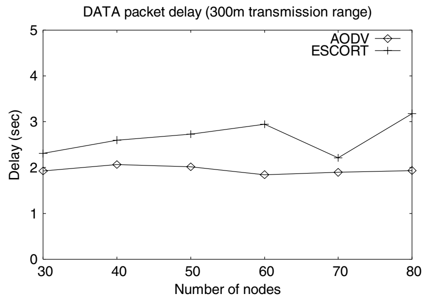


Fig. 9. Average DATA packet delivery delay for the 300m case

ESCORT can decrease the energy expense of communication with minimum tradeoffs in quality of service.

5 Related Work

ESCORT is most closely related to *topology-based* frameworks. The LEACH [5] protocol uses a cluster-based topology in which the cluster-head role is periodically rotated to fairly distribute energy dissipation. Within the clusters, data fusion is used to reduce the traffic load to the base station. Similar to ESCORT, nodes use signal strength measurements to decide which cluster to join. However, the authors focus on direct transmission, rather than multi-hop, WSNs. In GAF [10], nodes divide the network into a grid using GPS coordinates. Grid squares are composed of equivalent nodes which are all able to directly communicate with neighbors in adjacent squares. This work is similar to ESCORT. However, GAF makes no provisions for signal quality in forming communities. ESCORT also forgoes the expense of using GPS technology. Span [2] and ASCENT [1] are similar in nature. Both protocols focus on keeping enough nodes awake to maintain established backbones in the network. In Span, a node wakes up and joins the network only if its adjacent neighbors can not directly communicate with each other. ASCENT makes a similar effort, but causes nodes to join the network only when the quality of the link between its neighbors falls below a predefined threshold. ESCORT is closer related to ASCENT because of its attention given to link quality metrics. However, ESCORT assumes a more pro-active role constructing its neighborhoods, avoiding tradeoffs in packet loss and latency.

6 Conclusion

As we have shown, ESCORT effectively provides energy-efficient routing for wireless sensor networks. Our simulation results give an indication of the significant amount of energy that can be saved with ESCORT. Furthermore, ESCORT is fully distributed and scalable. Hence, our research is important to increasing the feasibility for WSNs.

We have identified several directions for our future research. One is researching ways in which ESCORT may accommodate asynchronous links. Second is to research how to make ESCORT adapt to *dynamic* environmental conditions, such as sporadic node failures. This is very challenging, as no previous work has focused on maintaining well-formed communities (considering signal quality assessment) in dynamic environments. Another aspect of adaptation involves dynamic node *arrivals*. We plan to extend ESCORT to dynamically incorporate new nodes into the routing framework during runtime operation. The third direction is to design mechanisms for waking up sleeping nodes in the case that they are summoned for data retrieval. This might involve the use of a separate low-energy radio cycle. Finally, we plan to research techniques to rotate the functionality of the coordinator node during run-time operation. This presents another significant challenge as the coordinator node must share high-quality wireless links which all of its community members.

References

1. Cerpa, A. and Estrin, D.: ASCENT: adaptive self-configuring sensor networks topologies. Proc. IEEE INFOCOM ’02 (2002) 1278-1287
2. Chen, B., Jamieson, K., Balakrishnan, H., and Morris, R.: Span: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. Proc. ACM MobiCom ’01 (2001) 85-96
3. Chen, G., Branch, J., Pflug, M., Zhu, L., and Szymanski, B.: SENSE: A wireless sensor network simulator. In: Szymanski, B., Yener, B. (eds.): Advances in Pervasive Computing and Networking. Springer, New York (2004) 249-267
4. Crossbow Technology, Inc.: MPR/MIB mote hardware users manual [Online]. Available: http://www.xbow.com
5. Heinzelman, W.R., Chandrakasan, A., and Balakrishnan, H.: Energy-efficient communication protocol for wireless microsensor networks. Proc. HICSS ’00 (2000) 3005-3014
6. Lal, D., Manjeshwar, A., Herrmann, F., Uysal-Biyikoglu, E., and Keshavarzian, A.: Measurement and characterization of link quality metrics in energy constrained wireless sensor networks. Proc. IEEE GLOBECOM ’03 (2003) 446-452
7. Lee, J. and Hashimoto, H.: Controlling mobile robots in distributed intelligent sensor network. IEEE Transactions in Industrial Electronics, Vol. 50, No. 5, (2000) 890-902
8. Mainwaring, A., Polastre, J., Szewczyk, R., Culler, D., and Anderson, J.: Wireless Sensor Networks for Habitat Monitoring. Proc. ACM WSNA’02 (2002) 88-97
9. Perkins, C., Belding-Royer, E., and Das, S.: RFC 3561–ad hoc on-demand distance vector (AODV) routing (2003) [Online]. Available: http://www.faqs.org/rfcs/rfc3561.html
10. Xu, Y., Heidemann, J., and Estrin, D. Geography-informed energy conservation for ad hoc routing. Proc. ACM MobiCom ’01 (2001) 70-84
11. Zhao, J. and Govindan, R.: Understanding packet delivery performance in dense wireless sensor networks. Proc. SenSys ’03 (2003) 1-13

On the Security of Cluster-Based Communication Protocols for Wireless Sensor Networks

Adrian Carlos Ferreira, Marcos Aurélio Vilaça, Leonardo B. Oliveira, Eduardo Habib, Hao Chi Wong, and Antonio A. Loureiro

Federal University of Minas Gerais, MG, Brazil
{adrian, vilaca, leob, habib, hcwong, loureiro}@dcc.ufmg.br

Abstract Wireless sensor networks are ad hoc networks comprised mainly of small sensor nodes with limited resources, and are rapidly emerging as a technology for large-scale, low-cost, automated sensing and monitoring of different environments of interest. Cluster-based communication has been proposed for these networks for various reasons such as scalability and energy efficiency. In this paper, we investigate the problem of adding security to cluster-based communication protocols for homogeneous wireless sensor networks consisting of sensor nodes with severely limited resources, and propose a security solution for LEACH, a protocol where clusters are formed dynamically and periodically. Our solution uses building blocks from SPINS, a suite of highly optimized security building blocks that rely solely on symmetric-key methods; is lightweight and preserves the core of the original LEACH.

1 Introduction

Wireless sensor networks (WSNs) are ad hoc networks comprised of small sensor nodes with limited resources and one or more base stations (BSs), which are much more powerful nodes that connect the sensor nodes to the rest of the world. WSNs are used for monitoring purposes, and can be used in different application areas, ranging from battlefield reconnaissance to environmental protection.

Cluster-based communication protocols (e.g., [1]) have been proposed for ad hoc networks in general and sensor networks in particular for various reasons including scalability and energy efficiency. In cluster-based networks, nodes are organized into clusters, with cluster heads (CHs) relaying messages from ordinary nodes in the cluster to the BSs. This 2-tier network is just an example of a hierarchically organized network that, in general, can have more than two tiers.

Like any wireless ad hoc network, WSNs are vulnerable to attacks [2, 3]. Besides the well-known vulnerabilities due to wireless communication and ad hocness, WSNs face additional problems, including 1) sensor nodes being small, cheap devices that are unlikely to be made tamper-resistant or tamper-proof; and 2) their being left unattended once deployed in unprotected, or even hostile areas (which makes them easily accessible to malicious parties). It is therefore

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 449–458, 2005.

crucial to add security to WSNs, specially those embedded in mission-critical applications.

Adding security to WSNs is specially challenging. Existing solutions for conventional and even other wireless ad hoc networks are not applicable here, given the lack of resources in sensor nodes. Public-key-based methods are one such example. In addition, efficient solutions can be achieved only if tailored to particular network organizations.

In this paper, we investigate the problem of adding security to cluster-based communication protocols for homogeneous WSNs (those in which all nodes in the network, except the BSs, have comparable capabilities). To be concrete, we use LEACH (Low Energy Adaptive Clustering Hierarchy) [1] as our example of protocol. LEACH is interesting for our investigation because it rearranges the network's clustering dynamically and periodically, making it difficult for us to rely on long-lasting node-to-node trust relationships to make the protocol secure.

To the best of our knowledge, this is the first study focused on adding security to cluster-based communication protocols in homogeneous WSNs with resource-constrained sensor nodes. We propose SLEACH, the first version of LEACH with cryptographic protection, using building blocks from SPINS [4]. Our solution is lightweight and preserves both the structure and the capabilities of the original LEACH.

In what follows, we first discuss related work (Section 2), then introduce LEACH and discuss its main security vulnerabilities (Section 3). We then present SLEACH (Section 4), analyze its security and evaluate its performance (Section 5).

2 Related Work

The number of studies specifically targeted to security of resource-constrained WSNs has grown significantly. Due to space constraints, we provide a sample of studies based on cryptographic methods, and focus on those targeted to access control.

Perrig et al. [4] proposed a suite of efficient symmetric key based security building blocks, which we use in our solution. Eschenauer et al. [5] looked at random key predistribution schemes, and originated a large number of follow-on studies which we do not list here. Most of the proposed key distribution schemes, probabilistic or otherwise (e.g., [6]), are not tied to particular network organizations, although they mostly assume flat network, with multi-hop communication; thus they are not well suited to clustered networks. Still others (e.g., [7, 8]) focused on detecting and dealing with injection of bogus data into the network.

Among those specifically targeted to cluster-based sensor networks, Bohge et al. [9] proposed an authentication framework for a concrete 2-tier network organization, in which a middle tier of more powerful nodes between the BS and the ordinary sensors were introduced for the purpose of carrying out authentication functions. In their solution, only the sensor nodes in the lowest tier do not perform public key operations. More recently, Oliveira et al. [10] propose solution that relies exclusively on symmetric key schemes and is suitable for networks with an arbitrary number of levels.

3 LEACH and Its Vulnerabilities

LEACH assumes two types of network nodes: a more powerful BS and resource-scarce sensor nodes. In homogeneous networks with resource-scarce sensor nodes, nodes do not typically communicate directly with the BS for two reasons. One, these nodes typically have transmitters with limited transmission range, and are unable to reach the BS directly. Two, even if the BS is within a node's communication range, direct communication typically demands a much higher energy consumption. Thus, nodes that are farther away usually send their messages to intermediate nodes, which will then forward them to the BS in a multi-hop fashion. The problem with this approach is that, even though peripheral nodes actually save energy, the intermediate nodes, which play the role of routers, end up having a shortened lifetime, when compared with other nodes, since they spend additional energy receiving and transmitting messages.

LEACH assumes every node can directly reach a BS by transmitting with sufficiently high power. However, to save energy and avoid the aforementioned problem, LEACH uses a novel type of routing that randomly rotates routing nodes among all nodes in the network. Briefly, LEACH works in rounds, and in each round, it uses a distributed algorithm to elect CHs and dynamically cluster the remaining nodes around the CHs. To avoid energy drainage of CHs, they do not remain CHs forever; nodes take turns in being CHs, and energy consumption spent on routing is thus distributed among all nodes. Using a set of 100 randomly distributed nodes, and a BS located at 75m from the closest node, simulation results show that LEACH spends up to 8 times less energy than other protocols [1].

Protocol Description. Rounds in LEACH (Fig. 1) have predetermined duration, and have a *setup* phase and a *steady state* phase. Through synchronized clocks nodes know when each round starts and ends.

The setup consists of three steps. In the *advertisement* step, nodes decide probabilistically whether or not to become a CH for the current round (based on its remaining energy and a globally known desired percentage of CHs). Those that will broadcast a message (adv) advertising this fact, at a level that can be heard by everyone in the network. To avoid collision, the CSMA-MAC protocol is used. In the *cluster joining* step, the remaining nodes pick a cluster to join based on the largest received signal strength of a adv message, and communicate their intention to join by sending a join_req (join request) message using CSMA-MAC. Given that the CHs' transmitters and receivers are calibrated, balanced and geographically distributed clusters should result. Once the CHs receive all the join requests, the *confirmation* step starts with the CHs broadcasting a confirmation message that includes a time slot schedule to be used by their cluster members for communication during the steady state phase.

Setup phase

1. $H \Rightarrow \mathcal{G} : h, \mathsf{adv}$
2. $A_i \rightarrow H : a_i, h, \mathsf{join\_req}$
3. $H \Rightarrow \mathcal{G} : h, (\ldots, \langle a_i, T_{a_i} \rangle, \ldots), \mathsf{sched}$

Steady-state phase

4. $A_i \rightarrow H : a_i, d_{a_i}$
5. $H \rightarrow BS : h, \mathcal{F}(\ldots, d_{a_i}, \ldots)$

Fig. 1. LEACH protocol

Setup phase

1.1. $H \Rightarrow \mathcal{G} : h, \mathsf{mac}_{kh}(h \mid ch \mid \mathsf{sec\_adv})$

A_i : store(h)

BS : if $\mathsf{mac}_{kh}(h \mid ch \mid \mathsf{sec\_adv})$ is valid
add(h, V)

1.2. $BS \Rightarrow \mathcal{G} : V, \mathsf{mac}_{k_j}(V)$

1.3. $BS \Rightarrow \mathcal{G} : k_j$

A_i : if $(f(k_j) = k_{j+1})$ and $(h \in V))$
h is authentic

2. $A_i \rightarrow H : a_i, h, \mathsf{join\_req}$

3. $H \Rightarrow \mathcal{G} : h, (\ldots, \langle a_i, T_{a_i} \rangle, \ldots), \mathsf{sched}$

Steady-state phase

4. $A_i \rightarrow H : a_i, d_{a_i}, \mathsf{mac}_{ka_i}(a_i \mid ca_i)$

5.1. $H \rightarrow BS : h, \mathcal{F}(\ldots, d_{a_i}, \ldots), \mathsf{mac}_{kh}(h \mid ch \mid \mathcal{F}(\ldots, d_{a_i}, \ldots))$

5.2. $H \rightarrow BS : h, \mathsf{mac\_array}, (\ldots, a_i, \mathsf{mac}_{ka_1}(a_i \mid ca_i), \cdots), \mathsf{mac}_{kh}(h \mid ch))$

6. $BS \rightarrow H$: intruder ids

Fig. 2. SLEACH protocol

The various symbols denote:

H, A_i : A CH and an ordinary node, respectively
$\mathcal{G}$: The set of all nodes in the network
$\Rightarrow, \rightarrow$: Broadcast and unicast transmissions, respectively
a, h : Node ids
adv, join_req, sched, mac_array : String identifiers for message types
$\langle x, T_x \rangle$: A node id x and its time slot T_x in its cluster's TDMA schedule
d_x : Sensing report from node x
$\mathcal{F}$: Data fusion function
V : An array of node ids
kx : Symmetric key shared by X and BS
cx : Counter shared by node X and BS
$\mathsf{mac}_{kx}()$: MAC calculated using kx
$f()$: One-way hash function
add(x, V) : Add id x to V
store(x) : Store id x for future validation

Once the the clusters are set up, the network moves on to the steady state phase, where actual communication between sensor nodes and the BS takes place. Each node knows when it is its turn to transmit, according to the time slot schedule. The CHs collect messages from all their cluster members, aggregate these data, and send the result to the BS. The steady state phase lasts much longer compared to the setup phase.

Security Vulnerabilities. Like most of the protocols for WSNs, LEACH is vulnerable to a number of security attacks including jamming, spoofing, replay. But because it is a cluster-based protocol, relying on their CHs for routing, attacks involving CHs are the most damaging. If an intruder manages to become a CH, it can stage attacks such as sinkhole and selective forwarding, thus disrupting the network. The intruder may also leave the routing alone, and try to inject bogus sensor data into the network, one way or another. A third type of attack is passive eavesdropping.

4 Adding Security to LEACH

Attacks to WSNs may come from *outsiders* or *insiders*. In cryptographically protected networks, outsiders do not have credentials (e.g., keys or certificates) to show that they are members of the network, whereas insiders do. Insiders may not always be trustworthy, as they may have been compromised, or have stolen their credentials from some legitimate node in the network. The solution we propose here is meant to protect the network from attacks by outsiders only. Another rather ordinary trust assumption we make is that BSs are trusted.

In this section, we add two of the most critical security properties to LEACH: **data authentication** (it should be possible for a recipient of a message to authenticate its originator), and **data freshness** (it should be possible for a recipient of a message to be sure that the message is not a replay of an old message). We focus on devising a solution to prevent an intruder from becoming a CH or injecting bogus sensor data into the network by pretending to be one of its members. Our solution uses building blocks from SPINS [4], a suite of lightweight security primitives for resource-constrained WSNs.

4.1 SPINS Overview

SPINS [4] consists of two symmetric-key security building blocks optimized for highly constrained sensor networks: SNEP and μTESLA. SNEP provides confidentiality, authentication, and freshness between nodes and the BS, and μTESLA provides authenticated broadcast. μTESLA implements the asymmetry required for authenticated broadcast using one-way key chains constructed with cryptographically secure hash functions, and delayed key disclosure. μTESLA requires loose time synchronization. See [4] for further details on SPINS.

4.2 Overview of Our Solution

SLEACH needs an authenticated broadcast mechanism to allow non-CHs to authenticate the broadcaster as being a particular, legitimate, node of the network. Our small nodes do not have, however, the resource level needed to run μTESLA (it requires the sender to store a long chain of symmetric keys).

We propose a solution that divides this authenticated broadcast into two smaller steps, leveraging on the BS, which is trusted and has more resources. In a nutshell, assuming that each sensor node shares a secret symmetric key with the BS, each CH can send a slightly modified adv message, including the id of the CH in plaintext (which will be used by the ordinary nodes as usual) and a message authentication code (MAC[1]) generated using the key the CH shares with the BS (the MAC will be used by the BS for the purpose of authentication). Once all these (modified) adv messages have been sent by the CHs, the BS will compile the list of legitimate CHs, and send this list to the network using the μTESLA broadcast authentication scheme. Ordinary nodes now know which of the (modified) advs they received are from legitimate nodes, and can proceed with the rest of the original protocol, choosing the CH from the list broadcast by the BS.

We can modify the rest of the setup protocol similarly. However, this would require that BS to authenticate each and all nodes of the network at the beginning of each round, which is not only prohibitively expensive, but also makes BS a bottleneck of the system. Thus, we leave these messages unauthenticated, and argue, in Section 5.1, why this decision does not bring devastating consequences, as long as we add an lighter-weight corrective measure.

4.3 Protocol Details

Predeployment. Each node X is preloaded with two keys: χ_X, a master symmetric key that X shares with the BS; and k_n, a group key that is shared by all members of the network. For freshness purposes, each node X also shares a counter C_X with the BS.

From χ_X, the key holders derive K_X, for MAC computation and verification. k_n is the last key of a sequence S generated by applying successively a one-way hash function f to an initial key k_0 ($S = k_0, k_1, k_2, \ldots, k_{n-1}, k_n$, where $f(k_i) = k_{i+1}$). The BS keeps S secret, but shares the last element k_n with the rest of the network.

Setup Phase: Advertisement. Once it decides to be a CH, a node H broadcasts a sec_adv message (step 1.1, Fig. 2), which is a concatenation of its own id with a MAC value produced using K_X. Ordinary nodes collect all these broadcasts, and record the signal strength of each. The BS receives each of these broadcasts, and verifies their authenticity.

[1] Note that MAC is often used to stand for medium access control in networking papers. In this paper, we use MAC to stand for message authentication code.

Once the BS has processed all the sec_adv messages, it compiles the list V of authenticated H's, identifies the last key k_j in S that has not been disclosed (note that all key k_i, such that $i > j$, have been disclosed, whereas all key k_i, such that $i \leq j$, have not), and broadcasts V (step 1.2) using μTESLA and k_j. k_j is disclosed after a certain time period (step 1.3), after all nodes in the network have received the previous message.

Cluster Joining. After receiving both the broadcast and the corresponding key, ordinary nodes in the network can authenticate the broadcast from the BS and learn the list of legitimate CHs for the current round. (Note that the key is authentic only if it is a an element of the key chain generated by the BS, and immediately precedes the one that was released last. That is, if $f(k_j) = k_{j+1}$.) They then choose a CH from this list using the original algorithm (based on signal strengths), and send the join_req message (step 2) to the CH they choose. Note that this message is unprotected, and identical to message 2, Fig. 1.

Confirmation. After the CHs receive all the join_reqs, they broadcast the time slot schedule to their cluster members (step 3). Depending on the security level required, we can take advantage of the same procedure used to authenticate sec_adv messages here.

Steady-State Phase. During this phase, sensor nodes send measurements to their CHs (step 4). To authenticate the origin of these measurements, they also enclose a MAC value produced, using the key they share with the BS. The CHs aggregate the measurements, and transmit the aggregate result, authenticated, to the BS; the MACs from the cluster members, are simply forwarded in mac_array messages, as they are unable to verify them. Note that the number of mac_array messages is dependent on the size of the cluster (step 5.2).

The BS verifies both the MAC value generated by the CH, as well as the ones from the ordinary nodes. Unless all verifications are successful, the BS will discard the corresponding aggregate result, and the originators of failed MACs will be seen as intruders. In case there are intruders among the ordinary nodes, the BS will report their identities to their CHs, which will then drop message from these nodes for the remaining of the round.

Note that the MAC values from the ordinary nodes do not take the measurements into account. In fact, they should not be, as the BS would not be able to verify them (note that the BS do not learn the measurements themselves, but only their aggregate value). Thus, the MAC values, in this case, only authenticate the fact that the key holder has sent one more message (as the counter value has been incremented). The BS needs to trust that the CHs indeed used the reports from their members to generate the aggregate result.

Also note that most of the messages (except the MACs from the non-CHs) travel single hop. This means that they do not go through intermediate nodes where they could potentially be corrupted maliciously. Thus, in this work, we use MAC just for authentication purposes. To handle non-malicious corruptions of messages from the environment, in single hop communications, we use mechanisms such as CRC – Cyclic Redundancy Check.

In this paper, for the purpose of simplicity, we assume that there are no additional control messages, aside from the ones we show. It is not difficult to see, however, that they can be handled the way setup messages are.

5 Security Analysis and Performance Evaluation

5.1 Security Analysis

In designing SLEACH, our goal was to implement access control, and prevent intruders from participating the network. We discuss below how well we achieve it.

Our solution allows authentication of sec_adv messages (steps 1.1, 1.2, and 1.3, Fig. 2), and prevents intruders from becoming CHs. Thus, unless there are insider attacks, the network is protected against selective forwarding, sinkhole, and HELLO flood attacks [2]. Note that we leave the confirmation message (step 3, Fig. 2) unauthenticated; and an intruder would be able to broadcast bogus time slot schedules, possibly causing DoS problems in the communication during the steady-state phase. We argue that the intruder will likely have simpler ways (jamming, e.g.) to accomplish the same objective.

Instead of trying to become CHs, intruders may try to join a cluster, with three goals in mind: (1) To send bogus sensor data to the CH, and introduce noise to the set of all sensor measurements; (2) To have the CH forward bogus messages to the BS, and deplete its energy reserve; and (3) To crowd the time slot schedule of a cluster, causing a DoS (Denial of Service) attack, or simply lowering the throughput of a CH. Our solution does not prevent intruders from joining the clusters (join_req messages, step 2, Fig. 2, are not authenticated), but does prevent them from achieving the first goal. Their rogue measurements (or better, the aggregate report that embed these measurements) will be discarded by the system, as they are unable to generate MACs that can be successfully verified by the BS, during the steady-state phase. Note that this verification also guarantees freshness, as the counter value should have been incremented from last time. We also prevent the intruders from achieving the second goal: their CHs will cease to forward their messages, once they are flagged and reported by the BS (again because of MACs that cannot be verified). Our scheme cannot prevent the intruders from achieving the third goal, but we argue that it can be accomplished by other much easier means, such as jamming the communications channels, for example.

Our solution does not guarantee data confidentiality. To do so, while still preserving the data fusion capability, pairwise keys shared between CHs and their cluster members would be needed.

5.2 Performance Evaluation

Our solution is extremely simple: each node, aside from the BS, is preloaded with only two keys, one for the BS to authenticate the legitimate members of the network, and the other for authenticated broadcasts from the BS.

In terms of communication and processing overhead, the SLEACH setup protocol incurs to the BS and negligible overhead: one authenticated broadcast and one key disclosure. For the CHs, sending a sec_adv message instead of adv incurs one MAC computation and energy for transmitting the MAC bits. For the non-CHs, the additional work has to do with receiving and processing the BS's authenticated broadcast (steps 1.2 and 1.3, Fig. 2), the computation overhead consisting of one MAC and one (a few in cases where desynchronization occurs) application of f. All these overheads are minimum.

For the steady-state phase, SLEACH has all nodes send authenticated messages, which requires one MAC computation and additional MAC bits in the message. In addition, the CHs also forward MACs from their cluster members to the BS. Taking the current values (e.g., cluster size from [11], and MAC size from [4]) into account, we believe that this overhead is tolerable.

6 Conclusion

To the best of our knowledge, this is the first study focused on adding security to cluster-based communication protocols in homogeneous WSNs with resource-constrained sensor nodes. We proposed SLEACH, the first modified version of LEACH with cryptographic protection against outsider attacks. It prevents an intruder from becoming a CH or injecting bogus sensor data into the network.

SLEACH is quite efficient, and preserves the structure of the original LEACH, including its ability to carry out data fusion.

The simplicity of our solution relies on LEACH's assumption that every node can reach a BS by transmitting with sufficiently high power. Thus, we expect our solution to be applicable to any cluster-based communication protocol where this assumption holds. In cases where it does not hold, alternative schemes are needed. This is topic for future work.

References

1. Heinzelman, W.R., Chandrakasan, A., Balakrishnan, H.: Energy-efficient communication protocol for wireless microsensor networks. In: IEEE Hawaii Int. Conf. on System Sciences. (2000) 4–7
2. Karlof, C., Wagner, D.: Secure routing in wireless sensor networks: Attacks and countermeasures. In: Elsevier's AdHoc Networks Journal, Special Issue on Sensor Network Applications and Protocols. Volume 1. (2003) 293–315 Also appeared in First IEEE International Workshop on Sensor Network Protocols and Applications.
3. Wood, A.D., Stankovic, J.A.: Denial of service in sensor networks. IEEE Computer **35** (2002) 54–62
4. Perrig, A., Szewczyk, R., Wen, V., Culler, D., Tygar, J.D.: SPINS: Security protocols for sensor networks. Wireless Networks **8** (2002) 521–534 Also appeared in Mobile Computing and Networking.
5. Eschenauer, L., Gligor, V.D.: A key management scheme for distributed sensor networks. In: Proceedings of the 9th ACM conference on Computer and communications security, ACM Press (2002) 41–47

6. Zhu, S., Setia, S., Jajodia, S.: Leap: efficient security mechanisms for large-scale distributed sensor networks. In: Proceedings of the 10th ACM conference on Computer and communication security, ACM Press (2003) 62–72
7. Przydatek, B., Song, D., Perrig, A.: SIA: Secure information aggregation in sensor networks. In: ACM SenSys 2003. (2003) 175–192
8. Yea, F., Luo, H., Lu, S., Zhang, L.: Statistical en-route filtering of injected false data in sensor networks. In: INFOCOM 2004. (2004)
9. Bohge, M., Trappe, W.: An authentication framework for hierarchical ad hoc sensor networks. In: Proceedings of the 2003 ACM workshop on Wireless security, ACM Press (2003) 79–87
10. Oliveira, L.B., Wong, H.C., Loureiro, A.A.F.: LHA-SP: Secure protocols for hierarchical wireless sensor networks. In: 9th IFIP/IEEE International Symposium on Integrated Network Management (IM'05), Nice, France (2005) To appear.
11. Melo, E.J.D., Liu, M.: The effect of organization on energy consumption in wireless sensor networks. In: IEEE Globecom 2002. (2002)

An Energy-Efficient Coverage Maintenance Scheme for Distributed Sensor Networks

Minsu Kim[1], Taeyoung Byun[2], Jung-Pil Ryu[1], Sungho Hwang[1], and Ki-Jun Han[1,*]

[1] Department of Computer Engineering Kyungpook National University, Korea
{kiunsen, goldmunt, sungho}@netopia.knu.ac.kr
kjhan@bh.knu.ac.kr
[2] Dept. of Computer & Information Communication Engineering, Catholic Univ. of Daegu
tybyun@cu.ac.kr

Abstract. This paper presents Tri-State Channel Access scheme to augment energy efficiency for wireless sensor networks. All sensor nodes have three states; SLEEP, ROUTE, and REPORT. In our scheme, the SLEEP state nodes are selected by simple perimeter coverage with backoff algorithm, and the ROUTE or REPORT state nodes are categorized by the priority based backoff algorithm. The performance of our scheme is investigated via computer simulations and simulation results show that the proposed scheme ensures full area coverage after turning off some redundant nodes. In additions, our scheme can be easily applied to dependable sensor network emphasizing reliability and robustness.

1 Introduction

In recent years, wireless sensor networking technology has seen a rapid development with many applications such as smart environments, disaster management, habitat monitoring, combat field reconnaissance, and security surveillance [7][9][10][13 – 17]. Sensors in these applications are expected to be remotely deployed and to operate autonomously in unattended environments.

A wireless sensor network consists of tiny sensing devices that are deployed in a region of interest. The sink node aggregates and analyzes the report message received and decides whether there is an unusual or salient event occurrence in the deployed area. Considering the limited capabilities and vulnerable nature of an individual sensor, a wireless sensor network has a large number of sensors deployed in high density and thus redundancy can be exploited to increase data accuracy and system dependability. In a wireless sensor network, the energy source provided for sensors is usually battery power, which has not yet reached the stage for sensors to operate for a long time without recharging. Moreover, sensors are often intended to be deployed in remote or hostile environments, such as a battlefield or desert; therefore, it is undesirable or impossible to recharge or replace the battery power of all the sensors. However, long system lifetime is expected by many monitoring applications. The system lifetime, which is measured by the time until all nodes have been drained out of their battery power or the network no longer provides an acceptable event detection ratio, directly affects network

* Correspondent author.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 459–466, 2005.

dependability. Therefore, an energy-efficient design for extending a system's lifetime without sacrificing system dependability is an important challenge to the design of a large wireless sensor network. In wireless sensor networks, all nodes share common sensing tasks. Thus, not all sensors are required to perform the sensing task during the whole system lifetime. Turning off some nodes does not affect the overall system function as long as there are enough working nodes to assure it. Therefore, if we can schedule sensors to work alternatively, the system lifetime can be prolonged correspondingly; i.e. the system lifetime can be prolonged by exploiting redundancy.

A number of studies for reducing the power consumption of sensor networks have been performed in recent years. These studies mainly focused on a data-aggregated routing algorithm [1 – 4], energy efficient MAC protocols [6 – 8], and the application of level transmission control [9][12]. While initially researchers believed that sensor networks will play a complementary role that enhances the quality of these applications, recent research results have encouraged practitioners to envision an increased reliance on sensor networks. To realize their potential, dependable design and operation of sensor networks have to be ensured. Dependability is a property that indicates the ability of a system to deliver specified services to the user. Dependability can be specified in terms of attributes, which include reliability, availability, safety, maintainability, and security.

In this paper, we suggest a Tri-State channel access scheme, which guarantees full-area coverage and reduces redundancy. All sensor nodes have three states; SLEEP, ROUTE, and REPORT. In our scheme, the SLEEP state nodes are selected by the SPCB (Simple Perimeter Coverage with Backoff) algorithm, and the ROUTE or REPORT state nodes are chosen by the PBB (Priority Based Backoff) algorithm. Using the SPCB algorithm, we can obtain the SLEEP state node lists without loss of the area coverage. Namely, the SPCB algorithm categorizes all nodes into active and inactive nodes, and the PBB algorithm divides the active nodes into report nodes and route nodes. The report nodes are classified into several subsets from the active nodes to augment the dependability of the sensing report, and report nodes belonging to the eligible subset generate sensing data and transmit the data to the adjacent router nodes. Eligibility can be allowed to several subsets for dependability at the same time or to only one subset for energy efficiency.

The performance of the proposed scheme was investigated via computer simulations. Simulation results show that our scheme reduces the report of redundant packets and that each subset of report nodes preserves most of the sensing coverage. Our paper is organized as follows. Section 2 reviews related works and section 3 introduces our scheme. In section 4 simulation results are presented. Finally, section 5 presents our conclusions.

2 Proposed Scheme

Due to the tight restrictions of the sensor node, low power consumption is one of the most important requirements. In this paper, we suggest a Tri-State channel access scheme, which consists of the SPCB [14 – 15] and the PBB algorithm. The priority in the PBB algorithm is determined by geographical density. Fig. 1 presents a state transition diagram of the Tri-State. As shown in this figure, the SLEEP state nodes selected by the SPCB algorithm are configured as inactive modes and other nodes must perform their given activity by the PBB algorithm. Namely, the REPORT state nodes make a sensing message and transmit it to the neighboring active nodes, and the ROUTE state nodes only forward the packets to the sink node.

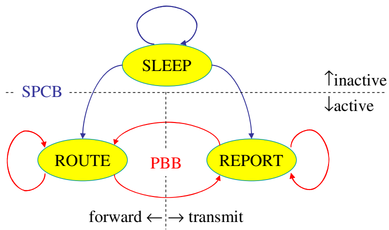
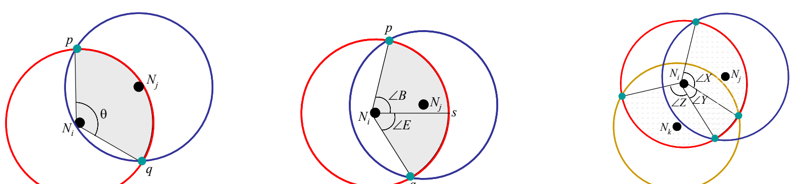


Fig. 1. State Transition Diagram of Tri-states

All active nodes are classified into several subsets. One or more subsets with report eligibility are activated as report nodes and other active nodes belonging to subsets without eligibility, that are routing nodes, perform only deliver the report messages. To augment system dependability, two or more subsets of the active nodes can be granted report eligibility.

2.1 Simple Perimeter Coverage with Backoff Algorithm

As discussed above, the main objective of SPCB is to maintain the original sensing coverage as well as minimize the number of working nodes [14]. Tian et al compute each node's sensing area and then compare it with its neighbors' [14]. If the whole sensing area of a node is fully embraced by the union set of its neighbors', i.e. if neighboring nodes can cover the current node's sensing area, this node can be turned off without reducing the system's overall sensing coverage. They investigated the redundant sensing area of a node by computing the union of central angles of all sponsored sectors. The sponsored sector of a node is included by the overlapped area with its neighbor. The sponsored sector, which is shaded, and its central angle, denoted by (, are illustrated as Fig. 2a. A node can be covered by its neighbors when the union of central angles of sponsored sectors reaches 360. If the whole sensing area of a node is fully embraced by the union set of its neighbors', i.e. if neighboring nodes can cover the current node's sensing area, this node can be turned off without reducing the system's overall sensing coverage.



(a) Sponsored sector (b) standard point for union (c) the union of sponsored sector

Fig. 2. Sponsored sector and its problem

From Fig. 2c, we can see that the central angle of a sponsored sector by node N_j is $\angle X + \angle Y$. Additionally, the central angle of a sponsored sector by node N_k is $\angle Y + \angle Z$. However, the central angle of the two sponsored sectors by N_j and N_k is $\angle X + \angle Y + \angle Z$, not $\angle X + 2\angle Y + \angle Z$. Therefore, the overlapped angle ($\angle Y$ in Fig. 2c) must be removed when computing the union of two central angles.

To compute the union of all central angles, the coordinates of point of the intersections, that is p and q in Fig. 2 must be computed. From (1) to (7), we are going to describe the method of computing the coordinates of p and q. Given coordinates of N_i and N_j, the coordinates of the middle point between N_i and N_j (x_c, y_c) can be obtained by

$$x_c = \frac{x_1 + x_2}{2}, y_c = \frac{y_1 + y_2}{2} \tag{1}$$

where (x_1, y_1) and (x_2, y_2) are coordinates of N_i and N_j, respectively.

Additionally, the slope of the equation of the line, which passes through the p and q, can be calculated by

$$y = -\frac{x_2 - x_1}{y_2 - y_1} x + y_c + \frac{x_2 - x_1}{y_2 - y_1} x_c \tag{2}$$

Because the p and q are the intersection point of this line and the circle of N_i, the equation of the circle can be expressed by

$$(x - x_1)^2 + (ax + c - y_1)^2 = r^2 \tag{3}$$

Using (2) and (3), we can compute two roots of the second order equation. In addition, we also can get the y-coordinate of p and q by substituting the two roots into (3).

Now, we are going to explain the method of union of all central angles. After computing the coordinates of p and q, two angles, $\angle B$ and $\angle E$, can be calculated by

$$\angle B = \angle pN_i s = \arccos\left(\frac{X_1 - x_1}{r}\right) \qquad \angle E = \angle qN_i s = \arccos\left(\frac{X_2 - x_1}{r}\right) \tag{4}$$

where s is the standard point for determining the overlap of two central angles, and its coordinates are $(x_1 + r, y_1)$, as shown in Fig. 2b. In addition, X_1 and X_2 are the two roots of second order equation in (3). These two angles are added as a pair to the central angle lists of N_i. As discussed, node N_i can set itself as an inactive state when the union of all elements of the list include all of 0 ~ 360. Otherwise, the node is responsible for routing or reporting. Note that only active neighbors can be used to compute the union of the central angle of the sponsored sector. Therefore, some determining arbitration is needed for the two adjacent undetermined nodes. We use a priority-based back-off scheme for the arbitration described in this paper.

2.2 A Priority Based Backoff Algorithm

All active nodes are divided into several subsets due to the rule. We need not consider a routing scheme or its reachability, because the connectivity of active nodes of SPCB scheme is proved in [15]. Therefore, all report nodes generate sensing data and deliver it to the sink node with help of route nodes.

To divide all active nodes into several subsets, we adopt a probabilistic approach. Each active node investigates the geographical densities, shares the densities with their active neighbors, and computes the report probabilities. Hence, the number of subsets and the report probability are in inverse proportion to each other.

For simplicity, we define the initial density of active node i, denoted by $n_i.d_1$, as follows:

$$n_i.d_1 = \|n_i.nn\| \qquad \text{for } i = 1, 2, 3, \ldots \tag{5}$$

where $n_i.nn$ is the set of active neighbor IDs of i^{th} active node, and $\|A\|$ means size of the set A.

As mentioned above, we utilize the density information for the calculation the report probability in the PBB. Because the density and the neighbor number are in inverse proportion to each other, we used the inverse of the density. Therefore, the average inverse density of active neighbors can be expressed by

$$E\left[n_i.nd_1^{-1}\right] = \frac{\sum_{m \in n_i.nn} n_m.d_1^{-1}}{n_i.d_1} \qquad \text{for } i = 1, 2, 3, \ldots \tag{6}$$

Using (5) and (6), the initial report probability of active node i, denoted by $n_i.p_1$, can be defined as

$$n_i.p_1 = \min\left(1,\ \alpha n_i.d_1 + (1-\alpha)E\left[n_i.nd_1^{-1}\right]\right) \qquad \text{for } i = 1, 2, 3, \ldots \tag{7}$$

where α is scaling factor.

Let us define the first subset as A_1 produced by the initial report probability at (7). Note that node i can gather a list of neighbors belonging to the subset A_1, that is $n_i.nn \cap (A_1)$. In our scheme each node selects its own subset using the backoff algorithm. The backoff window size, denoted by $n_i.cw$, and the backoff counter, denoted by $n_i.bc$, of node i which belongs to the subset A_1, can be computed as follows:

$$\begin{aligned} &n_i.cw = n_i.d_1 && \text{for } i = 1, 2, 3, \ldots \\ &n_i.bc = 0 && \text{for } i = 1, 2, 3, \ldots \end{aligned} \tag{8}$$

In addition, the backoff window size and backoff counter of node i belonging not to the subset A_1 can be computed as follows:

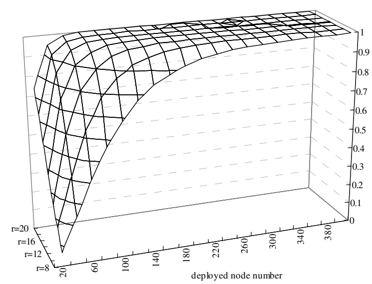
$$n_i.cw = \frac{n_i.d_1}{\|n_i.nn \cap (A_1)\|} \qquad \text{for } i = 1, 2, 3, \ldots \tag{9}$$

Note that these backoff parameters are only used to divide all active nodes into several subsets. Therefore, the backoff window size must be approximated to the number of all subsets.

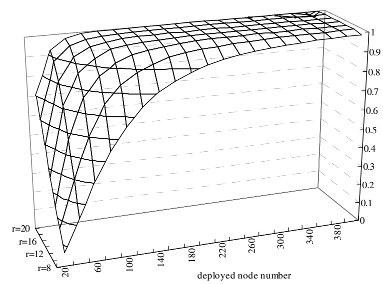
3 Simulations

To analyze the performance of our scheme, we carried out a number of experiments in static networks. We deployed 20 ~ 400 nodes in a square space (100 x 100). Nodes' x- and y-coordinates are set randomly. Each node had a sensing range of 8~20 meters and knew its neighbors. We let each active node decide whether to report or not based on its report probability. In this experiment, we assume that the sensing coverage of node is similar to the radio coverage because the node can deliver the sensing information via only radio. In fact, the existence of an optimal transmission radius in the request-spreading process suggests an advantage in having a transmission radius larger than the sensing radius because the sensing radius directly affects the average distance between area-dominant nodes. Moreover, enlarging the transmission radius can also benefit data-fusion schemes by allowing the construction of better-balanced trees. Our future research will study sensor networks in which the sensing and transmission radius are different in the future.

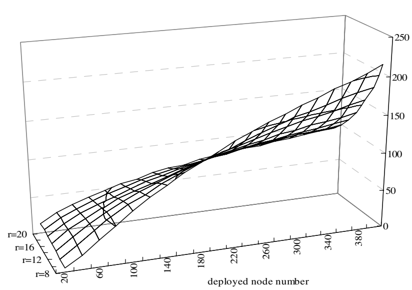
In our simulation, we assumed that the sink node is located in the center of the sensor field. Fig. 3a shows 3D surface plot of the coverage of all nodes in different sensing range and deployed node numbers. From this, we can see that increasing the number of the deployed nodes and increasing the sensing range will result in more nodes being idle, which is consistent with our expectation. Fig. 3b also shows 3D surface plot of the number of active nodes in different sensing ranges and deployed node numbers as well as Fig. 3a. We can see that the full coverage is preserved after turning off the inactive nodes.



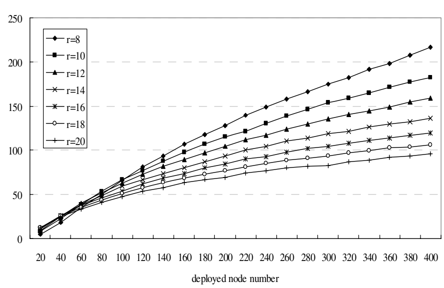
(a) The coverage of all nodes



(b) The coverage of active nodes



(c) Active node number (3D View)



(d) active node number (2D View)

Fig. 3. The coverage and active node number

Fig. 3c also shows a 3D surface plot of the number of active nodes in different sensing ranges and the deployed node numbers shown in Fig. 3b. We can see that the active node number increases slowly as the deployed node number increases. This means that the redundancy of SPCB scheme also increases when the deployed node number is high. These trends can be observed more precisely as illustrated in Fig. 3d.

Table 1. Coverage ratio vs. node density (r = 16)

Node density	Subset numbers		
	1	2	3
100	0.856047	0.952808	0.978567
200	0.823276	0.956346	0.987592
300	0.790469	0.946907	0.989562
400	0.784575	0.947001	0.988882

Table 1 summarizes coverage vs. the number of subsets. When the deployed node number is more than 100, the total coverage ratio reaches about 100% with three distinct subsets. For example, when the deployed node number is 400 and transmission range radius is 16, the number of report nodes in one subset is about 23 (see Fig. 8c and 9b), and the number of route nodes is about 100 (see Fig. 8c). This means that about 1/5 of all active nodes can monitor 70% of full-area coverage (see Fig. 7) and about 3/5 of all active nodes can monitor the entire coverage (Table 1).

4 Conclusions

This paper presents a Tri-State channel access scheme for wireless sensor networks. Our scheme consists of the SPCB and the PBB algorithm. The performance of our scheme is investigated via computer simulations, and the results show that only 20 ~ 50 % of the active nodes of all nodes can guarantee full-area coverage. In addition, one subset of active nodes can be constructed by 14 ~ 25 % of all active nodes (that is 4 ~ 7 subsets), and one subset can monitor about 70% of the area coverage. Additionally, 25 ~ 70 % of active nodes (that is, route nodes) can avoid their sensing duties. In other words, only three subsets (30 ~ 75 % of active nodes) can guarantee full-area coverage.

Acknowledgement. This research is supported by Program for the Training of Graduate Students for Regional Innovation.

References

[1] J. Gao, L.J. Guibas, J. Hershburger, L. Zhang, and A. Zhu, "Geometric Spanner for Routing in Mobile Networks," *Proc. Second ACM Symp. Mobile Ad Hoc Networking and Computing* (*MobiHoc 01*), pp. 45-55, Oct. 2001.

[2] J. Gao, L.J. Guibas, J. Hershburger, L. Zhang, and A. Zhu, "Discrete and Computational Geometry," *Proc. Second ACM Symp. Mobile Ad Hoc Networking and Computing* (*MobiHoc 01*), vol. 30, no. 1, pp. 45-65, 2003.

[3] K.M. Alzoubi, P.-J. Wan, and O. Frieder, "Message-Optimal Connected-Dominating-Set Construction for Routing in Mobile Ad Hoc Networks," *Proc. Third ACM Int'l Symp. Mobile Ad Hoc Networking and Computing* (*MobiHoc*), June 2002.

[4] Y. Wang and X.-Y. Li, "Geometric Spanners for Wireless Ad Hoc Networks," *Proc. 22nd Int'l Conf. Distributed Computing Systems* (*ICDCS 2002*), July 2002.

[5] A. Cerpa, N. Busek, and D. Estrin, "SCALE: A Tool for Simple Connectivity Assessment in Lossy Environments," *Technical Report CENS Technical Report 0021, Center for Embedded Networked Sensing*, Univ. of California, Los Angeles, Sept. 2003.

[6] K. Sohrabi and G. Pottie, "Performance of a Novel Self-Organization Protocol for Wireless Ad Hoc Sensor Networks," *Proc. IEEE Vehicular Technology Conf.*, Sept. 2000.

[7] W. Ye, J. Heidemann, and D. Estrin, "An Energy-Efficient MAC Protocol for Wireless Sensor Networks," *Proc. 21st Ann. Joint Conf. IEEE Computer and Comm. Soc.* (*INFOCOM*), pp. 1567-1576, June 2002.

[8] R. Zheng, J.C. Hou, and L. Sha, "Asynchronous Wakeup for Ad Hoc Networks," *ACM Int'l Symp. Mobile Ad Hoc Networking and Computing*, June 2003.

[9] B. Prabhakar, E. Uysal-Biyikoglu, and A.E. Gamal, "Energy-Efficient Transmission over a Wireless Link Via Lazy Packet Scheduling," *Proc. 20th Ann. Joint Conf. IEEE Computer and Comm. Soc.* (*INFOCOM*), pp. 386-394, Apr. 2001.

[10] R. Ramanathan and R. Rosales-Hain, "Topology Control of Multihop Wireless Networks Using Transmit Power Adjustment," *Proc. 19th Ann. Joint Conf. IEEE Computer and Comm. Soc.* (*INFOCOM*), pp. 404-413, Mar. 2000.

[11] Y. Xu, J. Heidemann, and D. Estrin, "Geography-Informed Energy Conservation for Ad Hoc Routing," *Proc. Seventh Ann. ACM/IEEE Int'l Conf. Mobile Computing and Networking* (*MobiCom*), pp. 70-84, July 2001.

[12] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, "Span: An Energy-Efficient Coordination Algorithm for Topology Maintenance in Ad Hoc Wireless Networks," *Proc. Seventh Ann. ACM/IEEE Int'l Conf. Mobile Computing and Networking* (*MobiCom*), pp. 85-96, July 2001.

[13] A. Cerpa and D. Estrin, "ASCENT: Adaptive Self-Configuring sEnsor Networks Topologies," *IEEE Transaction on Mobile Computing and Networking*, vol. 3, issue. 3, pp. 272-285, July 2004.

[14] D. Tian and N.D. Georganas, "A Coverage-Preserving Node Scheduling Scheme for Large Wireless Sensor Networks," *Proc. 1st ACM Workshop Wireless Sensor Networks and Applications*, pp. 32-41, 2002.

[15] J. Carle and D. Simplot-Ryl, "Energy-Efficient Area Monitoring for Sensor Networks," *IEEE Computer*, vol. 37, issue. 2, pp. 40-46, Feb. 2004.

A Cluster-Based Energy Balancing Scheme in Heterogeneous Wireless Sensor Networks

Jing Ai, Damla Turgut, and Ladislau Bölöni

Networking and Mobile Computing Research Laboratory (NetMoC)
Department of Electrical and Computer Engineering
University of Central Florida, Orlando, FL 32816
{jingai, turgut, lboloni}@cpe.ucf.edu

Abstract. In this paper, we propose a novel, cluster-based energy balancing scheme. We assume the existence of a fraction of "strong" nodes in terms of abundant storage, computing and communication abilities as well as energy. With the transformation of the flat network infrastructure into a hierarchical one, we obtained significant improvements in energy balancing leading to a longer connected time of the network. The improvement is quantified by mathematical analysis and extensive numerical simulations.

1 Introduction

Unbalanced energy consumption is an inherent problem in wireless sensor networks, and it is largely orthogonal to the general energy efficiency problem. For example, in a data gathering application, multi-hop wireless links are utilized to relay information to destination points called *sinks*. Inevitably, the nodes closer to the sink will experience higher traffic and higher energy consumption rate. These nodes will be the first ones which run out of power. Algorithms which allow "routing around" failed nodes will increase the load even more on the remaining active nodes close to the sink.

Our proposed cluster-based energy balancing scheme is intended to ameliorate the above energy unbalancing phenomena. We exploit the observation that in a heterogeneous sensor network there are nodes which are more powerful in terms of energy reserve and wireless communication ability. We transform the flat communication infrastructure into a hierarchical one where "strong" nodes act as clusterheads to gather information within the clusters and then communicate with the sink directly via single-hop link. In such a way, the "hot spot" around the sink is divided into multiple regions around the clusterheads in the hierarchical infrastructure. These distributed regions will assume fewer burdens due to the smaller scale of sensor nodes within the clusters.

2 A Cluster-Based Energy Balancing Scheme

2.1 Motivation

The sensor nodes usually collaborate with each other via multi-hop links. The multi-hop organization presents many advantages, from the increase of the network capacity,

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 467–474, 2005.

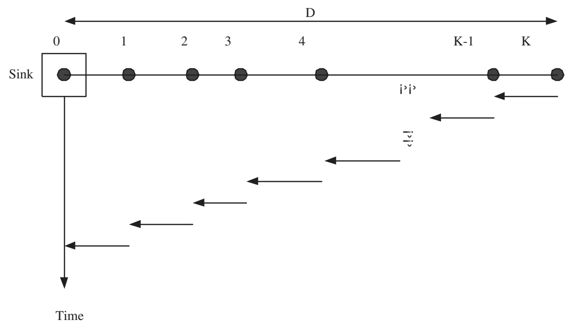


Fig. 1. $(K+1)$-node line network assumed to be connected. It illustrates a transmission schedule when only node K transmits a data packet to node 0 (sink) via multi-hop links

ability to perform data fusion and a more efficient energy utilization. However, under many scenarios, multi-hop sensor networks are utilizing energy in an unbalanced manner.

To illustrate this phenomena, let us consider a simple, unidirectional example in Figure 1. We assume that all nodes communicate only with their neighbors and all the nodes are sending their observations back to the sink. We assume the nodes to be equidistant, and thus the dissipated energy being roughly the same for each node. Normally, if all nodes have the same initial energy upon deployment, the node closer to the sink will drain earlier since it has heavier forwarding burden. Moreover, the further nodes which may still have plentiful energy supplies cannot find the routes to the sink. The energy unbalancing problem will aggravate with the increase of the network depth (defined as the largest number of hop from a node to the sink) [6].

The best resource utilization is achieved when every sensor node has the same rate of energy dissipation (or as close as possible), such that the network remains functional for the maximum possible time. Such a forwarding schedule is theoretically obtainable, by the algorithms proposed by Bhardwaj and Chandrakasan [2].

Although the proposed algorithm executes on polynomial time only, it also requires the global knowledge of the traffic, and thus is not feasible except for centrally managed networks and very large data packets. For a typical sensor network, where the individual measurements are small, the collection of global traffic information would be as expensive as the actual data communication itself. Our algorithm proposes a cluster-based organization of the traffic, which does not require global information, and proposes to ameliorate the energy unbalancing problem by decreasing or confining the network depth within each cluster.

2.2 Scheme Description

In a heterogeneous sensor network, we identify a subset of nodes as "strong" nodes with more powerful communication capabilities and energy resources. Instead of the flat organization of nodes, we assume a hierarchical structure where the strong nodes act as clusterheads. The clusterheads should be able to form a connected backbone between themselves such that they can communicate without relying on regular nodes. We assume two types of communication: one between the the regular nodes and the clusterheads with low transmission power, and the communication between clusterheads with higher transmission range spawning larger distances. In a practical deployment, these two types of traffic may be carried on different frequency bands or encoding techniques.

During the initialization phase, strong nodes broadcast their willingness to act as clusterheads. The sensor nodes decide to which cluster they wish to belong based on the strength of signal from the broadcast: the stronger the signal, the closer the clusterhead is and therefore the clusterhead with the strongest signal is chosen. At this point multiple clustering algorithms can be used, provided that they can be adapted to the specific condition of having a pre-determined clusterhead. On the other hand, algorithms which rely on dynamic leader election [5] are not appropriate for this purpose.

After the clusters are formed, the sensor nodes can use various algorithms for energy-efficient schedule for transmission such as in [6]. The clusterhead gathers the information from the sensor node within its cluster via multi-hop link and then forwards the aggregated information to the sink through the backbone of clusterheads.

This approach has all the desirable properties of similar schemes [3], such as localized traffic and scalability. The clusterheads are the natural points to implement data fusion and data compression algorithms. First, there is a potential correlation in the data from neighboring sensor nodes (given their physical proximity), and second, the higher energy resources of the strong nodes allows them to execute more complex computations. The proposed clustering scheme reduces the depth of the average multi-hop path to the clusterhead and transforms the single heavy "hot spot" around the sink to various distributed lighter "hot spots" around corresponding clusterheads. The ratio of the strong nodes to regular nodes determines the average depth of the multi-hop path inside the cluster. The essence of proposed the scheme explores the tradeoff between the multi-hop communication within the clusters and single-hop communication among clusters to achieve a better utilization of the energy resources.

3 Performance Evaluation

3.1 Preliminaries

To facilitate the performance analysis, we make the following assumptions:

i There is only one sink node with abundant energy resources.
ii There are N identical regular sensor nodes uniformly distributed in a planar disk whose radius is R.
iii There are S identical "strong" nodes with pre-determined locations in the same area such that they form clusters of roughly equal size.
iv The regular sensor nodes consume their energy much faster than the strong nodes such that the bottleneck is the energy of the regular nodes.
v The maximum transmission range, r, of regular sensor nodes ensures the connectivity of the network while the transmission range of "strong" nodes is large enough for strong nodes and the sink to form a connected backbone.
vi There is no interference between the communication in the backbone and the intra-cluster communication.

vii The nodes may fail only when they deplete their energy resource.
viii All nodes deploy an ideal MAC protocol and there is no collision among packets.
ix All nodes have an ideal sleep scheduling and consume energy only during transmission and reception.

The energy consumption of a sensor node is divided between the three components of a wireless sensor: sensing, computation and communication components [2].

1. Sensing: We assume that every sensor node captures b bits/sec data from its environment. The energy needed to sense a bit of data is (α_3). Thus, the energy comsumed for sensing is $p_{sense} = \alpha_3 b$.
2. Computation: The computational power of a sensing node is used for operations, such as data aggregation. It is difficult to quantify the energy used for data agregation in absolute terms without specific knowledge about the nature of the data. However, in our analysis, we are interested in the *relative* performance of the hierarchical organization against a flat network of sensor nodes. We will assume that any scheme will benefit both organizations approximately equally, thus we will ignore this term in our calculations.
3. Communication: We use the following model for the energy dissipation used for communication [7]:

$$p_{tx}(n_1, n_2) = (\alpha_{11} + \alpha_2 d(n_1, n_2)^n) b \tag{1}$$

$$p_{rx} = \alpha_{12} b \tag{2}$$

where $p_{tx}(n_1, n_2)$ is the power dissipated in node n_1 when it is transmitting to node n_2, $d(n_1, n_2)$ is the distance between the two nodes, n is the path loss index, and the α_i are positive constants.

3.2 Analysis

The energy consumption of the wireless sensor network is determined by the spatial distribution of the sensor nodes. Although in our approach the strong nodes are in predetermined locations, the distribution of the locations of the regular nodes is essentially random. Thus, our analysis will be based on establishing lower bounds of the energy consumptions. We will rely on two theorems introduced in [1]:

Theorem 1. Given D and number of intervening relays ($K-1$) as shown in Figure 1, $P_{link}(D)$ is minimized when all the hop distances are equal to $\frac{D}{K}$.

This theorem gives us a bound of energy dissipation rate in a line network via multihop links. It is interesting to note that increasing the number of hops can effectively decrease the transmission power while increase the reception power. There is an optimal number of hops K_{opt} which minimizes the total energy dissipation by trading of the power consumed for transmission and reception.

Theorem 2. The optimal number of hops K_{opt} is always one of

$$K_{opt} = \lfloor \frac{D}{d_{char}} \rfloor or \lceil \frac{D}{d_{char}} \rceil \tag{3}$$

where the distance d_{char} , called the characteristic distance, is independent of D and is given by

$$d_{char} = \sqrt[n]{\frac{\alpha_1}{\alpha_2(n-1)}} \tag{4}$$

We conclude, that for any path loss index n, the energy cost of transmitting a bit can always be made linear with distance. Moreover, for any given distance D, there is an optimal number K_{opt} of intervening nodes. Using more or less than this optimal number leads to energy inefficiencies.

Case I: Flat Network Architecture. In our environment, there are N identical sensor nodes uniformly distributed in a planar disk of radius R. Using the results from [1], we derive the lower bound of the energy dissipation rate:

$$P_{flat\_network} \geq (\sum_{i=1}^{N} \alpha_1 \frac{n}{n-1} \frac{d_i}{d_{char}} - N\alpha_{12})b \tag{5}$$

where, d_i is the distance of sensor i from the center of the disk.

Thus, the expected value of the lower bound of dissipated energy is as follows:

$$E[min(P_{flat\_network})] = [\alpha_1 \frac{n}{n-1} \frac{RN}{2d_{char}} - N\alpha_{12}]b \tag{6}$$

Case II: Hierarchical Clustering Scheme. According to the clustering scheme described above, when S "strong" nodes are deployed, S clusters will automatically be formed. In each cluster, the expected number of nodes is $\frac{N}{S}$.

The individual clusters have a similar structure like the flat network, but we also need to consider both the reception energy consumption of the strong nodes and the energy consumption related to the communication between the strong node and the sink, which follows the equation (1):

$$P_{clustered\_network} \geq S \sum_{i=1}^{\frac{N}{S}} \alpha_1 \frac{n}{n-1} \frac{d_i}{d_{char}} b + \sum_{i=1}^{S} (\alpha_{11} + \alpha_2 d_i'^n) b \tag{7}$$

where, d_i is a random variable following the uniform distribution over the interval [0, $\frac{R}{\sqrt{S}}$] and $d_i'^n$ is a random variable following the uniform distribution over the interval [0, R].

Thus, the expected value of the minimum $min(P_{clustered\_network})$ is as follows:

$$\begin{aligned} E[min(P_{clustered\_network})] = [\alpha_1 \frac{n}{n-1} \frac{RN}{2\sqrt{S} d_{char}} + S\alpha_{11} + \\ \alpha_2 S \frac{R^n}{n+1}]b \end{aligned} \tag{8}$$

An important consequence is that the communication cost of multi-hop links increases with the number of clusters while the communication cost of messaging on the backbone increases with the number of clusters. Thus, there exists an optimal number of clusters which trades off the power consumption between multi-hop and single-hop links to minimize the energy dissipation rate. Applying the techniques of previous two theorems, we can deduct that optimal number clusters is always one of

$$S_{opt} = \lfloor (\frac{\alpha_1 \frac{n}{n-1} \frac{RN}{d_{char}}}{\alpha_{11} + \alpha_2 \frac{R^n}{n+1}})^{\frac{2}{3}} \rfloor or \lceil (\frac{\alpha_1 \frac{n}{n-1} \frac{RN}{d_{char}}}{\alpha_{11} + \alpha_2 \frac{R^n}{n+1}})^{\frac{2}{3}} \rceil \quad (9)$$

This result is important from the practical deployment point of view of a sensor network. We need to limit the number of clusters to the one shown in the Equation 9 even if we have a larger number of nodes which, based on their hardware characteristics, would qualify as "strong" nodes.

3.3 Numerical Simulation

We will numerically analyze the energy dissipation rate of our scheme compared with the flat network architecture. In addition, we examine the impact of the various parameters, n, N, R, S. We assume a sensor network is composed of $N = 10000$ sensor nodes distributed on a radius of $R = 1000$ meters with communication path loss index $n = 2$ and data bit rate $b = 1 bits/sec$.

Figure 2, left, shows the energy dissipation in function of the number of clusters S ranging from $1\% \sim 10\%$ of N. Another property of interest is the optimal *percentage* of clusterheads or strong nodes. Thus, in Figure 2 right, we plot the calculated optimum percentage in function of the total number of nodes, N. We found that the optimal percentage of strong nodes decreases with the number of total sensor nodes and it is between 9% to 2% in a typical field of 10,000 to 100,000 nodes of deployment. Thus, the remarkable gain in energy dissipation rate can be obtained with relatively small percentage of strong nodes.

Next, we determine the optimum number of "strong" nodes with the increase of the R while keeping other parameters unchanged as can be seen in Figure 3. Thus, we

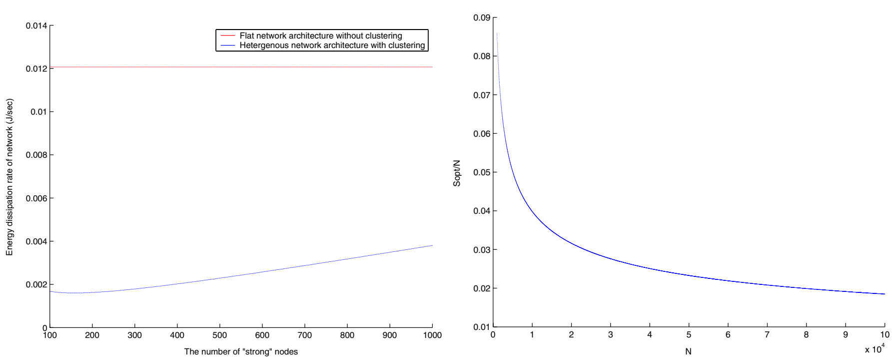


Fig. 2. Energy dissipation vs. number of clusters (left) and the optimal percentage of clusterheads to the total number of nodes, N (right)

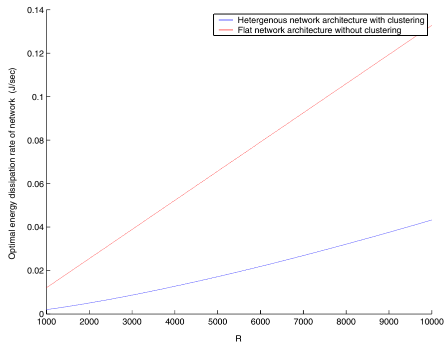


Fig. 3. The impact of network density on the optimal performance of two paradigms of networks where $n = 2$, $N = 10000$

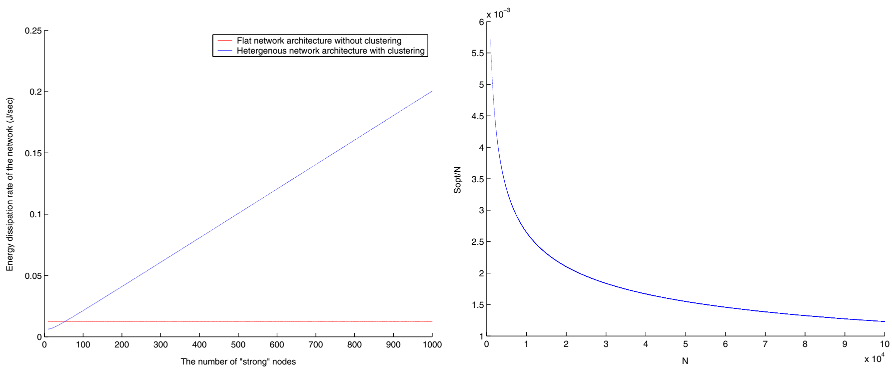


Fig. 4. Performance comparison of two paradigms of networks where $n = 4$, $N = 10000$, $R = 1000$ (left) and the relationship between $\frac{S_opt}{N}$ and N where $n = 4$, $R = 1000$ (right)

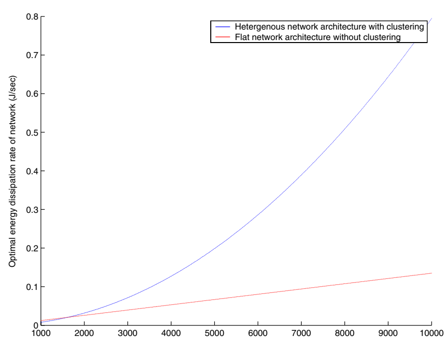


Fig. 5. The impact of network density on the optimal performance of two paradigms of networks where $n = 4$, $N = 10000$

visualize the impact of the density of the network on the optimal energy dissipation rate when the optimal number of "strong" nodes is deployed.

By repeating the experiments with $n = 4$, we obtained the results in Figure 4 and 5. Contrary to our expectations, our clustering scheme does not show any benefits for this experimental setup. This is explained by the fact that in an environment with large path loss index, the single-hop operations are much more expensive than multi-hop communications. We conclude that the benefits of our scheme is highly dependent on the environment and the it is better adapted for low path loss index values.

4 Conclusions

Through introducing a series of "strong" nodes as clusterheads, we change the communication structure of the original data fusion in wireless sensor networks from a flat to an hierarchical one which has better energy-balancing properties. Compared to other energy-balancing schemes, our scheme is rather simple and effective. Future work includes adapting the protocol that does not depend on neither the environment nor the path loss index as well as an extensive simulation work to validate the analytical results.

References

1. M. Bhardwaj, T. Garnett, A.P. Chandrakasan. "Upper Bounds on the Lifetime of Sensor Networks". *In Proceedings of International Conference on Communications (ICC)*, 2001.
2. M. Bhardwaj, A.P. Chandrakasan. "Bounding the Lifetime of Sensor Networks via Optimal Role Assignments". *In Proceedings of INFOCOM 2002*, pp. 1587-1596, New York, June 2002.
3. W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan. "Energy-Efficient Communication Protocol for Wireless Microsensor Networks". *Proceedings of the 33rd International Conference on System Sciences (HICSS '00)*, January 2000.
4. W. R. Heinzelman. "Application-Specific Protocol Architectures for Wireless Sensor Networks". *Ph.D. Thesis, Massachusetts Institution of Technology*, 2000.
5. H. Garcia-Molina. "Election in a distributed computer system". *IEEE Transactions on Computers*, C-31(2), pp. 48-59, 1982.
6. M. L. Sichitiu. "Cross-Layer Scheduling for Power Efficiency in Wireless Sensor Networks". *In Proceedings of INFOCOM 2004*, Hong Kong, P. R. China, March 2004.
7. T. Rappaport. *Wireless Communications: Principles & Practice*. New Jersey; Prentice-Hall, Inc., 1996.

An Optimal Node Scheduling for Flat Wireless Sensor Networks

Fabíola Guerra Nakamura, Frederico Paiva Quintão,
Gustavo Campos Menezes, and Geraldo Robson Mateus

Federal University of Minas Gerais – UFMG – Brazil
{fgnaka, fred, gcm, mateus}@dcc.ufmg.br

Abstract. The determination of a topology that minimizes the energy consumption and assures the application requirements is one of the greatest challenges about Wireless Sensor Networks (WSNs). This work presents a dynamic mixed integer linear programming (MILP) model to solve the coverage and connectivity dynamic problems (CCDP) in flat WSNs. The model solution provides a node scheduling scheme indicating the network topology in pre-defined time periods. The objective consists of assuring the coverage area and network connectivity at each period minimizing the energy consumption. The model tests use the optimization commercial package CPLEX 7.0. The results show that the proposed node scheduling scheme allows the network operation during all the defined periods guaranteeing the best possible coverage, and can extend the network lifetime besides the horizon of time.

1 Introduction

A Wireless Sensor Network (WSN) is a special kind of an ad hoc network composed by autonomous and compact devices with sensing, communication, and processing capacities, called sensor nodes [1]. Basically, in a WSN application, the sensor nodes are deployed over an area to collect data from a phenomenon. The data are disseminated from source nodes to sink nodes and then to an observer [2], where they are processed and provide information about the environment.

There are several challenges regarding WSNs once these networks present several particularities as energy restrictions, node redundancy,limited bandwidth, and dynamic topology. These unique features allow a wide variety of research in energy-efficient network protocols, low-power hardware design and encourage proposals of management architectures for WSNs, which aim to increase the network resources productivity, and to maintain the quality of service [3].

This work presents a dynamic mixed integer linear programming (MILP) model, whose solution determines an optimal node scheduling for flat WSNs. The objective function aims to minimize the network energy consumption and the model constraints assure the quality of service requirements such as coverage, connectivity, with respect to nodes energy restrictions. The work contribution

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 475–482, 2005.

is a mathematical formulation that models coverage, connectivity, and energy WSNs features, and whose solutions can be inserted in a WSN management scheme.

The remainder of the paper is organized as follows: In the next section we present the model proposed. Section 3 contains the experimental results and their analysis. We list the related work in section 4. In section 5 we present our conclusions and describe the directions of our future work.

2 Dynamic Mixed Integer Linear Programming Model

2.1 Basic Concepts

Coverage in Wireless Sensor Networks. In order to quantify the coverage area of a WSN, we define the node sensing area as the region around the node where a phenomenon can be detected and define this region as a circle of range R, where R is the sensing range [4]. The coverage area of a WSN consists of the sensing areas union of all active nodes in the network.

The coverage area is modelled through the use of demanda points, which represent the center of a small square area in the sensor field. This concept allows to evaluate the coverage in a discrete space and is useful for modelling purposes. To guarantee the coverage, at least one active sensor should cover each demand point, otherwise the coverage fails.

Energy Consumption Model. One of the main features of WSNs is a high energy restriction, due to the limited sensor node battery, and to the impossibility of battery recharge. The definition of a node energy consumption model can allow WSNs researches to focus the studies on topics that have higher impacts on the network lifetime [5]. The node operations consumption depends on the current necessary to perform the task and time period to execute the task. The energy consumed can be estimated by the following equation:

$$E = \alpha \times \Delta t$$

where: E is the total energy consumed in mAh.
α is the current consumed in mA.
Δt is the period of time in h.

The WSN application dependency makes really important that we define a work scenario. On the development of our model we make the following assumptions: each sensor node knows its localization, and has an unique id, the application requirements are continuous data collection and periodic data dissemination, and battery discharge follows a linear model. Only source nodes generate traffic in the network.

2.2 Mathematical Formulation

Our problem can be stated as: *Given a sensor field A, a set of demand points D, a set of sensor nodes S, a set of sink nodes M, and t time periods the coverage* *and connectivity dynamic problem (CCDP) consists of assuring that at least m sensor nodes $i \in S$ are covering each demand point $j \in D$ in the sensor field A, and that there is a path between these nodes, and a sink node $j \in M$ in each time period.*

The CCDP is formulated as a mixed integer linear programming(MILP) problem. The following parameters are used in our formulation:

S set of sensor nodes
M set of sink nodes
D set of demand points
T set of time periods
A^d set of arcs that connect sensor nodes to demand points
A^s set of arcs that connect sensor nodes
A^m set of arcs that connect sensor nodes to sink nodes
$E^d(A)$ set of arcs $(i,j) \in A$ entering on the demand point $j \in D$
$E^s(A)$ set of arcs $(i,j) \in A$ entering on the node $j \in S$
$S^s(A)$ set of arcs $(i,j) \in A$ emanating from the node $i \in S$
n defines the number of nodes the should cover a demand point
BE node battery capacity
AE_i energy to activate a node $i \in S$
ME_i energy to keep a node $i \in S$ active during a time period $t \in T$
TE_ij energy to transmit packets from $i \in S$ to $j \in S$ during a time period $t \in T$
RE_i energy to recept packets in node $i \in S$ during a time period $t \in T$
HE_j penalty of no coverage of a demand point $j \in D$ during a time period $t \in T$

The model variables are:

x_{ij}^t has value 1 if node $i \in S$ covers demand point $j \in D$ on time period $t \in T$, and 0 otherwise
z_{lij}^t has value 1 if arc (i,j) is in the path between sensor node $l \in S$, and a sink node $m \in M$ on time period $t \in T$, and 0 otherwise
w_i^t has value 1 if node $i \in S$ is activated on time period $t \in T$, and 0 otherwise
y_i^t has value 1 if node $i \in S$ is active on time period $t \in T$, and 0 otherwise
h_j^t indicates if demand point $j \in D$ is not covered on time period $t \in T$
e_i indicates the value of the energy consumed by node $i \in S$ during the network lifetime

The model proposed is presented below. The objective function 1 minimizes the network energy consumption during its lifetime.

$$min \sum_{i \in S} e_i + \sum_{j \in D} \sum_{t \in T} EH_j^t \times h_j^t \tag{1}$$

Constraints (2), (3), (4), and (5) deal with the coverage problem. They assure that the active nodes cover the demand points. Constraints (2) also assure the possibility of a demand point not be covered. A demand point is not covered when it is not in the coverage area of any active node or when the node that could cover it has no residual energy.

$$\sum_{ij \in E_j^d(A^d)} x_{ij}^t + h_j^t \geq n, \forall j \in D \text{ e } \forall t \in T \tag{2}$$

$$x_{ij}^t \leq y_i^t, \forall i \in S, \forall ij \in A^d \text{ e } \forall t \in T \tag{3}$$

$$0 \leq x_{ij}^t \leq 1, \forall ij \in A^d \text{ e } \forall t \in T \tag{4}$$

$$h_j^t \geq 0, \forall j \in D \text{ e } \forall t \in T \tag{5}$$

Constraints (6), (7), (8), and (9) are related to the connectivity problem. They impose a path between each active sensor node and a sink node.

$$\sum_{ij \in E_j^s(A^s)} z_{lij}^t - \sum_{jk \in S_j^s(A^s \cup A^m)} z_{ljk}^t = 0, \forall j \in (S \cup M - l), \forall l \in S \text{ e } \forall t \in T \tag{6}$$

$$- \sum_{jk \in S_j^s(A^s \cup A^m)} z_{ljk}^t = -y_l^t, j = l, \forall l \in S \text{ e } \forall t \in T \tag{7}$$

$$z_{lij}^t \leq y_i^t, \forall i \in S, \forall l \in (S - j), \forall ij \in (A^s \cup A^m) \text{ e } \forall t \in T \tag{8}$$

$$z_{lij}^t \leq y_j^t, \forall j \in S, \forall l \in (S - j), \forall ij \in (A^s \cup A^m) \text{ e } \forall t \in T \tag{9}$$

The node residual energy is defined by constraints (10) which indicate that a node can only be active if it has residual energy, and by (11), and (12), this energy must be nonnegative and less than the battery capacity.

$$\textstyle\sum_{t \in T}(EM_i \times y_i^t + EA_i \times w_i^t + \sum_{l \in (S-i)} \sum_{ki \in E_i^s(A^s \cup A^m)} ER_i \times z_{lki}^t +$$

$$\sum_{l \in S} \sum_{ij \in S_i^s(A^s \cup A^m)} ET_{ij} \times z_{lij}^t) \leq e_i, \forall i \in S \tag{10}$$

$$e_i \leq EB_i, \forall i \in S \tag{11}$$

$$e_i \geq 0, \forall i \in S \tag{12}$$

The constraints (13), and (14) indicate activation node period.

$$w_i^0 - y_i^0 \geq 0, \forall i \in S \tag{13}$$

$$w_i^t - y_i^t + y_i^{t-1} \geq 0, \forall i \in S, \forall t \in T \text{ e } t > 0 \tag{14}$$

Constraints (15) define the variables y,z, and w as boolean, and constraints (16) define the variables x,h, and e as real.

$$y, z, w \in \{0, 1\} \tag{15}$$

$$x, h, e \in \Re \tag{16}$$

For each time period, the model solution indicates which nodes are actives, which demand points are not covered, and provides a path between the actives nodes and the sink node, guaranteeing the network connectivity. The solution also estimates the network energy consumption.

3 Experimental Results

3.1 Input Parameters

We consider a flat network, and homogeneous nodes. The sensor nodes are deployed over the sensor field in a random way with uniform distribution.

The model input parameters are: one demand point for each m^2, $625m^2$ sensor field, 16 sensor nodes, one sink node in the center or in corner of the area, and coverage guaranteed by $n = 1$ or $n = 2$. The energy parameters are based on the values provided by the supplier, [6], that brings the current consumption of the sensor node MICA2. Besides that we work with instances of 4 time periods and a battery capacity that allows the nodes to be active for two periods.

3.2 Computational Results

The tests use the optimization commercial package CPLEX 7.0 [7]. The optimal solutions for instances with the sink node place in the center of the sensor field are in Table 1. The value of active nodes is the arithmetic mean of active nodes in each period. The value of coverage fail is the arithmetic mean of the fail (demand points not covered / total of demand points) in each period. The standard deviation regards the value of this mean.

The results for instances with the sink node in the bottom left corner of the sensor field are in Table 2. The results for instances with the sink node in the center of the sensor field and precision $n = 2$ are in Table 3. For these test we show the coverage fail as total coverage fail and parcial coverage fail. The first one represents the arithmetic mean of non covered demand points and the second the arithmetic mean of demand points covered for one sensor node. The demand points covered by only one node can be seen as areas whose sensing data are less precise, but that still can be used by the observer to infer environment features.

Comparing the results of Table 1 and Table 2 we notice that when we move the sink node to the sensor field corner the number of actives sensor nodes

Table 1. Optimal Solution for 1 sink node in the center

Communication Range (m)	Sensing Range (m)	Active Nodes	Standard Deviation (nodes)	Energy Consumption (mAh)	Coverage Fail (%)	Standard Deviation (coverage)
7.5	7.5	1.5	1.73	43.95	78.91	24.20
7.5	10	1.5	1.73	43.95	70.50	33.90
7.5	12.5	1.0	1.73	27.23	61.58	44.25
10	7.5	7.5	0.58	211.83	7.0	3.60
10	10	7.0	0.0	195.02	0.6	0.64
10	12.5	5.0	0.0	136.36	0.0	-
12.5	7.5	7.0	0.0	184.81	2.24	0.74
12.5	10	6.5	0.58	172.19	0.0	-
12.5	12.5	4	0.0	103.00	0.0	-

Table 2. Solution for 1 sink node in the corner

Communication Range (m)	Sensing Range (m)	Active Nodes	Standard Deviation (nodes)	Energy Consumption (mAh)	Coverage Fail (%)	Standard Deviation (coverage)
7.5	7.5	1.5	1.73	43.95	80.00	23.00
7.5	10	1.5	1.73	43.95	73.24	30.76
7.5	12.5	1.5	1.73	43.95	67.50	37.42
12.5	12.5	5.5	0.58	159.70	0.0	-

Table 3. Optimal Solution for precision $n = 2$

Communication Range (m)	Sensing Range (m)	Active Nodes	Total Coverage Fail (%)	Standard Deviation (total coverage)	Parcial Coverage Fail (%)	Standard Deviation (parcial coverage)
7.5	7.5	1.5	78.10	24.20	10.4	12.10
7.5	10	1.5	70.52	33.90	10.7	12.38
7.5	12.5	1.5	61.58	44.25	11.10	12.84
12.5	7.5	8	3.60	1.39	42.17	2.03
12.5	10	8	0.80	0.74	7.75	1.02
12.5	12.5	7	0.00	-	1.11	1.30

increase because the path to sink also increases. Table 3 shows that the greater the precision is, the more the actives nodes are.

The high coverage fail and standard deviation values for the communication range of 7.5m have two main causes: low network connectivity and battery capacity. The low network connectivity, due to the short communication range, allows the activation of few nodes because if there is no path between the source node and one of the active sink nodes this node remains inactive. Besides that, in all tests we use a battery capacity that allows all nodes to remain actives for two periods only.

The results show that the model is sensible to different sensing range values, the greater the range is, the less the actives nodes are. However, regarding the communication range this affirmation is not true, because the communication range assures the network connectivity and when this range is really short and the nodes cannot reach each other, they are not activated.

The model's main problem is its complexity, which requires a great computational effort to find the solutions and for some instances it is impossible to reach an optimal or even a feasible solution at reasonable time.

3.3 Energy Consumption

The energy savings with the node scheduling are evaluated comparing networks with and without node scheduling schemes. We assume that in the network without scheduling all nodes are active, and the model solutions provides the routes

Table 4. Energy consumption comparison

Period	With scheduling		Without scheduling	
	Active Nodes	Energy Consumption (mAh)	Active Nodes	Energy Consumption (mAh)
0	8	6137,283	16	12274,464
1	8	5961,283	16	11922,646
2	8	6137,181	16	11003,073
3	8	5961,181	0	0,000

for data dissemination. Table 4 presents the comparison between topologies with and without scheduling for an area of $3600m^2$, 16 sensor nodes, four sink nodes in the sensor field corners, communication range of 25m, sensing range of 15m, and grid positioning. As we can note, without scheduling there is no active node after the third period. Although the node scheduling can causes coverage fail, it allows network activities during all time periods, because the solution can schedule nodes in all periods assuring the best possible coverage.

4 Related Works

Megerian et al. [8] propose several ILPs models to solve the coverage problem. Their focus is the energy efficient operation strategies for WSN. This approach is similar to ours, except that it defines areas sets that should be covered instead of demand points, and the work does not deal with dynamic problems.

Chakrabarty et al. in [9] present a Integer Linear Programming (ILP) Model that minimizes the cost of heterogenous sensor nodes, and guarantees sensor field coverage. Their problem is defined as the placement of sensor nodes on grid points, and they propose two approaches: a minimum-cost sensor placement, and a sensor placement for target location.

The dynamic multi-product problem of facilities location is formulated for Hinojosa, Puerto, and Fernández in [10] in a mixed integer linear programming model. In this work the objective is to minimize the total cost to demand attendance of the products in a planning horizon and also assure that the producers and intermediate deposits capacities are not exceeded. The problem lower bound is obtained by Lagrangian Relaxation. With this solution a heuristic is used to obtain feasible solutions.

5 Conclusion

This work presents a dynamic mixed integer linear programming (MILP) model to solve the coverage and connectivity dynamic problem(CCDP) in flat WSNs. The model optimal solution indicates the set of sensor nodes that should be actives to guarantee the sensor field coverage and a path between each active sensor node and a sink node for each time period. The solution is chosen in order to minimize the network energy consumption.

In general we can conclude that the dynamic planning as proposed save energy compared with a network without node scheduling and also assure activity during all periods. The model provides a route between the source nodes, and the sink node, and different routing protocols can be used over the topology provided for the model solution.

Future work includes the development of algorithms and heuristics to solve bigger problems and to decrease the solution time because the model complexity requires a great computational effort and sometimes it is impossible to reach an optimal solution in reasonable time. The first chosen technique is Lagrangian Relaxation [11].

Acknowledgments

This work is supported by The National Council for Scientific and Technological Development (CNPq), Brazil. Process 55.2111/2002-3.

References

1. Rentala, P., Musunuri, R., Gandham, S., Saxena, U.: Survey on sensor network. In: Mobile Computing (CS6392) Course. (2001)
2. Tilak, S., Abu-Ghazaleh, N.B., Heinzelman, W.: A taxonomy of wireless micro-sensor network models. ACM SIGMOBILE Mobile Computing and Communications Review **6** (2002) 28–36
3. Ruiz, L., Nogueira, J., Loureiro, A.: Manna: A management architecture for wireless sensor networks. IEEE Communication Magazine **41** (2003)
4. Wang, X., Xing, G., Zhang, Y., Lu, C., Pless, R., Gill, C.: Integrated coverage and connectivity configuration in wireless sensor networks. In: First ACM Conference on Embedded Networked Sensor Systems (SenSys'03). (2003)
5. Bhardwaj, M., Chandrakasan, A., Garnett, T.: Upper bounds on the lifetime of sensor networks. In: IEEE International Conference on Communications. (2001) 785 – 790
6. Crossbow: (Mica2 - wireless measurement system) http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MICA2.pdf.
7. ILOG: (Cplex) http://www.ilog.com/products/cplex/.
8. Megerian, S., Potkonjak, M.: Low power 0/1 coverage and scheduling techniques in sensor networks. Technical Reports 030001, University of California, Los Angeles (2003)
9. Chakrabarty, K., Iyengar, S.S., Qi, H., Cho, E.: Grid coverage for surveillance and target location in distributed sensor networks. In: IEEE Transactions on Computers. (2002) 51(12):1448–1453
10. Hinojosa, Y., Puerto, J., Fernández, F.R.: A multiperiod two-echelon multicommodity capacitaded plant location problem. European Journal of Operational Research **123** (2000) 271–291
11. Fisher, M.L.: An applications oriented guide to lagrangian relaxation. Interfaces **15** (1985) 10–21

A Congestion Control Scheme Based on the Periodic Buffer Information in Multiple Beam Satellite Networks

Seungcheon Kim

Department of Information&Telecommunication Eng.,
Hansung Univ., Seoul, Korea
kimsc@hansung.ac.kr

Abstract. This paper introduces a new congestion control scheme improving performance of the future broadband satellite networks. The proposed scheme regulates the data rate based on the buffer information sent by the satellite in periodic manner. The complexity of the proposed scheme is comparable with the existing flow control techniques, as it does not require the additional information exchange with the satellite. The throughput and the satellite queue size performances of the proposed scheme are mathematically analyzed. The results show the significant improvement in the proposed scheme comparing with the conventional window-based and rate-based congestion control techniques.

1 Introduction

Considering the data services through satellites, all the impairments encountered first are the long transmitting delay and the high bit error rate including the occurrence of errors in burst [1]. Those impairments can be covered with an aid of enhanced transmitting method but still remains as obstacles degrading the performances of the data services through satellite [2]. Related with the architecture and the basic construction the satellite networks will be built on, many efforts have been carried out [3-4]. Some of them are defining the satellite networks on the basis of the Asynchronous Transfer Mode (ATM) and others are on IP [5]. Whatever they are based on, however, the main systemic mechanisms controlling transmitting and receiving data through satellite should be modified or substituted with more suitable ones in satellite networks. One of those would be the congestion control scheme [6]. A well know Internet transport protocol, TCP, is also designed to be used in the wired networks with low BER on the order of 10^{-8}. Hence in different environments with different link characteristics, TCP would not perform well as in the terrestrial networks. ATM would show the same result in the satellite networks.

Here is our motivation to propose a new congestion control mechanism working in the multiple spot beam satellite networks. The proposed scheme is based on the information that is sent by the satellite. This information is just broadcasted in each beam areas periodically. Based on the information, satellite terminals and earth stations regulate their data rate to avoid congestion in satellite switch. This will act on improving the total communication performance significantly.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 483–491, 2005.

2 Congestion Control Schemes

In a multiple spot beam satellite environment, satellites have switching capability among beam areas, which means that each satellite has the ability of on-board processing (OBP) and on-board switching (OBS). In such an environment, the network congestion may occur just as in the terrestrial networks. The buffer overflow can also happen easily in the case that the whole data of the satellite network rush into a single beam area not only to one specific station. We, thus, need a congestion control scheme especially suitable to the satellite networks that have different network characteristics.

2.1 Window-Based Congestion Control

This is the typical method that is used mainly in Internet. It allows sender to regulate the number of sending packets by adjusting window size according to the network situations. In this scheme, window size is increased in a situation that there is no congestion in the network and decreased rapidly to reduce the traffic amount when the congestion occurs. Normally a congestion at the node is indicated when the length of buffer goes beyond a certain threshold. The window based flow control can be described as the following equations:

$$\begin{aligned} W &\leftarrow d \cdot W, && \text{when } Q \geq L \\ W &\leftarrow W + b, && \text{when } Q < L \end{aligned} \tag{1}$$

where W is the window size, d is the decrease factor, b is the increase factor, Q is the queue size, and L is the buffer threshold.

2.2 Rate-Based Congestion Control

The ATM Forum proposed rate-based flow control for providing ABR service. [7] This scheme can be described by two schemes, one of which is rate control using explicit forward congestion indication (EFCI) bit of resource management (RM) ATM cell and the other is explicit rate (ER) control, which indicates the specific data rate to the sending node from the intermediate nodes. As the method of rate regulation, there are proportional rate control algorithm (PRCA) and enhanced PRCA (EPRCA) that adopt ER at PRCA. Basically satellite communication network has long transmission delay, e.g. 250ms round trip time (RTT) in a GEO satellite communication system. The method that the satellite indicates the data rate of each earth station could cause unpredictable results or severe congestion because the time gap of data rate indication of satellite and effect time of the data rate to the satellite would be same as the round trip time. This data rate indication would increase the burden of the satellite and it is better to consider only EFCI in the satellite network in order not to increase the complexity and processing cost of satellite communications.

In PRCA, data rate starts from a specific level and reduces exponentially. If the sender doesn't see the EFCI setting of the periodic RM cell on receiving RM cells, data rate is increased to a specific one. Otherwise, data rate is decreased continuously.

2.3 Proposed Congestion Control Scheme

In satellite communication, because of the long propagation delay, the data rate from the earth station can affect the satellite with some delay. Therefore, even if the control is performed promptly by the earth station after receiving information from the satellite, it will be effective to the satellite after some time later.

As a result, any rate control or window control based on the feedback information of the satellite can hardly change the situation of the satellite. On contrary, it can cause the buffer overflow and underflow of the satellite switch [8]. Therefore, in order to utilize the buffer information efficiently, we need to keep in mind the distance between satellite and earth stations or satellite terminals and consider the buffer level variation rather than the specific buffer levels.

In the proposed scheme, the data rate in the earth station is varying like stepwise movement based on the information that is broadcasted through the control channel by the satellite periodically. The satellite simply broadcasts the buffer information to the each beam area and earth stations regulate the data rate based on that buffer information. Since the proposed scheme has stepwise characteristics in rate variation, we will refer it to as step-increase step-decrease (SISD) scheme.

In SISD rate control, when the information from the satellite indicates that the buffer size is below a certain threshold, earth stations increase the data rate by unit step rate after maintaining the current data rate for a period of time equal to a RTT. If the buffer level is above the threshold, the data rate is decreased by the step size rate. But if the level is higher than the threshold and is decreasing, the current data rate is kept. This procedure can be summarized as follows:

(1) When queue level at satellite is below the threshold, then “Step Increase”.

(2) When queue level at satellite is above the threshold and increasing; i.e. $dq(t) / dt \geq 0$, then “Step Decrease”.

(3) When queue level at satellite is above the threshold and decreasing; i.e. $dq(t) / dt > 0$, then current data rate is maintained.

3 Mathematical Analysis

To see the performance of the each scheme, we assume a simple queuing situation as explained in the followings. Data arrival rate is assumed to be $\lambda(t)$ and service rate to be $\mu(t)$ based on the *fluid-flow* approximation. τ_f and τ_b represent forward delay and feedback delay from satellite to earth stations, respectively. Thus, the total delay is $\tau = \tau_f + \tau_b$. For the sake of simplicity in analysis, we assume the buffer threshold to be zero. Maximum queue length of the satellite and average throughput of window control, PRCA and proposed SISD schemes are compared through deterministic analysis [9-10].

3.1 Window Control

The basic rule of window-based flow control can be summarized as follows:

$$W \leftarrow d \cdot W, \quad \text{if } q(t - \tau_f) > 0 \text{ and } W \leftarrow W + b, \quad \text{if } q(t - \tau_f) = 0 \tag{2}$$

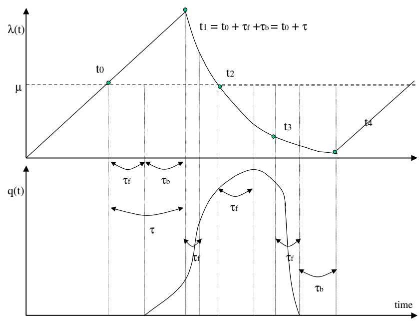


Fig. 1. Rate and Queue size variation of Window control

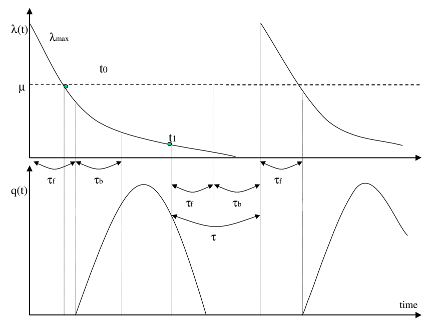


Fig. 2. Rate and queue size variations in PRCA

where *q(t)* means the queue length at time '*t*' on the satellite. This can be translated into the rate variation as follows because the variation of window size means the variation in number of packets that can be sent:

$$\frac{d\lambda(t)}{dt} = \alpha, \text{if } \ q(t-\tau_b) = 0 \ \alpha : \text{increasing factor}$$
$$\frac{d\lambda(t)}{dt} = -\frac{\lambda(t)}{\beta}, \text{if } \ q(t-\tau_b) > 0 \ \ \beta : \text{decreasing factor} \tag{3}$$

Also, the queue size variation at the satellite can be defined as follows due to the arrival rate variation:

$$\frac{dq(t)}{dt} = 0, \quad \text{if } \ \lambda(t-\tau_f) < \mu \ \text{ and } \ \frac{dq(t)}{dt} = \lambda(t-\tau_f) - \mu, \quad \text{if } \ \lambda(t-\tau_f) > \mu \tag{4}$$

If we solve the above two equations together then we can get the maximum queue size at the satellite. The relationship between the queue and the arrival rate is shown in Fig. 1. First let us consider the time duration between t_1 and t_4 in order to find the time function of the queue size. In this period, if we assume the rate variation as exponential, $\lambda(t)=Ae^{\alpha t}$, then we can find the rate as follows:

$$\lambda(t)=(\mu+\alpha\tau)e^{-(t-t1)/\beta} \tag{5}$$

and in the increase duration of the rate, it can be described as: $\lambda(t)=a(t-t_4)+\lambda_{min}$ (6)

The queue function can be described as: $q(t) = \int_{t_0}^{t} \{\lambda(t-\tau_f) - \mu\} dt$ (7)

In order to calculate the time when the queue size reaches its maximum, we differentiate both sides of (7) to find the following equation as:

$$\frac{dq(t)}{dt} = \lambda(t-\tau_f) - \mu = 0, \quad \lambda(t-\tau_f) = \mu \tag{8}$$

According to (8), the queue has its maximum size at $t = t_0 + \tau_f$ and $t = t_2 + \tau_f$. But the real maximum value can be obtained at the latter time according to Fig. 1. To find the value of t_2, we can use the equation $\lambda(t_2)=\mu$.

$$\lambda(t_2) = (\mu+\alpha\tau)e^{-\frac{(t_2-t_1)}{\beta}} = \mu \tag{9}$$

which results in: $t_2 = t_0 + \tau - \beta \ln(\mu / \mu + \alpha\mu)$ (10)

Therefore, the maximum value of the queue is calculated as:

$$q_{\max} = \int_{t_0}^{t_2+\tau_f}\{\lambda(t-\tau_f)-\mu\}dt = \int_{t_0}^{t_0+\tau_f}\{\lambda(t-\tau_f)-\mu\}dt + \int_{t_0+\tau_f}^{t_1+\tau_f}\{\lambda(t-\tau_f)-\mu\}dt + \int_{t_1+\tau_f}^{t_2+\tau_f}\{\lambda(t-\tau_f)-\mu\}dt$$
$$= \int_{t_0+\tau_f}^{t_1+\tau_f}\{\lambda(t-\tau_f)-\mu\}dt + \int_{t_1+\tau_f}^{t_2+\tau_f}\{\lambda(t-\tau_f)-\mu\}dt$$

where, $$\int_{t_0+\tau_f}^{t_1+\tau_f}\{\lambda(t-\tau_f)-\mu\}dt = \int_{t_0}^{t_1}\{ay-\mu\}dy = \frac{\alpha\tau^2}{2} \quad (11)$$

$$\int_{t_1+\tau_f}^{t_2+\tau_f}\{\lambda(t-\tau_f)-\mu\}dt = \int_{t_1}^{t_2}\{(\mu+\alpha\tau)e^{-\frac{(y-t_1)}{\beta}}-\mu\}dy = -\beta(\mu+\alpha\tau)\{e^{-\frac{(y-t_1)}{\beta}}-1\}-\mu(t_2-t_1) = \alpha\beta\tau+\mu\beta\ln(\frac{\mu}{\mu+\alpha\tau})$$

Thus, $$q_{\max} = \frac{\alpha\tau^2}{2}+\alpha\beta\tau+\mu\beta\ln(\frac{\mu}{\mu+\alpha\tau}) \quad (12)$$

Let us now find the time period of the window-based flow control in the above situation. The time period for this case is: $T=\mu/\alpha+2\tau+t$ (13)

In (13) t is the time when the queue is empty. That means if we integrate the difference between arrival rate and service rate during that period we can find the maximum queue size. This is described by following set of equations:

$$\int_0^t\{(\alpha\tau+\mu)e^{-\frac{x}{\beta}}-\mu\}dx = -\frac{\alpha\tau^2}{2}, -\beta(\alpha\tau+\mu)e^{-\frac{t}{\beta}}-\mu t+\beta(\alpha\tau+\mu) = -\frac{\alpha\tau^2}{2}$$
$$e^{-\frac{t}{\beta}} = -\frac{2\alpha\tau^2+2\beta(\alpha\tau+\mu)}{2\beta(\alpha\tau+\mu)}-\frac{\mu}{\beta(\alpha\tau+\mu)}t \quad (14)$$

The approximate value of t in (14) can be obtained through the *Taylor Series.*

$$t = (e^{-\frac{A}{\beta}}(1+\frac{A}{\beta})-\frac{\mu}{\beta(\alpha\tau+\mu)}A)/(\frac{1}{\beta}e^{-\frac{A}{\beta}}-\frac{\mu}{\beta(\alpha\tau+\mu)}), \quad \text{where } A=\frac{2\alpha\tau^2+2\beta(\alpha\tau+\mu)}{2\mu}$$

Finally, the average throughput can be calculated as (15).

$$Average\ Throughput = \frac{(\mu/\alpha+\tau)(\mu+\alpha\tau)}{2}+\int_0^{t+\tau}\{(\alpha\tau+\mu)e^{-\frac{x}{\beta}}-\mu\}dx/T \quad (15)$$

3.2 Proportional Rate Control Algorithm (PRCA)

As shown in Fig. 2, PRCA is the method that reduces the data rate from its maximum value, λ_{max}. When the release of the congestion is issued with EFCI=0 by a RM cell, data rate increases to its maximum value again. Here we find the queue size and the data rate variations. Data rate variation of PRCA can be described as:

$$\lambda(t)=\lambda_{\max},\ \text{if } q(t-\tau_b)=0 \ \text{ and } \ \lambda(t)=\lambda_{\max},\ \frac{d\lambda(t)}{dt}=-\frac{\lambda(t)}{\beta},\quad \text{if } q(t-\tau_b)>0 \quad (16)$$

and the queue size variation is:

$$\frac{dq(t)}{dt}=0 \quad \text{if } \lambda(t-\tau_f)<\mu \ \text{ and } \ \frac{dq(t)}{dt}=\lambda(t-\tau_f)-\mu \quad \text{if } \lambda(t-\tau_f)>\mu \quad (17)$$

$\lambda(t)$ (in the rate decreasing area) and the queue size can be found as:

$$\lambda(t)=\lambda_{\max}e^{-t/\beta} \quad (18) \qquad q(t)=\int_0^t\{\lambda(k-\tau_f)-\mu\}dk \quad (19)$$

From the above equations, we need to find the time t when the queue has its maximum value. To do this we need to differentiate from both sides of (19) as:

$$\frac{dq(t)}{dt}=\lambda(t-\tau_f)-\mu=0,\ \lambda(t-\tau_f)=\mu \quad (20)$$

Thus, we know that the queue has its maximum value at $t = t_0 + \tau_f$ and therefore the maximum size of the queue can be obtained as:

$$q_{\max} = \int_{\tau_f}^{t_0+\tau_f} \{\lambda(t-\tau_f)-\mu\}dt = \int_0^{t_0} \{\lambda(x)-\mu\}dx + \int_{\tau_f}^{t_0+\tau_f} \{\lambda(x)-\mu\}dx \qquad (21)$$

However, the data rate function, $\lambda(t-\tau_f)$, is not defined during the period between 0 and τ_f and thus we leave the expression for q_{max} as:

$$q_{\max} = \int_{\tau_f}^{t_0+\tau_f} \{\lambda(t-\tau_f)-\mu\}dt = \int_0^{t_0} \{\lambda(x)-\mu\}dx \qquad (22)$$

In order to find the maximum value of the queue size, we need to know t_0, the time t when $\lambda(t)= \mu$. It can be expressed as:

$$\lambda(t_0) = \mu, \qquad \lambda_{\max} e^{-\frac{t_0}{\beta}} = \mu, \qquad t_0 = -\beta \ln \frac{\mu}{\lambda_{\max}} \qquad (23)$$

$$q_{\max} = \int_0^{-\beta\ln\frac{\mu}{\lambda_{\max}}} \{\lambda_{\max} e^{-\frac{x}{\beta}} - \mu\}dx = -\beta\mu + \beta\lambda_{\max} + \beta\mu \ln \frac{\mu}{\lambda_{\max}} \qquad (24)$$

Let's now find the average throughput over the time period. The time period can be calculated as: $T=\tau+t$ (25)

In (25), t is the time when the queue is empty after τ_f. t can be obtained like this.

$$\int_0^t \{\lambda_{\max} e^{-\frac{x}{\beta}} - \mu\}dx = 0, \qquad e^{-\frac{x}{\beta}} + \frac{\mu}{\beta\lambda_{\max}} t - 1 = 0, \qquad e^{-\frac{x}{\beta}} = 1 - \frac{\mu}{\beta\lambda_{\max}} t \qquad (26)$$

An approximate value for t in (26) can be obtained through the *Taylor Series*.

$$t = (e^{-\frac{\lambda_{\max}}{\beta}} (1+\frac{\lambda_{\max}}{\mu}) - 1) / (\frac{1}{\beta} e^{-\frac{\lambda_{\max}}{\beta}} - \frac{\mu}{\beta\lambda_{\max}}) \qquad (27)$$

The average throughput over the periodic time is calculated as (28).

$$Average\ \ Throughput = \int_0^{t+\tau} \lambda_{\max} e^{-\frac{x}{\beta}} dx / T \qquad (28)$$

3.3 The Proposed SISD Scheme

The proposed rate control scheme does not exchange any information with satellite. This regulates the data rate based on the information that is broadcasted by the satellite periodically through some control satellite channels and piggybacking way.

In the following, the maximum queue size and the average throughput of the SISD scheme are calculated. As shown in Fig. 3, it is assumed that for the SISD scheme $\lambda(t)= \mu=n\lambda_i$, where λ_i is the step size of the rate increase and decrease.

As it can be seen from Fig. 3, even if the data rate is μ, data rate will be increased because the queue level is still under the threshold. When the data rate reaches $\mu + \lambda_i$, the queue size starts to approach zero after the time τ_f, which will affect the data rate after the time τ_b. This information, which is broadcasted by the satellite, causes earth stations to reduce their data rate and data reduction will affect the queue size on the satellite.

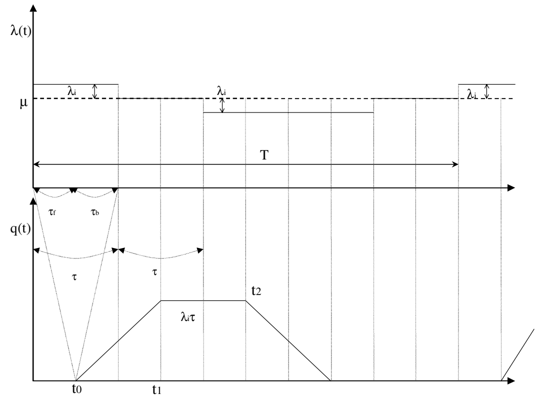


Fig. 3. Data rate and queue size variations in SISD scheme

The time function of the queue size variation can be expressed by:

$$\frac{dq(t)}{dt} = 0, \quad \text{if } \lambda(t-\tau_f) < \mu \text{ and } \frac{dq(t)}{dt} = \lambda(t-\tau_f) - \mu, \quad \text{if } \lambda(t-\tau_f) > \mu \tag{29}$$

Therefore, as shown in Fig. 3, the maximum size of the queue can be given by (30).

$$q_{\max} = \int_0^{\tau} \{\mu + \lambda_i - \mu\} dt = \lambda_i \tau \tag{30}$$

Note that in SISD, even if the queue size is above the threshold but decreasing, the data rate is maintained steady as it is shown in Fig. 3. Because of the steady state nature of the data rate in SISD scheme, the time period of the data rate variation is 5τ and the amount of data transmitted during this period is:

$$\text{Transmitted Data} = (5\mu - \lambda_i)\tau \tag{31}$$

Thus, the average throughput for the SISD scheme over the time period will be,

$$\text{Average Throughput} = (5\mu - \lambda_i)\tau / T \tag{32}$$

4 Analysis Results

In this section, numeric performance comparison of the three schemes analyzed in Section 3 is provided. For the performance comparison, following parameters are assumed: *μ=1000 packets/s, increasing factor α=500packets/s, decreasing factor β=0.875 sec, λ_{max} in PRCA=1500packets/s and λ_i in SISD=100packets/s.*

Figure 4 and 5 show the maximum queue size and the average throughput for the three congestion control schemes as a function of RTT, respectively. From Fig. 4, we can see that the maximum queue size in PRCA is fixed regardless of the change in RTT but in window control and SISD schemes it is growing as RTT increases. However, the maximum queue size in these two schemes remains much smaller than PRCA. The proposed SISD scheme, however, shows the best performance among three congestion control schemes for the average throughput, as shown in Fig. 5. PRCA experiences significant decrease in average throughput as RTT increases whereas the window control throughput increases slightly.

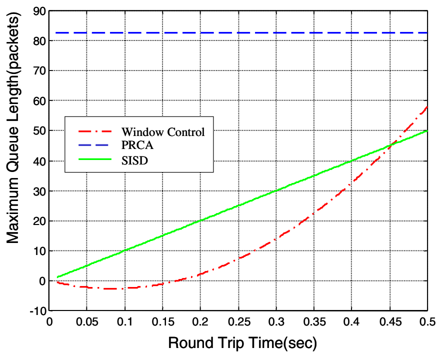


Fig. 4. Maximum queue size variations

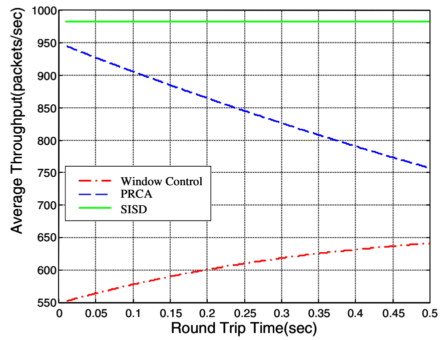


Fig. 5. Average throughput variations

According to these results, the proposed SISD scheme can transmit more data than other schemes with a relatively small size queue. This means that SISD can regulate the data rate efficiently to prevent congestions in satellite networks when the long propagation delay is considered to be a restrictive factor in data rate control. Thus, SISD is more suitable in providing Internet services in satellite networks compared to other methods.

Performance of each scheme, of course, may vary with different set of parameters. For example, if α in window based flow control is increased, the rate will reach easily to μ, which increases the throughput but causes the queue size to increase significantly. In PRCA, increasing λ_{max} should improve the throughput but make the queue explode. In the proposed SISD, increasing λ_i would affect the queue size and throughput although data rate can reach μ easily. Thus the queue size could be increased and the throughput should be degraded. But in SISD, the maximum queue size increases linearly and the throughput degrades by λ_i /5. Thus the increase in maximum queue size and degradation in throughput of SISD are very small compared with other schemes. Therefore the proposed SISD can be considered to be more stable and efficient in satellite networks with long propagation delay.

5 Conclusions

This paper has introduced a new congestion control scheme using periodic buffer information from the satellite in the multiple-spot beam environment. In the proposed scheme, the satellite simply broadcasts information of its own buffer status to all earth stations periodically in the beam areas and the earth stations regulate the data transmission rate for the proper data services accordingly. In addition, the mathematical analyses for comparing performance of the proposed scheme with the existing congestion control schemes have been performed. Through those results, it was found that the proposed scheme is more suitable in resolving congestion in the multiple-spot beam satellite networks because it has a better throughput under a reasonable satellite buffer size. As a future work, we need to consider the fairness of bandwidth sharing in satellite networks when various types of traffic have been involved in the satellite communications.

Acknowledgement

This research was financially supported by Hansung University in the year of 2004.

References

[1] A. Jamalipour and T. Tung, "The role of satellites in global IT: Trends and implications", IEEE Personal Communications Magazine, June 2001, vol. 8, no. 3, pp. 5-11.
[2] I. F. Akyildiz and S. Jeong, "Satellite ATM Networks: A Survey", IEEE Communications Magazine, July 1997, vol. 35, no. 7, pp. 30-43.
[3] P. Chitre and F. Yegenoglu, "Next-Generation Satellite Networks: Architectures and Implementations", IEEE Communications Magazine, March 1999, vol. 37, no. 3, pp. 30-36.
[4] I. Mertzanis, G. Sfikas, R. Tafazolli, and B. G. Evans, "Protocol Architectures for Satellite ATM Broadband Networks", IEEE Communications Magazine, March 1999, vol. 37, no. 3, pp. 46-54.
[5] R. Goyal et al., "Traffic Management for TCP/IP Over Satellite ATM Networks," IEEE Commun. Mag., vol. 37, no. 3, Mar. 1999, pp. 56–61.
[6] T. Inzerilli and A. Pietrabissa, "Satellite QoS Architecture in the SATIP6 Project", Proceedings of IST Mobile&Wireless Communications Summit 2003, June 2003, pp. 232-236.
[7] R. Jain, "Tutorial Paper on ATM Congestion Control," ATM Forum/95-0177.
[8] B. G. Evans, et al, "Future Multimedia Communications via Satellite," International Journals of Satellite Communications, vol. 16, 1996, pp. 467-474.
[9] M. Schwartz, Broadband Integrated Networks, Prentice Hall, 1996.
[10] T. T. Ha, Digital Satellite Communications, McGraw-Hill, 2nd Edition, 1990.

Real-Time Network Traffic Prediction Based on a Multiscale Decomposition

Guoqiang Mao

The University of Sydney, NSW 2006, Australia
guoqiang@ee.usyd.edu.au

Abstract. The presence of the complex scaling behavior in network traffic makes accurate forecasting of the traffic a challenging task. In this paper we propose a multiscale decomposition approach to real time traffic prediction. The raw traffic data is first decomposed into multiple timescales using the *à trous* Haar wavelet transform. The wavelet coefficients and the scaling coefficients at each scale are predicted independently using the ARIMA model. The predicted wavelet coefficients and scaling coefficient are then combined to give the predicted traffic. This multiscale decomposition approach can better capture the correlation structure of traffic caused by different network mechanisms, which may not be obvious when examining the raw data directly. The proposed prediction algorithm is applied to real network traffic. It is shown that the proposed algorithm generally outperforms traffic prediction using neural network approach and gives more accurate result. The complexity of the prediction algorithm is also significantly lower than that using neural network.

1 Introduction

Accurate forecasting of the traffic is important in the planning, design, control and management of networks. Traffic prediction at different timescales has been used in various fields of networks, such as long-term traffic prediction for network planning, design and routing; and short-term traffic prediction for dynamic bandwidth allocation, and predictive and reactive traffic and congestion control.

Some algorithms have been proposed in the literature for real-time traffic prediction, which include traffic prediction using the FARIMA (fractional autoregressive integrated moving average) model [1], neural network approach [2], [3] and methods based on α-stable models [4], [5], etc. Traffic prediction using the FARIMA model relies on accurate estimation of the Hurst parameter, which is a measure of the self-similarity of the traffic. Despite a number of estimators reported in the literature, accurate estimation of the Hurst parameter remains a difficult problem even in off-line conditions. The presence of non-stationarity and complex scaling behavior in network traffic makes the situation even worse. Therefore real applications of traffic prediction based on the FARIMA model are not optimistic. Neural network approach can be quite complicated to implement in reality. The accuracy and applicability of neural network approach to

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 492–499, 2005.

traffic prediction is limited [3]. Finally, α-stable model is based on a generalized central limit theorem and its application is limited by that. It might achieve a good performance in heavy traffic or when there is a high level of traffic aggregations. However when traffic conditions deviate from that, the performance may be poor. Moreover, α-stable model is a parsimonious model, which may not be able to capture the complex scaling behavior of the traffic. In this paper we propose a traffic prediction algorithm based on a multiscale decomposition approach. Using the $\grave{a} - trous$ Haar wavelet transform, the traffic is decomposed into components at multiple timescales. Traffic component at each timescale is predicted independently with an ARIMA (autoregressive integrated moving average) model. Then they are combined to form the predicted traffic.

The rest of the paper is organized as follows: in section 2, we shall introduce the use of the *à trous* Haar wavelet transform in decomposing the traffic into different timescales; in section 3 the prediction algorithm will be introduced; some simulation results using real traffic trace are given in section 4 and finally some conclusions are given in section 5.

2 Multiscale Traffic Decomposition

Wavelet tools have been widely used in the area of traffic analysis. Discrete wavelet transform (DWT) consists of the collection of coefficients:

$$c_J(k) =< X, \varphi_{Jk}(t) >, \quad d_j(k) =< X, \psi_{jk}(t) >, \;\; j,\; k \in Z, \tag{1}$$

where $< *, * >$ denotes inner product, $\{d_j(k)\}$ are the wavelet coefficients and $\{c_J(k)\}$ are the scaling coefficients. The analysis functions $\psi_{jk}(t)$ are constructed from a reference pattern $\psi(t)$ called the mother-wavelet by a time-shift operation and a dilation operation: $\psi_{jk}(t) = 2^{-j/2}\psi(2^{-j}t - k)$. The mother wavelet is a band-pass or oscillating function, hence the name "wavelet". Function $\varphi_{Jk}(t)$ is a time shifted function of the mother scaling function $\varphi_J(t)$: $\varphi_{Jk}(t) = \varphi_J(t-k)$. $\varphi_J(t)$ is a low-pass function which can separate large timescale (low frequency) component of the signal. Thus wavelet transform decomposes a signal into a large timescale approximation (coarse approximation) and a collection of details at different smaller timescales (finer details). This allows us to zoom into any timescale that we are interested in and use the coefficients of a wavelet transform to directly study the scale dependent properties of the data. Moreover the analysis of each scale is largely decoupled from that at other scales [6]. Refer to [7], [8] for details of wavelet theory.

In addition to the characteristics of applications generating the traffic, traffic variations at different timescales are caused by different network mechanisms. Traffic variations at small timescales (i.e. in the order of ms or smaller timescale) are caused by buffers and scheduling algorithms etc. Traffic variations at larger timescales (i.e. in the order of 100ms) are caused by traffic and congestion control protocols, e.g. TCP protocols. Traffic variations at even larger timescales are caused by routing changes, daily and weekly cyclic shift in user populations.

Finally long-term traffic changes are caused by long-term increases in user population as well as increases in bandwidth requirement of users due to the emergence of new network applications. This fact motivates us to decompose traffic into different timescales and predict traffic independently at each timescale. The proposed multiscale decomposition approach to traffic prediction allows us to explore the correlation structure of network traffic at different timescales caused by different network mechanisms, which may not be easy to investigate when examining the raw data directly.

The roles of the mother scaling and wavelet functions $\varphi(t)$ and $\psi(t)$ can also be represented by a low-pass filter h and a high pass filter g. Consequently, the multiresolution analysis and synthesis of a signal $x(t)$ can be implemented efficiently as a filter bank [7]. The approximation at scale j, $c_j(k)$, is passed through the low-pass filter h and the high pass filter g to produce the approximation $c_{j+1}(k)$ and the detail $d_{j+1}(k)$ at scale $j+1$. At each stage, the number of coefficients at scale $j+1$ is decimated into half of that at scale j, due to downsampling. This decimation reduces the number of data points to be processed at coarser time scales and removes the redundancy information in the wavelet coefficients and the scaling coefficients at the coarser time scales. Decimation allows us to represent a signal X by its wavelet and scaling coefficients whose total length is the same as the original signal. However decimation has the undesirable effect that we cannot relate information at a given time point at the different scales in a simple manner. Moreover, while it is desirable in some applications (e.g. image compression) to remove the redundancy information, in time series prediction the redundancy information can be used to improve the accuracy of the prediction.

In this paper, we use a redundant wavelet transform, i.e. the $\grave{a}-trous$ wavelet transform, to decompose the signal [9]. Using the redundant information from the original signal, the $\grave{a}-trous$ wavelet transform produces smoother approximations by filling the "gap" caused by decimation. Using the $\grave{a}-trous$ wavelet transform, the scaling coefficients and the wavelet coefficients of $x(t)$ at different scales can be obtained as:

$$c_0(t) = x(t) \tag{2}$$

$$c_j(t) = \sum_{l=-\infty}^{\infty} h(l) c_{j-1}(t + 2^{j-1} l). \tag{3}$$

where $1 \leq j \leq J$, and h is a low-pass filter with compact support. The detail of $x(t)$ at scale j is given by:

$$d_j(t) = c_{j-1}(t) - c_j(t). \tag{4}$$

The set $d_1, d_2, ..., d_J, c_J$ represents the wavelet transform of the signal up to the scale J, and the signal can be expressed as a sum of the wavelet coefficients and the scaling coefficients: $x(t) = c_J(t) + \sum_{j=1}^{J} d_j(t)$.

Many wavelet filters are available, such as Daubechies' family of wavelet filters, $B3$ spline filter, etc. Here we choose Haar wavelet filter to implement the $\grave{a}-trous$ wavelet transform. A major reason for choosing the Haar wavelet filter is the calculation of the scaling coefficients and wavelet coefficients at time t uses information before time t only. This is a very desirable feature in time series prediction. The Haar wavelet uses a simple filter $h = (1/2, 1/2)$. The scaling coefficients at higher scale can be easily obtained from the scaling coefficients at lower scale:

$$c_{j+1,t} = \frac{1}{2}(c_{j,t-2^j} + c_{j,t}). \tag{5}$$

The wavelet coefficients can then be obtained from Equation (4).

3 The Prediction Algorithm

In this section, we use the aforementioned $\grave{a}-trous$ Haar wavelet decomposition for traffic prediction. Instead of predicting the original signal directly, we predict the wavelet coefficients and the scaling coefficients independently at each scale and use the wavelet coefficients and the scaling coefficients to construct the prediction of the original signal.

Fig. 1 shows the architecture of the prediction algorithm. Coefficient prediction can be represented mathematically as

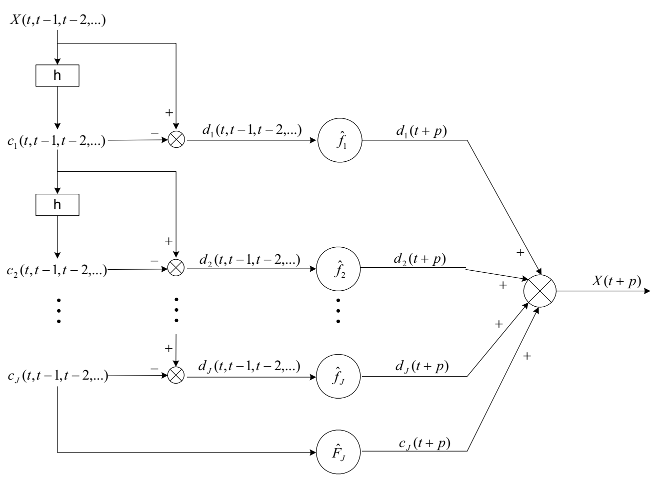


Fig. 1. Architecture of the prediction algorithm

$$\widehat{c}_J(t+p) = \widehat{F}_J(c_J(t), c_J(t-1), ..., c_J(t-m)), \tag{6}$$

$$\widehat{d}_j(t+p) = \widehat{f}_j(d_j(t), d_j(t-1), ..., d_j(t-n_j)), \tag{7}$$

where m and n_j is the number of coefficients used for prediction and p is the prediction depth. In this paper, we only use one-step prediction, i.e. p=1. Multistep prediction can be achieved by using the predicted value as the real value or by aggregating the traffic into larger time interval.

$ARIMA(p, d, q)$ model is used for predicting the wavelet and the scaling coefficients at each scale. An $ARMA(p, q)$ (autoregressive moving average) model can be represented as:

$$\phi(B)X_t = \theta(B)Z_t, \tag{8}$$

where Z_t is a Gaussian distributed random variable with zero mean and variance σ^2, , i.e. $Z_t \sim WN(0, \sigma^2)$. The polynomials ϕ and θ are polynomials of degree p and q respectively and they have no common factors [10]. B is the backward shift operator: $B^j X_t = X_{t-j}$. ARMA model assumes the time series are stationary. If the time series exhibits variations that violate the stationarity assumption, differencing operation can be used to remove the non-stationary trend in the time series. We define the lag-1 difference operator ∇ by:

$$\nabla X_t = X_t - X_{t-1} = (1 - B)X_t. \tag{9}$$

An ARIMA(p,d,q) model is an ARMA(p,q) model that has been differenced d times. Therefore it can be represented as:

$$\phi(B)(1 - B)^d X_t = \theta(B)Z_t. \tag{10}$$

Box-Jenkins forecasting methodology is used to establish the ARIMA(p,d,q) model for prediction at each scale [10].

4 Simulation

In this section, we apply the proposed model to the real network traffic for prediction. The traffic traces used were collected by WAND research group at the University of Waikato Computer Science Department. It is the LAN traffic at the University of Auckland on campus level. The traffic traces were collected between 6am and 12pm from June 9, 2001 to June 13, 2001 on a 100Mbps Ethernet link. IP headers in the traffic trace are GPS synchronized and have an accuracy of $1\mu s$. More information on the traffic trace and the measurement infrastructure can be found on their webpage: http://atm.cs.waikato.ac.nz/wand/wits/auck/6/. Five traffic traces are used. Table 1 shows information of the traffic traces.

We use the traffic rate measured in the previous 1s time intervals to predict the traffic rate in the next second. Prediction over longer or shorter time intervals can be achieved by reducing the length of the time interval or by multistep

Table 1. Trace traces used in the simulation

Trace ID	File name	Measurement time	Duration
1	20010609-060000-e0.gz	Saturday June 9, 2001	6am-12pm
2	20010610-060000-e0.gz	Sunday June 10, 2001	6am-12pm
3	20010611-060000-e0.gz	Monday June 11, 2001	6am-12pm
4	20010612-060000-e0.gz	Tuesday June 12, 2001	6am-12pm
5	20010613-060000-e0.gz	Wednesday June 13, 2001	6am-9am

Table 2. Model parameter of the prediction model

Scale	Model name	Parameters ϕ	Parameters θ	Noise σ^2
Scale 1 Wavelet	ARIMA(1,0,4)	$\phi_1 = 0.8842$	$\theta_1 = 1.311, \theta_2 = -0.2185,$ $\theta_3 = 0, \theta_4 = -0.1008$	2.147×10^9
Scale 2 Wavelet	ARIMA(4,0,4)	$\phi_1 = 1.443,$ $\phi_2 = -0.4782,$ $\phi_3 = 0.04215,$ $\phi_4 = -0.02682$	$\theta_1 = -0.04322, \theta_2 = 1.768$ $\theta_3 = 0.04953, \theta_4 = -0.7767$	5.847×10^8
Scale 3 Wavelet	ARIMA(4,0,8)	$\phi_1 = 1.384,$ $\phi_2 = -0.435,$ $\phi_3 = 0.02306,$ $\phi_4 = -0.004911$	$\theta_1 = -0.1833, \theta_2 = -0.1531,$ $\theta_3 = -0.1824, \theta_4 = 1.751,$ $\theta_5 = 0.1789, \theta_6 = 0.1508,$ $\theta_7 = 0.1782, \theta_8 = -0.7583$	1.422×10^8
Scale 3 Scaling	ARIMA(2,1,8)	$\phi_1 = 0.508,$ $\phi_2 = 0.02201$	$\theta_1 = -0.07853, \theta_2 = -0.08036$ $\theta_3 = -0.07985, \theta_4 = -0.08014,$ $\theta_5 = -0.07935, \theta_6 = -0.08083,$ $\theta_7 = -0.0796, \theta_8 = 0.9188$	1.348×10^8

prediction. To validate the performance of the proposed prediction model, one of the traffic traces (i.e. trace 4) was picked randomly to establish the prediction model and the prediction model is then applied to other traffic traces for prediction.

Table 2 shows the model parameters of the ARIMA(p,d,q) model for wavelet and scaling coefficients at each scale. Three scales are chosen. The choice on the number of scales used for prediction is made based on the tradeoff between model complexity and accuracy. Further increase in the number of scales significantly increases the complexity of the algorithm but there is only a marginal increase in accuracy. As shown in the table, most noise in the model comes from wavelet coefficients at scale 1. In comparison with wavelet coefficients and scaling coefficients at other scales, wavelet coefficients at scale 1 has very weak autocorrelations and a white noise like power spectral density. It is the wavelet coefficients at scale 1 that limit the overall performance that can be achieved by the prediction algorithm.

The ARIMA models developed from trace 4 are then applied to the other traffic traces to establish the performance of the prediction algorithm. To measure the performance of the prediction algorithm, two metrics are used. One is the normalized mean square error (NMSE): $NMSE = \frac{\frac{1}{N}\sum_{n=1}^{N}(X(n)-\hat{X}(n))^2}{var(X(n))}$, where $\hat{X}(n)$ is the predicted value of $X(n)$ and $var(X(n))$ denotes the variance of $X(n)$. The other is the mean absolute relative error (MARE), which is defined as follows: $MARE = \frac{1}{N}\sum_{n=1}^{N}\left|\frac{X(n)-\hat{X}(n)}{X(n)}\right|$. Since the relative error may be unduely affected by vary small values of $X(n)$, to make meaningful observations, we only count the MARE of $X(n)$ whose value is not small than the average value of $X(n)$. Table 3 shows the performance of the prediction algorithm. For comparison purpose, the performance of traffic prediction using neural network approach is also shown in the table. A number of neural network models with

Table 3. Performance of the prediction model

Trace ID	Multiscale ARIMA		Neural network	
	NMSE	MARE	NMSE	MARE
1	0.1319	0.1633	0.1603	0.1667
2	0.2296	0.2165	0.3168	0.2053
3	0.1507	0.1403	0.1565	0.1493
4	0.1592	0.1313	0.1622	0.1386
5	0.21972	0.1731	0.2258	0.1823

different number of input nodes, hidden nodes and transfer functions are evaluated, including those reported in [3], [11]. It is found that the 32-16-4-1 network architecture used in [11] gives the best performance. Hyperbolic tangent sigmoid transfer function is used in the hidden layer and linear transfer function is used in the output layer. The performance of the 32-16-4-1 neural network model is shown in Table 3 to represent the prediction performance using neural networks. To achieve a fair comparison, the same trace used for building ARIMA(p,d,q) models is used to train the neural network. The very large data size in the training trace ensures the convergence of the neural network parameters, which is also confirmed by a visual inspection of the error signal.

As shown in Table 3, the ARIMA model with multiscale decomposition (referred to as multiscale ARIMA model for simplicity) gives better performance than the neural network approach in most cases except for trace 2, where the MARE metric of neural network approach is slightly better than that achieved by multiscale ARIMA approach. However, the NMSE metric of the neural network approach is much worse than that of multiscale ARIMA approach for trace 2. Therefore the exception on trace 2 cannot be used as an evidence that neural network performs better for trace 2. As such, it can be concluded that ARIMA model with multiscale decomposition generally achieves better performance than the neural network prediction. Moreover, only three scales are employed in the proposed prediction algorithm, which requires a memory length (here memory length refers to the number of past raw data samples required for prediction) of about 8. In comparison, neural network requires a memory length of 32. The computation using multiscale ARIMA model is also much easier than that using neural network.

5 Conclusion

In this paper we proposed a real-time network traffic prediction algorithm based on a multiscale decomposition. The raw traffic data is first decomposed into different timescales using the *à trous* Haar wavelet transform. The prediction of the wavelet coefficients and the scaling coefficients are performed independently at each timescale using the ARIMA model. The predicted wavelet coefficients and scaling coefficient are then combined to give the predicted traffic value. As traffic variations at different timescales are caused by different network mechanisms, the proposed multiscale decomposition approach to traffic prediction can better capture the correlation structure of traffic caused by different network mechanisms, which may not be obvious when examining the raw data directly.

The prediction algorithm was applied to real network traffic. The performance of the proposed prediction algorithm was compared with that using neural network. It was shown that the proposed algorithm generally outperforms traffic prediction algorithm using neural network approach and gives more accurate prediction. The complexity of the prediction algorithm is also significantly lower than that using neural network.

References

1. Shu, Y., Jin, Z., Zhang, L., Wang, L.: Traffic prediction using farima models. In: IEEE International Conference on Communications. Volume 2. (1999) 891–895
2. Liang, Y.: Real-time vbr video traffic prediction for dynamic bandwidth allocation. IEEE Transactions on Systems, Man and Cybernetics, Part C **34** (2004) 32–47
3. Hall, J., Mars, P.: Limitations of artificial neural networks for traffic prediction in broadband networks. IEE Proceedings Communications **147** (2000) 114–118
4. Lopez-Guerrero, M., Gallardo, J., Makrakis, D., Orozco-Barbosa, L.: Optimizing linear prediction of network traffic using modeling based on fractional stable noise. In: 2001 International Conferences on Info-tech and Info-net. Volume 2. (2001) 587–592 vol.2
5. Karasaridis, A., Hatzinakos, D.: Network heavy traffic modeling using *alpha*-stable self-similar processes. IEEE Transactions on Communications **49** (2001) 1203–1214
6. Abry, P., Flandrin, P., Taqqu, M.S., Veitch, D.: Wavelets for the analysis, estimation, and synthesis of scaling data. In Park, K., Willinger, W., eds.: Self-Similar Network Traffic and Performance Evaluation. John Wiley and Sons, Inc. (2000) 39–88
7. Strang, G., Nguyen, T.: Wavelets and Filter Banks. Wellesley-Cambridge Press (1996)
8. Daubechies, I.: Ten Lectures on Wavelets. Capital City Press, Montpelier, Vermont (1992)
9. Shensa, M.: The discrete wavelet transform: wedding the a trous and mallat algorithms. IEEE Transactions on Signal Processing **40** (1992) 2464–2482
10. Bowerman, B.L., O'Connell, R.T.: Time Series Forecasting - Unified Concepts and Computer Implementation. 2 edn. PWS publishers (1987)
11. Liang, Y., Page, E.: Multiresolution learning paradigm and signal prediction. IEEE Transactions on Signal Processing **45** (1997) 2858–2864

Provisioning VPN over Shared Network Infrastructure

Quanshi Xia

IC-Parc, Imperial College London, London SW7 2AZ, UK
q.xia@imperial.ac.uk

Abstract. This paper addresses the provisioning VPN services over shared network infrastructure with QoS guarantees (bandwidth and propagation delay), attempt to minimise the bandwidth reservation on the network. We present general MILP formulations based on the hose model and the pipe model. The *reformulated* MILP can be solved by standard MILP packages efficiently and scalably. The benchmark results show that the over-provisioning factor about 5 in bandwidth reservation for VPN can be reduced by the hose model, compared with the pipe model.

1 Introduction

Virtual private network (VPN) establishes connectivity between a set of geographically dispersed endpoints over a shared network infrastructure. Providing VPN service is playing an important role in the revenue stream of Internet service provider (ISP). In order to be able to support a wide variety of customer requirements, network operators need a flexible and bandwidth efficient model, comparable to a private dedicated network established with *leased* lines.

Traditionally, provisioning VPN, *i.e.* setting up the path between every customer pair within VPN, is based on the *pipe* model, in which the traffic demand is specified for each customer pair, and the bandwidth is reserved for point-to-point connection tunnel. For the pipe model, a traffic matrix which describes the required bandwidth between each VPN endpoint pair must be known *a priori*. However, the communication pattern between the endpoints is difficult to predict [1]. It is almost impossible to estimate the exact traffic matrix required by the pipe model.

A different scheme for provisioning VPN, the *hose* model, was recently proposed [2]. The hose model specifies, instead of a complete traffic matrix, the total amount of traffic which a customer injects into the network and the total amount of traffic which it receives from the network. This VPN specification is in fact backed up by the service level agreement (SLA).

The bandwidth efficiency between the hose model and the pipe model for provisioning VPN was studied [3], which shows that the significant over-provisioning factor can be reduced by the hose model. Setting up the tunnel between every customer pair based on the hose model was initially investigated [4]. The tree structure is used to connect all VPN endpoints. An algorithm for computing

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 500–507, 2005.

optimal tree structure was presented, assuming that the capacity of the link is *infinite*. Thus, the provisioned tunnel may *violate* the limited bandwidth.

For the limited link capacity, the provisioning VPN under multi-path routing by the hose model has recently been addressed in [5]. Although multi-path routing has the advantage of reducing the bandwidth reservation, it is difficult for network operator to implement such multiple tunnels. Also the Quality of Service (QoS) like propagation delay cannot been guaranteed.

In this paper, we study the general optimisation problem for provisioning VPN services with QoS guarantees (*i.e.* bandwidth, propagation delay), attempt to minimise the bandwidth reservation for VPN on the network. The general mixed integer linear programming (MILP) formulations based on the *hose* model and the *pipe* model are presented. And the reformulated MILP can be solved by standard MILP packages efficiently and scalably. The comparison between the hose model and the pipe model is made in terms of the model size, the solving time and the bandwidth reservation. The benchmark results on a set of test cases show that the over-provisioning factor about 5 in bandwidth reservation can be reduced by the hose model, compared with the pipe model.

This paper is organised as follows. In section 2, we present general MILP formulations based on the pipe model and the hose model. The hose model formulation is then reformulated and relaxed. Section 3 gives the benchmark test results and comparisons. Section 4 concludes the paper.

2 Provisioning VPN over Shared Network Infrastructure

2.1 Problem Statement

We model the *underlying* network as a set of nodes $\mathbf{N}$ and a set of *directed* edges $\mathbf{E}$. Each edge $e(k,l) \in \mathbf{E}$ directly connects node $k \in \mathbf{N}$ to $l \in \mathbf{N}$. It is assumed that edge $e \in \mathbf{E}$ has a *limited* bandwidth capacity c_e and a propagation delay d_e. For each node $n \in \mathbf{N}$ there is a set of edges $\mathbf{I}(n) \subset \mathbf{E}$ entering n and a set of edges $\mathbf{O}(n) \subset \mathbf{E}$ leaving n.

Each VPN specification consists of: (1) A set of nodes $\mathbf{P} \subseteq \mathbf{N}$ corresponding to the VPN customer endpoints; (2) The ingress and egress bandwidths, respectively, B_i^{in} and B_i^{out} for each customer node $i \in \mathbf{P}$. The ingress bandwidth is the maximum amount of traffic to send to all the other VPN endpoints, while the egress bandwidth specifies the maximum amount of traffic can be received from all the other VPN endpoints; and (3) The QoS parameters such as the propagation delay D^{ij} for each VPN customer pair $(i, j \in \mathbf{P}, (i \neq j))$.

Let us first introduce some commonly used variables and constraints.

Routing variable $P_e^{ij} \in \{0,1\}$ *states whether edge* $e \in \mathbf{E}$ *is used for the path from customer node* $i \in \mathbf{P}$ *to* $j \in \mathbf{P}(i \neq j)$.

Utilisation variable $U_e \in [0,1]$ *is the link utilisation of edge* $e \in \mathbf{E}$.

Path constraint *states that for every customer pair* $(\forall i, j \in \mathbf{P}(i \neq j))$ *within VPN, there must be a continuous path from the origin* i *to the destination* j*:*

$$\forall n \in \mathbf{N}: \quad \sum_{e \in \mathbf{O}(n)} P_e^{ij} - \sum_{e \in \mathbf{I}(n)} P_e^{ij} = \begin{cases} 1 & n = i \\ -1 & n = j \\ 0 & \text{otherwise} \end{cases}$$

Delay constraint *constrains the propagation delay along the path from customer node* $i \in \mathbf{P}$ *to* $j \in \mathbf{P}(i \neq j)$*:*

$$\sum_{e \in \mathbf{E}} d_e P_e^{ij} \leq D^{ij}$$

2.2 Provisioning VPN on Pipe Model

For the pipe model, a traffic matrix $T = \{T_{ij}\}$ describes the required bandwidth between each VPN endpoint pair. Traffic between each pair of customer access points is carried through the customer pipes (point-to-point connections) with a given pre-allocated bandwidth according to T_{ij}. However, the communication pattern between endpoints is difficult to forecast. It is almost impossible to predict the exact traffic matrix required by the pipe model. Therefore, between each customer endpoint pair (i, j) the *maximum* (worst case) traffic is $min\{B_i^{in}, B_i^{out}\}$ which is used to approximate T_{ij}.

Capacity constraint (pipe model) *states that under pipe model, the bandwidth required on edge* $e \in \mathbf{E}$ *by the* worst case *traffic within the VPN cannot exceed its capacity.*

$$c_e U_e \geq \sum_{i \in \mathbf{P}} \sum_{j \in \mathbf{P}(i \neq j)} min\{B_i^{in}, B_j^{out}\} P_e^{ij}$$

Formulation 1 (minimise bandwidth reservation - pipe model) *Based on the pipe model, the provisioning VPN to minimise the total reserved bandwidth can be formulated as:*

$$\min_{\{P_e^{ij} \in \{0,1\}, U_e \in [0,1]\}} \quad \sum_{e \in \mathbf{E}} c_e U_e$$
$$st. \begin{cases} \forall n \in \mathbf{N}, \forall i, j \in \mathbf{P}(i \neq j): \sum_{e \in \mathbf{O}(n)} P_e^{ij} - \sum_{e \in \mathbf{I}(n)} P_e^{ij} = \begin{cases} 1 & n = i \\ -1 & n = j \\ 0 & \text{otherwise} \end{cases} \\ \forall i, j \in \mathbf{P}(i \neq j): \quad \sum_{e \in \mathbf{E}} d_e P_e^{ij} \leq D^{ij} \\ \forall e \in \mathbf{E}: \quad c_e U_e \geq \sum_{i \in \mathbf{P}} \sum_{j \in \mathbf{P}(i \neq j)} min\{B_i^{in}, B_j^{out}\} P_e^{ij} \end{cases} \tag{1}$$

This MILP model can be efficiently solved by standard MILP solvers.

2.3 Provisioning VPN on Hose Model

In the hose model, instead stating the traffic matrix $T = \{T_{ij}\}$, only the ingress bandwidth B_i^{in} and the egress bandwidth B_i^{out} are specified for each customer access point. The traffic to and from a customer endpoint is *arbitrarily* distributed to other VPN endpoints. Therefore, any *possible* traffic matrix $T_{ij} \geq 0$ is constrained by:

$$\begin{cases} \forall i \in \mathbf{P}: & \sum_{j \in \mathbf{P}(j \neq i)} T_{ij} \leq B_i^{in} \\ \forall j \in \mathbf{P}: & \sum_{i \in \mathbf{P}(i \neq j)} T_{ij} \leq B_j^{out} \end{cases} \tag{2}$$

However, on the edges carrying multiple traffic flows originating from a single ingress node (or destinate to a single egress node), only the minimum bandwidth of the ingress node (or egress node) is allocated.

Traffic distribution variable $F_e^{ij} \geq 0$ *is an arbitrary distribution on edge* $e \in \mathbf{E}$ *of traffic* T_{ij} *from customer node* $i \in \mathbf{P}$ *to* $j \in \mathbf{P}(i \neq j)$.

Capacity constraint (hose model) *states that under the hose model, the bandwidth required on edge* $e \in \mathbf{E}$ *by* worst case *traffic distribution within VPN cannot exceed its capacity.*

$$c_e U_e \geq \begin{cases} \max\limits_{F_e^{ij} \geq 0} \sum_{i \in \mathbf{P}} \sum_{j \in \mathbf{P}(i \neq j)} P_e^{ij} F_e^{ij} \\ st. \begin{cases} \forall i \in \mathbf{P}: & \sum_{j \in \mathbf{P}(j \neq i)} F_e^{ij} \leq B_i^{in} \\ \forall j \in \mathbf{P}: & \sum_{i \in \mathbf{P}(i \neq j)} F_e^{ij} \leq B_j^{out} \end{cases} \end{cases}$$

This constraint guarantees the sufficient bandwidth to accommodate the *worst case* traffic among VPN endpoints that satisfies the ingress and egress bandwidth bounds. Therefore, the reserved bandwidth is sufficient to support *every possible* traffic matrix T_{ij} that is consistent with the constraints (2).

Formulation 2 (minimise bandwidth reservation - hose model) *Based on the hose model, the provisioning VPN to minimise the total reserved bandwidth for VPN can be formulated as:*

$$\min_{\{P_e^{ij} \in \{0,1\}, U_e \in [0,1]\}} \sum_{e \in \mathbf{E}} c_e U_e$$
$$st. \begin{cases} \forall n \in \mathbf{N}, \forall i,j \in \mathbf{P}(i \neq j): \sum_{e \in \mathbf{O}(n)} P_e^{ij} - \sum_{e \in \mathbf{I}(n)} P_e^{ij} = \begin{cases} 1 & n = i \\ -1 & n = j \\ 0 & \text{otherwise} \end{cases} \\ \forall i,j \in \mathbf{P}(i \neq j): \quad \sum_{e \in \mathbf{E}} d_e P_e^{ij} \leq D^{ij} \\ \forall e \in \mathbf{E}: \quad c_e U_e \geq \begin{cases} \max\limits_{F_e^{ij} \geq 0} \sum_{i \in \mathbf{P}} \sum_{j \in \mathbf{P}(i \neq j)} P_e^{ij} F_e^{ij} \\ st. \begin{cases} \forall i \in \mathbf{P}: & \sum_{j \in \mathbf{P}(j \neq i)} F_e^{ij} \leq B_i^{in} \\ \forall j \in \mathbf{P}: & \sum_{i \in \mathbf{P}(i \neq j)} F_e^{ij} \leq B_j^{out} \end{cases} \end{cases} \end{cases} \tag{3}$$

In this formulation, the awkward capacity constraints are expresses by using a maximisation to guarantee the capacity in the *worst case* scenario. It is this subsidiary optimisation that prevents formulation (3) from being solved by MILP solver straightaway. However, by using the technique developed in [6], it can be reformulated as a simple MILP.

Reformulation 3 (minimise bandwidth reservation - hose model) *Based on the hose model, the provisioning VPN to minimise the total reserved bandwidth for VPN can be* **reformulated** *as:*

$$\min_{\{P_e^{ij}\in\{0,1\},D_{ie}\in[0,1],D_{ej}\in[0,1],U_e\in[0,1]\}} \sum_{e\in\mathbf{E}} c_e U_e$$
$$st. \begin{cases} \forall n\in\mathbf{N},\forall i,j\in\mathbf{P}(i\neq j): \displaystyle\sum_{e\in\mathbf{O}(n)} P_e^{ij} - \sum_{e\in\mathbf{I}(n)} P_e^{ij} = \begin{cases} 1 & n=i \\ -1 & n=j \\ 0 & \text{otherwise} \end{cases} \\ \forall i,j\in\mathbf{P}(i\neq j): \quad \displaystyle\sum_{e\in\mathbf{E}} d_e P_e^{ij} \leq D^{ij} \\ \forall e\in\mathbf{E}: \quad c_e U_e \geq \displaystyle\sum_{i\in\mathbf{P}} B_i^{in} D_{ie} + \sum_{j\in\mathbf{P}} B_j^{out} D_{ej} \\ \forall e\in\mathbf{E},\forall i,j\in\mathbf{P}(i\neq j): \quad D_{ie}+D_{ej}\geq P_e^{ij} \end{cases} \tag{4}$$

where D_{ie} and D_{ej} are dual variables. This new *reformulated* MILP can be efficiently solved by any MILP packages, such as CPLEX [7].

Furthermore, if the multi-path routing between the customer pair within VPN is allowed, this corresponds to relax variable $P_e^{ij}\in\{0,1\}$ into $P_e^{ij}\in[0,1]$ in MILP model (4). Therefore a very simple linear programming (LP) formulation is obtained as:

Relaxation 4 (minimise bandwidth reservation - hose model) *By using multi-path routing on the hose model, the provisioning VPN to minimise the total reserved bandwidth for VPN can be* **relaxed** *as:*

$$\min_{\{P_e^{ij}\in[0,1],D_{ie}\in[0,1],D_{ej}\in[0,1],U_e\in[0,1]\}} \sum_{e\in\mathbf{E}} c_e U_e$$
$$st. \begin{cases} \forall n\in\mathbf{N},\forall i,j\in\mathbf{P}(i\neq j): \displaystyle\sum_{e\in\mathbf{O}(n)} P_e^{ij} - \sum_{e\in\mathbf{I}(n)} P_e^{ij} = \begin{cases} 1 & n=i \\ -1 & n=j \\ 0 & \text{otherwise} \end{cases} \\ \forall i,j\in\mathbf{P}(i\neq j): \quad \displaystyle\sum_{e\in\mathbf{E}} d_e P_e^{ij} \leq D^{ij} \\ \forall e\in\mathbf{E}: \quad c_e U_e \geq \displaystyle\sum_{i\in\mathbf{P}} B_i^{in} D_{ie} + \sum_{j\in\mathbf{P}} B_j^{out} D_{ej} \\ \forall e\in\mathbf{E},\forall i,j\in\mathbf{P}(i\neq j): \quad D_{ie}+D_{ej}\geq P_e^{ij} \end{cases} \tag{5}$$

If the *ellipsoid algorithm* is used to solve this LP, there is a polynomial algorithm *in theory* for provisioning VPN under multi-path routing. This is the main conclusion of [5]. However, the LP model (5) we give here is more efficient and more scalable.

3 Numerical Results and Comparison

We present two examples to demonstrate the provisioning VPN based on the hose model and the pipe model, respectively, to minimise the total reserved bandwidth, while satisfying the propagation delay constraints.

AT&T Worldnet IP Network. The AT&T Worldnet IP backbone network topology [2] comprises 12 core routers spanning the continental U.S. There are 42 directed links with assumed capacity of 2 Mbps. The randomly generated VPN consists of 5 customer endpoints with ingress bandwidth, egress bandwidth and maximum propagation delay, which are specified in Table 1.

Table 1. VPN Specifications in AT&T Network

Customer Endpoint	Ingress Bandwidth	Egress Bandwidth	Maximum Delay
Chicago	300 Kbps	300 Kbps	5 ms
San Francisco	200 Kbps	200 Kbps	5 ms
Dallas	400 Kbps	400 Kbps	5 ms
New York	500 Kbps	500 Kbps	5 ms
St.Louis	600 Kbps	600 Kbps	5 ms

To minimise the bandwidth reservation, the provisioned VPN on the pipe model and on the hose model is shown in Figure 1. By using the pipe model, 18 network links are used and total reserved bandwidth is 17600 Kbps, while using the hose model, only 8 network links are used and total reserved bandwidth is 4200 Kbps. It is worthy to point out that based on the hose model the provisioned VPN has a *tree structure* routes. And compared with the pipe model, the over-provisioning factor 4.19 = 17600/4200 of total reserved bandwidth can be reduced by the hose model.

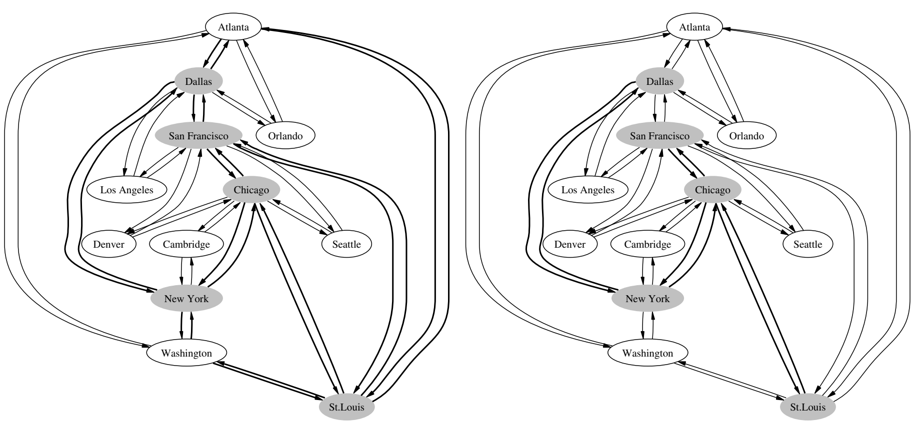


Fig. 1. Provisioning VPN in AT&T Network

Table 2. Provisioning VPNs on dexa Network - Hose Model

ID	NCE	TRB	Vars	Cstrs	CPU	OPF
1	5	23584	6344	7245	4.40	4.58
2	6	69584	8788	10625	1.18	9.31
3	6	286000	8708	10585	3.33	
4	6	16832	9028	10745	14.13	5.23
5	7	17664	12200	14945	20.11	8.17
6	8	30338	15860	19845	6.03	
7	8	30010	15860	19845	52.39	9.47
8	10	153392	24540	31693	14.66	

Table 3. Provisioning VPNs on dexa Network - Pipe Model

ID	NCE	TRB	Vars	Cstrs	CPU
1	5	107968	5002	2243	0.37
2	6	648000	7202	3179	0.99
3	6	Fail	7123	3139	0.04
4	6	88064	7442	3303	0.67
5	7	144384	10370	4575	0.60
6	8	Fail	13787	6059	0.08
7	8	284212	13786	6059	1.97
8	10	Fail	21979	9611	0.07

Schlumberger dexa Network. The Schlumberger dexa network topology consists of 53 core routers and 122 directed links. There are 8 VPNs needed to be provisioned which range from 5 to 10 customer endpoints.

To minimise the bandwidth reservation for each VPN, the provisioned VPNs are summarised in Table 2 (hose model) and Table 3 (pipe model). In these tables, the first 2 columns show VPN identity (ID) and the number of customer endpoints (NCE). The third column is the minimised total reserved bandwidth (TRB) in Kbps for each VPN (or Fail if the VPN cannot be provisioned). The next 3 columns show the MILP model size in term of the number of variables

Table 4. Provisioning VPNs on dexa Network - Multi-path Hose Model

ID	NCE	TRB	Vars	Cstrs	CPU
1	5	23584	6344	7245	3.54
2	6	69584	8788	10625	3.96
3	6	286000	8708	10585	6.10
4	6	16832	9028	10745	9.08
5	7	17664	12200	14945	8.25
6	8	30338	15860	19845	18.38
7	8	30010	15860	19845	16.57
8	10	153392	24540	31693	31.97

(Vars) and constraints (Cstrs), and the solution time (CPU) in seconds on Intel(R) Pentium(R) 4 CPU 2.00GHz.

Comparing the provisioned VPNs by the hose model (Table 2) with that by the pipe model (Table 3), the over-provisioning factor about 5 in term of total reserved bandwidth can be achieved, which is shown as OPF (ratio of TRB on pipe model and TRB on hose model) in the last column of Table 2. In particular, 3 VPNs (ID 3, 6 and 8) cannot be provisioned by the pipe model because of the limited link capacity.

By using multi-path routing on hose model, the provisioned VPNs are summarised in Table 4. Comparing the provisioned VPNs by the single-path routing (Table 2) with that by the multi-path routing (Table 4), it is surprising that total reserved bandwidth are all same, although the some different routing path are used by same customer pair within VPN!

4 Conclusions

Aim to minimise the bandwidth reservation, we investigate the provisioning VPN with QoS guarantees (*i.e.* bandwidth, propagation delay) over shared network infrastructure based on the hose model and the pipe model. The MILP formulations are presented and reformulated, which can be efficiently solved by standard MILP packages. The numerical results on benchmark networks show that the hose model can dramatically reduce the over-provisioning. Therefore, the hose model is a bandwidth efficient model to provision VPN.

Acknowledgements. We are grateful to Helmut Simonis and Tom Richards of Parc Technologies for providing Schlumberger dexa Network data and advices.

References

1. Medina, A., Taft, N., Salamatian, K., Bhattacharyya S. and C. Diot: *"Traffic matrices estimation: existing techniques and new directions"*, **ACM SIGCOMM'02**, August 19-23 (2002), Pittsburgh, USA
2. Duffield, N., Goyal, P. and A. Greeberg: *"A Flexible model for resource management in virtual private networks"*, **ACM SIGCOMM'98**, August 31 - September 2 (1998), Vancouver, Canada
3. Juttner, A., Szabo, I. and A. Szentesi: *"On bandwidth efficiency of the hose resource management model in virtual private networks"*, **IEEE INFOCOM'03**, March 30 - April 3 (2003), San Francisco, USA
4. Kumar, A., Rastogi, R., Silberschatz, A. and B. Yener: *"Algorithms for provisioning virtual private networks in the hose model"*, **ACM SIGCOMM'01**, August 27-31 (2001), San Diego, USA
5. Erlebach, T. and M. Ruegg: *"Optimal bandwidth reservation in hose-model VPNs with multi-path routing"*, **IEEE INFOCOM'04**, March 7-11 (2004), Hong Kong, China
6. Q. Xia: *"Traffic diversion problem: reformulation and new solutions"*, **Submitted for Publication**, August (2004)
7. ILOG Inc.: *"ILOG CPLEX 6.5 User's Manual"*, `http://www.cplex.com`, 1999

Potential Risks of Deploying Large Scale Overlay Networks

Maoke Chen and Xing Li

Network Research Center, Tsinghua University,
Beijing 100084 P.R. China

Abstract. In recent years, a variety of overlay networks are created over the Internet via virtual links. We investigate the impact of the virtual link configuration on the network capacity with a dual-layer lattice network model, focusing on the critical value of the input rate of user traffics. A mean-field theory suggests that the critical traffic is, approximately, inversely proportional to the average trip of virtual hops on the infrastructure. Simulations verify the analytic result and further show that the behavior of the overlay-physical network interactions is significantly divergent with different link configurations. Therefore, the optimization of virtual links will be conducive to improving the effectiveness of overlays.

1 Introduction

In recent years, a variety of overlay networks (or, overlays) have been deployed in today's Internet, with the utilization of virtual links (or, "tunnels" as regularly called in the Internet community [1, 2, 3]). We are focusing on overlays connected with virtual links, along which the user traffics are routed in the way of packet-switching, i.e. storing and forwarding. Two major classes of overlays fall into this category: (1) peer-to-peer resource sharing systems whose virtual topologies are built on the application layer[1] and (2) dual-stack overlays such as IPv6 (Internet Protocol version 6) networks over traditional Internet which is running IPv4[4], or, vice-versa, IPv4 over IPv6[3]. Rapid deployment of these systems motivates modeling overlay-physical network interactions. Overlays like the hyperlink topology among the World Wide Web (WWW) (which has been deeply studied by physicists, see e.g.[5]), or content-addressable networks (CAN) for information retrieving [6], and any other kinds of distributed hash table (DHT) [7] are not categorized as packet delivery systems, and accordingly are out of the range of this paper.

Our methodology of modeling overlay-physical network system with regular lattices originates from the works of Deane et al. [8], Ohira et al. [9], Fukś and Lawinczak [10], and Chen [11], where people studied phase-transition phenomena in latticed packet-switched network models.

[1] Such as Gnutella (http://www.gnutella.com/), KaZaA (http://www.kazaa.com), eDonkey/overnet (http://www.edonkey.com/) and so forth.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 508–517, 2005.

In this paper, we propose and study a new, dual-layer lattice model in order to extend the understanding of phase-transition behavior in packet-switched networks to that of overlay-physical network interactions. The sections of the paper are organized as follows. First, it is explained why the critical traffic plays the central role in our analysis. Then three components of the model, i.e. the overlay, the physical network and the virtual link configuration, are defined respectively. Then a mean-field theory is presented for the critical traffic of end-users in the overlays analytically. Further, we investigate a set of simulation cases, showing how the virtual link configuration impacts the overlay-physical network interaction dramatically. Finally, we conclude the paper with some guide to overlay practices.

2 Meanings of the Critical Traffic

Ohira et al. used the term of "critical traffic" to describe the point where the phase transition happens in their latticed router-host model [9]. Fukś et al. then studied a similar phenomenon in a regular lattice with identical nodes [10]. These are almost the earliest works in modeling phase transition behavior in the Internet, though they contains nothing but extensions of people's knowledge on the stability of queueing systems. Actually, the so-called "critical traffic" is just the input traffic rate at each node, critical to the stability of queues in a packet-switched network with unlimited buffers.

Therefore the critical traffic represents the capability that a network can provide for end user communications without loss. In a single-server queueing system, the critical traffic is equal to the service rate, while it is degraded in a queueing network. In a packet-switched network, as observed by Fukś [10] and analyzed by Chen [11], the degree of degradation is determined by the end-to-end delay that an arbitrary packet is routed among network nodes without being queued.[2]

For an overlay network connected by virtual links, we concern the capability of the physical infrastructure handling communications among end users — the users of the overlay. Because the meaning of "end-to-end" for users (at the layer of overlay) differs from that for network nodes (at the layer of physical infrastructure), the behavior of critical traffic is not only impacted by both overlay and physical network topologies and by the routing mechanisms, but also impacted by the relationship between them, i.e. the virtual link configuration. Therefore, we have built a latticed overlay-physical network model and focused our study on the influence of virtual link configuration to the model's critical

[2] This proposition stands under the context of the Mean-Field Theory, where we ignore the spatial fluctuations of the traffic observed at each node in a network; otherwise, the critical traffic might be various at different positions and then its global value should be the lower bound of local values. We take the advantage of Mean-Field Theory to avoid the difficulties of studying the local dynamics and get an approximated but simple and inspiring result.

traffic under the context of the Mean-Field Theory. It is shown in both analysis and simulations that blindly created virtual links are harmful to the network capability, unless the deployed overlay network is small enough.

3 Model Definition

The proposed model consists of 3 major components: the overlay layer lattice $\aleph_{\mathrm{v}}$, the physical layer lattice $\aleph_{\mathrm{p}}$, and the virtual link configuration $\mathcal{F}$, which is defined with a mapping from overlay links to pairs of end points on the physical layer. (See Fig. 1) Both overlay and physical layers could be defined with any dimensionality. Later in this paper, we will take the simplest case of one-dimensional lattices to do simulations.

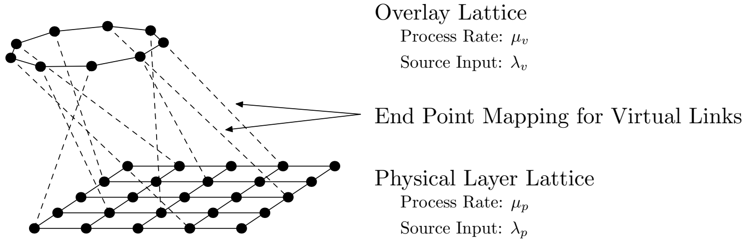


Fig. 1. A typical case of the model for overlay-physical network interaction

Packets might be generated by the overlay nodes, and called as "overlay packets"; or they might be generated within the physical layer, and the traffic caused by such packets are not of our concern, therefore we call them "background traffic". Traffics among the physical layer nodes consist of both background traffics and traffics injected from the overlay.

When an overlay node "forwards" a packet to the next-hop, it must encapsulate the packet with the physical-layer information of both the current node and the next-hop, and move the packet to the physical layer. Until the packet arrives at its final destination on the overlay, it should have come back to the overlay and soon re-enter the physical layer for several times.

It should be emphasized that the forwarding processes in overlay are performed in the context of the metrics in its own topology. That is, overlay nodes choose next-hop for packets independently, without referring the physical layer topology and state.

For either overlay or physical layer, we respectively use d, V, E, λ and μ to denote its dimensionality, node set, link set, input traffic on a node and process rate on a node[3], and apply subscript $_{\mathrm{v}}$ or $_{\mathrm{p}}$ to indicate the layer. The configu-

[3] For the convenience of the discussion, we suppose both packet generation and packet processing are Markovian.

ration for virtual links then is represented with a mapping from the overlay link set to the set of node-pairs in the physical layer, i.e.

$$\digamma : \quad E_{\mathrm{v}} \mapsto V_{\mathrm{v}} \times V_{\mathrm{v}} \tag{1}$$

4 A Mean-Field Theory

Previous works have shown that the mean value of an observed traffic in a packet-switched network is equal to the input traffic λ amplified by the end-to-end forwarding times of packet delivery without queueing, $\mu\bar{\tau}^f$, where the superscript f means "free", i.e. the "free delay" as called in literature [10, 11].

$$\overline{\sigma} = \lambda\mu\bar{\tau}^f \tag{2}$$

If the spatial fluctuation of σ could be ignored, and accordingly the mean value $\overline{\sigma}$ could represent all the local values, then the critical rate of input traffic, λ^c, is equal to the reciprocal of the free delay, i.e. $\lambda^c = \frac{1}{\bar{\tau}^f}$ [11]. The superscript c means "critical".

For the overlay-physical network system, we'd like to find a quantitative relation between the traffic observed in the physical layer, σ_{p} and the structure of the system, esp. the end point mapping, $\digamma$.

4.1 Zero-Background Condition

The analysis is focused on the capability of physical network carrying end user traffics. Therefore, it is assumed that the background traffic is zero, i.e. $\lambda_{\mathrm{p}} = 0$. This is called zero-background (shortly, ZBK) condition in this paper. The physical layer observed traffic under ZBK condition is denoted by $\sigma_{\mathrm{p}}^{\mathrm{ZBK}}$, while the critical traffic of overlay inputs is $\lambda_{\mathrm{v}\rightsquigarrow\mathrm{p}}^{c-\mathrm{ZBK}}$. The notation $\mathrm{v} \rightsquigarrow \mathrm{p}$ means such traffic enters the physical layer from the overlay. It may leads the physical layer queues into an unstable state.

4.2 Critical Traffic Under Zero-Background Condition

Strictly speaking, the end points of virtual links accept injected traffics from the overlay and meanwhile undertake the tasks of forwarding, so more traffic should be observed at these points. However, for the convenience of analysis, we take the ease of mean-field theory and study the value of the spatially average injected traffic, $\overline{\lambda}_{\mathrm{v}\rightsquigarrow\mathrm{p}}$.

Firstly, as is shown in Eqn. (3), a piece input traffic leaves its origin in overlay for a trip in the physical layer. In average, there are $\frac{|V_{\mathrm{p}}|}{|V_{\mathrm{v}}|}$ physical nodes can provide service for this traffic.

$$\overline{\lambda}^o_{\mathrm{v}\rightsquigarrow\mathrm{p}} = \lambda_{\mathrm{v}} \frac{|V_{\mathrm{v}}|}{|V_{\mathrm{p}}|} \tag{3}$$

The superscript o means "original", indicating traffics originally injected into the physical layer without re-entering.

After an end-to-end trip in the physical layer, a packet returns to the overlay, getting the route information for the next-hop, and re-enters the physical layer. A same overlay packet should inject into physical network several times. Therefore, the total injected traffic should be $\overline{D}_{\mathrm{v}}$ times of the original, where $\overline{D}_{\mathrm{v}}$ is the average path length between any pair of overlay nodes, measured with hop count.

$$\overline{\lambda}_{\mathrm{v}\rightsquigarrow\mathrm{p}} = \overline{\lambda}^{o}_{\mathrm{v}\rightsquigarrow\mathrm{p}} \overline{D}_{\mathrm{v}} \tag{4}$$

Finally, under the ZBK condition, the free delay of physical layer packets is just the delay of the injected packets. This should be the average process time $\frac{1}{\mu_{\mathrm{p}}}$ times the average end-to-end trip of the injected packets, say $\overline{D_{\mathrm{p}}(\aleph_{\mathrm{v}}|\aleph_{\mathrm{p}})}$.

$$\overline{\tau}_{\mathrm{p}}^{f-\mathrm{ZBK}} = \frac{1}{\mu_{\mathrm{p}}} \overline{D_{\mathrm{p}}(\aleph_{\mathrm{v}}|\aleph_{\mathrm{p}})} \tag{5}$$

$$\overline{D_{\mathrm{p}}(\aleph_{\mathrm{v}}|\aleph_{\mathrm{p}})} \triangleq \frac{\displaystyle\sum_{x\in V_{\mathrm{v}}} \sum_{y\in N_{\mathrm{v}}(x)} \sum_{z,w\in V_{\mathrm{v}}} D_{\mathrm{p}}(\digamma(\overrightarrow{xy})) \left\{ \frac{1}{|\mathcal{P}_{\mathrm{v}}(z,w)|} \sum_{P\in\mathcal{P}_{\mathrm{v}}(z,w)} \chi(\overrightarrow{xy}\in E(P)) \right\}}{\displaystyle\sum_{x\in V_{\mathrm{v}}} \sum_{y\in N_{\mathrm{v}}(x)} \sum_{z,w\in V_{\mathrm{v}}} \left\{ \frac{1}{|\mathcal{P}_{\mathrm{v}}(z,w)|} \sum_{P\in\mathcal{P}_{\mathrm{v}}(z,w)} \chi(\overrightarrow{xy}\in E(P)) \right\}} \tag{6}$$

The average end-to-end trip of the injected packets is determined by the mapping $\digamma$, as shown in Eqn.(6), where $N_{\mathrm{v}} : V_{\mathrm{v}} \mapsto 2^{V_{\mathrm{v}}}$ determines the neighbor set of an overlay node, and $N_{\mathrm{v}}(x) \triangleq \{y \in V_{\mathrm{v}} : \overrightarrow{xy} \in E_{\mathrm{v}}\}$; $\mathcal{P}_{\mathrm{v}}(z,w)$ is the set of shortest path from node z to w over the overlay; χ is the event indicator that equals to 1 if the event expression is true and to 0 otherwise; $E(P)$ represents all the arcs on a path P; $D_{\mathrm{p}}(\cdot,\cdot)$ is the distance between a pair of points on the physical layer, measured with hop count, and therefore $D_{\mathrm{p}}(\digamma(\overrightarrow{xy}))$ is the physical length of virtual link from x to y.

Note that Eqn.(6) might be seen as the ensemble average of virtual link lengths, which are weighted by their utilization.

Thus, recalling the Eqn.(2) that has been proved in [11], we have

$$\overline{\sigma}_{\mathrm{p}}^{\mathrm{ZBK}} = \frac{|V_{\mathrm{v}}|}{|V_{\mathrm{p}}|} \overline{D_{\mathrm{p}}(\aleph_{\mathrm{v}}|\aleph_{\mathrm{p}})} \lambda_{\mathrm{v}} \overline{D_{\mathrm{v}}} \tag{7}$$

With the help of the Jackson Theorem [12, 13], a law on the critical traffic under zero-background condition is obtained.

Law 1 (Zero-Background Critical Traffic). *In a overlay-physical network system, under the condition of zero-background, approximately, the critical value of the input rate at each overlay node with respect to the physical layer capability is:*

$$\lambda_{\mathrm{v}}^{c-\mathrm{ZBK}} \simeq \frac{\mu_{\mathrm{p}}|V_{\mathrm{p}}|}{\overline{D_{\mathrm{p}}(\aleph_{\mathrm{v}}|\aleph_{\mathrm{p}})}|V_{\mathrm{v}}|\overline{D_{\mathrm{v}}}} \tag{8}$$

Notice that, numerically, the $\overline{D_{\mathrm{v}}}$ equals to the free delay of a lattice network with the same topology of the overlay, i.e. $\overline{D_{\mathrm{v}}} = \mu_{\mathrm{v}}\bar{\tau}^f$, and the Eqn.(8) may be simplified to

$$\lambda_{\mathrm{v}}^{c-\mathrm{ZBK}} \simeq \frac{\mu_{\mathrm{p}}\left(|V_{\mathrm{p}}|/|V_{\mathrm{v}}|\right)}{\mu_{\mathrm{v}}\overline{D_{\mathrm{p}}(\aleph_{\mathrm{v}}|\aleph_{\mathrm{p}})}}\lambda^c \tag{9}$$

Eqn.(9) expresses how the overlay-physical network relationship affects their interaction: on one hand, the carrying capability is improved by the scale of the physical network; on the other, it is degraded by the average length of the virtual links.

In both Eqn.(8) and (9), we take the symbol of similar equality instead of taking average for $\lambda_{\mathrm{v}}^{c-\mathrm{ZBK}}$, because the Mean-Field Theory is employed to $\sigma_{\mathrm{p}}^{\mathrm{ZBK}}$ but the global critical traffic is the minimum rather than the average of its local values. The approximate analytical result conforms to the reality, when the spatial fluctuations of the observed traffic could be omitted.

5 Simulations

The Law 1 gives a critical point for the overlay user traffic. In this section, simulations over a group of typical instances of the model defined in Section 3 will provide further understandings on the overlay-physical network interactions.

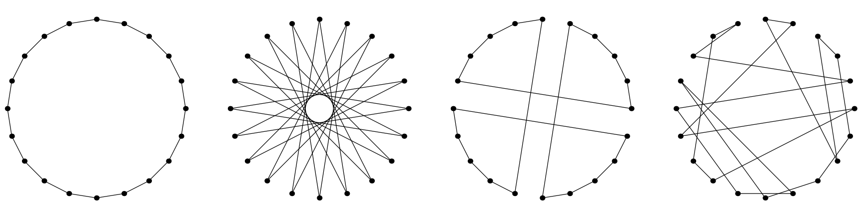
The instances for simulation are defined in one-dimensional periodical lattices for both overlay and the physical layers. The periodicity ensures that the geometries are symmetric.

5.1 Instances and Configurations

The simulated instances are different in only the virtual link configuration, i.e. the end point mapping $\mathcal{F}$. The common components of the instances are:

- The physical network $\aleph_{\mathrm{p}}$: $d_{\mathrm{p}} = 1$, and $L_{\mathrm{p}} = 100$, and mark the nodes as $V_{\mathrm{p}} = \{0, 1, \cdots, 99\}$, and the link can be written as

$$E_{\mathrm{p}} = \{(x, y) : y \equiv x \pm 1 \mod L_{\mathrm{p}}; \quad x, y \in V_{\mathrm{p}}\}$$



(a) (b) (c) (d)

Fig. 2. Four instances of the node mapping $\mathcal{F}'$

For the convenience of digital computation, we take $\mu_{\mathrm{p}} = 1$, which means one packet processed by a node within a unit of time.

- The overlay $\aleph_{\mathrm{v}}$: $d_{\mathrm{v}} = 1$, and $L_{\mathrm{v}} = 20$, and mark the nodes as $V_{\mathrm{v}} = \{0, 1, \cdots, 19\}$, and the link can be written as

$$E_{\mathrm{v}} = \{(x, y) : y \equiv x \pm 1 \mod L_{\mathrm{v}}; \quad x, y \in V_{\mathrm{v}}\}$$

It is also assumed that $\mu_{\mathrm{v}} = 1$. Therefore, the free delay and the critical traffic of a network with the same scale of the overlay are respectively

$$\bar{\tau}^f = 5; \quad \lambda^c = 0.2$$

And the scale ratio $|V_{\mathrm{p}}|/|V_{\mathrm{v}}| = 5$.

- There are not two links sharing a common end point, and accordingly, the link end point mapping $\digamma$ can be derived from a node mapping $\digamma' : V_{\mathrm{v}} \mapsto V_{\mathrm{p}}$. In the instances, the target set of this mapping $\digamma'(V_{\mathrm{v}})$ is uniformly selected among V_{p}. Because the geometry is periodical, it might be as well to simply take $\digamma'(V_{\mathrm{v}}) = \{0, 5, 10, \cdots, 95\}$; the details of the mapping are different among the instances.
- The running time of simulation: $k = 1000$ steps.
- Input traffic in the overlay: $\lambda_{\mathrm{v}} = 0.002 \sim 0.998$ with a step of $\Delta\lambda_{\mathrm{v}} = 0.002$.
- Background traffic rate: $\lambda_{\mathrm{p}} = 0$ (zero-background).

Four instances with difference in node mapping are taken for simulation. They are shown in Fig.2, where nodes in the set $\digamma'(V_{\mathrm{v}})$ is enlarged, and lines represent virtual links while the physical links are expresses by the adjacency of nodes along the circle without being drawn out.

In order to compare the impact of different virtual link configuration, the four instances are set so:

(a) The shortest path between any pair of nodes in the overlay conforms to the shortest path between the mapped pair in the physical layer, and the packet is transmitted without any detour accordingly. Its $\overline{D_{\mathrm{p}}(\aleph_{\mathrm{v}}|\aleph_{\mathrm{p}})} = 5$.

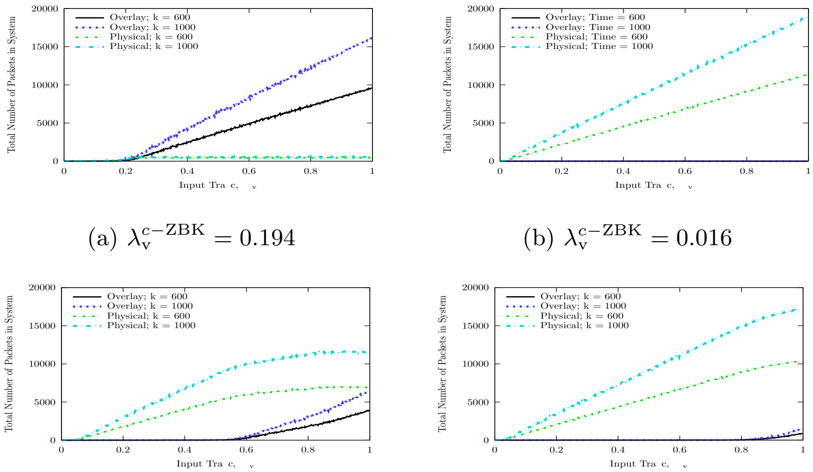
(b) Each virtual link bypasses many nearer nodes in the physical layer, and the shortest path for each pair of nodes in the overlay corresponds to a very long trip with serious detour. Its $\overline{D_{\mathrm{p}}(\aleph_{\mathrm{v}}|\aleph_{\mathrm{p}})} = 45$.

(c) Some virtual links are set bypassing nearer nodes intentionally, and some path experiences serious detour. In this case, $\overline{D_{\mathrm{p}}(\aleph_{\mathrm{v}}|\aleph_{\mathrm{p}})} = 13$.

(d) A random setting, where the links are configured blindly. For the sample case presented here, we have $\overline{D_{\mathrm{p}}(\aleph_{\mathrm{v}}|\aleph_{\mathrm{p}})} = 22.1$.

5.2 Queueing Length Behavior

Fig.3 shows the simulation results of queueing behavior for the instances correspondingly. In each subfigure for a instance, there are two curve representing the total number of queued packets at $k = 600, 1000$ for either the overlay or the physical layer. The point where curves for the two moments depart from each other indicates the critical traffic. The following facts about these subfigures should be noted:



(a) $\lambda_{\rm v}^{c-\rm ZBK} = 0.194$ (b) $\lambda_{\rm v}^{c-\rm ZBK} = 0.016$

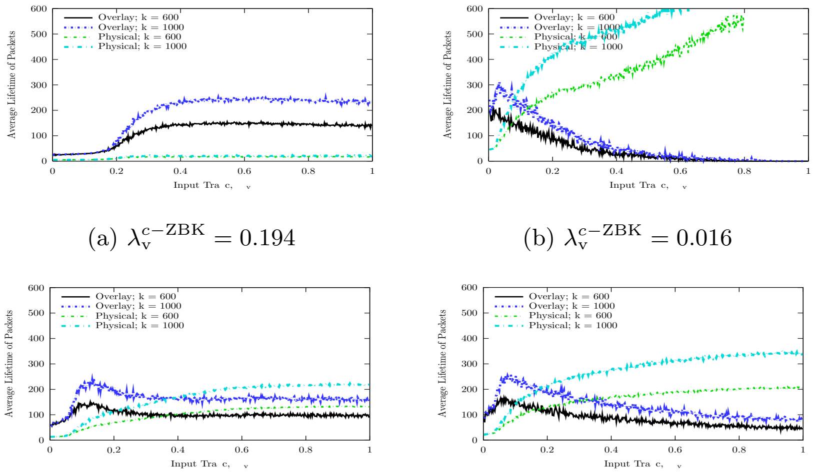
(c) $\lambda_{\rm v}^{c-\rm ZBK} = 0.066$ (d) $\lambda_{\rm v}^{c-\rm ZBK} = 0.034$

Fig. 3. Packets in queues after $k = 600, 1000$ steps of iteration

1. The zero-background critical traffic is obviously impacted by the link configurations. Its value is a little smaller than what the Mean-Field Theory approximates. This is related to the fact that the Mean-Field Theory has omitted the fluctuations.
2. It must be indicated that the queues in the physical layer of the instance (a) do really increase to infinity when input rate $\lambda_{\rm v} > 0.2$, though the two curves of moment $k = 600$ and $k = 1000$ are much closer to each other.
3. In Fig.3(a), the critical point for overlay queues is almost the same as that for physical layer queues, but in other instances, the former one is shifted right but the latter is shifted left.
4. It is dramatic that the behavior of growth after the critical point is so different between instance (a) and all the others. In instance (a), the overlay queues significantly grow after the critical point, while the physical layer queues' growth seems trivial. In the other instances, however, the overlay users may not feel the traffic jam in the physical infrastructure.

Fig. 4 shows the simulation result for delay behavior. Unlike in the Fig. 3, the critical point for physical layer end-to-end delay is also critical for end-to-end delay of overlay packets. Moreover, it is impressive in case (b)-(d) where the overlay delay is much decreased at higher traffic rate. This happens because only a small portion of packets with nearer destinations have completed their trips at the moment of observation.

The queueing behavior means that, if virtual links are not well configured, overlay traffic control techniques might be never useful. If the virtual links are created blindly, then the capacity that physical network carry user communication is seriously degraded. Packets are congested in the physical layer queues



(a) $\lambda_{\rm v}^{c-\rm ZBK} = 0.194$ (b) $\lambda_{\rm v}^{c-\rm ZBK} = 0.016$

(c) $\lambda_{\rm v}^{c-\rm ZBK} = 0.066$ (d) $\lambda_{\rm v}^{c-\rm ZBK} = 0.034$

Fig. 4. Average delay of packet having arrived after $k = 600, 1000$ steps of iteration

rather than in the overlay, and therefore overlay facilities might not make right control.

On the other hand, though the ever-arrived packets' delay behaves the same that long delay in physical layer but short in overlay is observed, the overlay end systems can observe that much more requests timed out when the physical layer is jammed. Therefore, end-to-end measurement rather than monitoring at intermediate systems would be much more useful for congestion and performance control in overlay-physical network systems.

6 Conclusions

This paper discusses the problem that the virtual links impacts the transmission ability of the physical network for end users. Both analytical and simulation results suggest that improperly or blindly configured virtual links might be harmful. The risks would be much more serious for an overlay network which is deployed on a large scale. The degradation of the physical network performance is significant as overlay packets are routed with detours.

In view of practice, however, virtual links (or tunnels) are the reality and are useful in many circumstances. According to the analysis presented in this paper, it is suggested that deploying virtual links for overlay networks does not impact the network performance seriously only if (1) the overlay network is far less scaled (with a big ratio of $|V_{\rm p}|/|V_{\rm v}|$) or (2) the service power of the overlay network nodes is great enough (with a big $\mu_{\rm p}$ with comparison to $\overline{D}_{\rm v}$); otherwise, the virtual link configuration for the overlay network is better to be optimized.

As inspired by the simulation results, a possible way for the optimization is making virtual links so that virtual paths will conform to the paths in the physical layer as much as possible. This can be achieved through providing the physical layer structure information for various overlays via a common topology service. On the other hand, from the view of network management, it is important to uncover all virtual links and identify those that are created poorly. The authors have embarked on studying this topic over nation-wide dual-stack overlay networks.

References

1. Woodburn, R.A., Mills, D.L.: Scheme for an internet encapsulation protocol: Version 1. RFC 1241, Internet Engineering Task Force (1991)
2. Provan, D.: Tunneling IPX traffic through IP networks. RFC 1234, Internet Engineering Task Force (1991)
3. Conta, A., Deering, S.E.: Generic packet tunneling in IPv6 specification. IETF RFC 2473 (1998)
4. Gilligan, R., Normark, E.: Transition mechanisms for IPv6 hosts and routers. IETF RFC 2893 (2000)
5. Albert, R., Barabási, A.L., Jeong, H., Bianconi, G.: Power-law distribution of the world wide web. Science **287** (2000) 2115a
6. Ratnasamy, S., Francis, P., Handley, M., Karp, R., Shenker, S.: A scalable content-addressable network. In: Proceedings of ACM SIGCOMM. (2001)
7. Balakrishnan, H., Kaashoek, M.F., Karger, D., Morris, R., Stoica, I.: Looking up data in p2p systems. Communications of the ACM (2003)
8. Deane, J., Smythe, C., Jefferies, D.: Self-similarity in a deterministic model of data transfer. Journal of Electronics **80** (1996) 677–691
9. Ohira, T., Sawatari, R.: Phase transition in a computer network traffic model. Physics Review E **58** (1998) 193–195
10. Fukś, H., Lawniczak, A.T.: Performance of data networks with random links. Mathematics and Computers in Simulation **51** (1999) 101–117
11. Chen, M., He, T., Li, X.: A Mean-Field Theory of cellular automata model for distributed packet networks. In: Proceedings of ICOIN 2004. Volume I. (2004) 372–381
12. Sheng, Y.Z.: Queueing Theory and its Applications in Computer Communications (in chinese). Beijing University of Post and Telecommunication Press (1998)
13. Kleinrock, L.: Queueing Systems. Volume I: Theory. Jon Wiley & Sons (1975)

Utility-Based Buffer Management for Networks*

Cedric Angelo M. Festin[1] and Søren-Aksel Sørensen[2]

[1] Department of Computer Science, University of the Philippines, Diliman, QC, 1101 Philippines
cmfestin@up.edu.ph

[2] Computer Science Department, University College London, Gower Street, WC1E 6BT UK
S.Sorensen@cs.ucl.ac.uk

Abstract. User satisfaction from a given network service or resource allocation can be viewed as having two aspects, a state and a degree. The state defines whether the user is happy or unhappy. A user is happy when its expectations are met. The degree defines the level of happiness or unhappiness. We present the use of perceived knowledge of the state and degree of user satisfaction in managing router resources and functions and examine whether such knowledge could help a router improve local resource allocation decisions. We describe our formulation, Value-Based Utility (VBU) that incorporates both aspects of user satisfaction. We establish a framework of VBU use and demonstrate its application to buffer management. We propose a FIFO scheme that uses VBU and evaluate its success in meeting user expectations. The main conclusion we draw from this work is that the VBU framework offers a different perspective of performance definition and analysis and allows for the effective distribution of resources especially in times of high demand and low resource availability. Its adoption into existing traffic management schemes is further motivated by the improved performance of our proposed scheme over its non-VBU aware counterpart.

1 Introduction

In terms of happiness with a given service, users of a multiservice network can be in one of two *states*. They could either be happy or unhappy. When resources are low and demand for them is great, it is difficult to make every user happy but it is not impossible. Resources from satisfied users may be transferred to the dissatisfied users to try to make them less unhappy or even happy. This is possible because applications impose variable demands. Some applications may have high *expectations* of the service they need while others do not. This expectation is an indicator of how an application perceives a performance target or requirement.

* This work is supported by research grants from University of the Philippines (UP), UP Engineering Research and Development Foundation Inc. (UPERDFI) and Diliman Computer Science Foundation (DCSF).

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 518–526, 2005.

For an application who has expectations to be *satisfied*, the network must provide a service equal to its requirements. Any excess can only make it *happier* while failure to meet such requirements results in *dissatisfaction*. This notion of the *degree* (how much more or less) of *satisfaction* an application derives from a service has often been overlooked. This is because service concerns focus more on meeting quantitative demands than on qualitative attributes like satisfaction. We develop a formulation called *Value-Based Utility* (VBU) to quantify both the state and degree of satisfaction. The formulation is simple and its construction is fairly straightforward. Value-Based Utility uses the QoS requirements to define a *utility function* that associates a utility value with the service received by (or promised to) an application. Given this value, we can then characterise the application's state and degree of satisfaction with any given service.

In this paper we adopt a policy based on utility. However, the way we define and use utility is slightly different from its normal usage. In economics, utility is mainly used to express user preferences for choices. These preferences have often been linked to pricing [2] [7] [8]. This means they have assumed that if you prefer A over B, you are willing to pay more for A than for B. We do not assume this link between preference and pricing. We use utility to express a preference for A but this preference is not necessarily linked to the willingness and capability to pay for A. The preference we deal with is solely based on need. For example, consider a voice application with a utility function U, and two service bundles A and B with the following performance characteristics:

- A :<1000 ms delay @ 97% of the time, 20% loss, 32Kbps>
- B :<1000 ms delay @ 90% of the time, 20% loss, 16Kbps>

Economic utility states that the voice application prefers A to B if the application of U to A yields a higher value than the application of U to B; i.e., $U(A) > U(B)$. The problem with this proposition is that it does not tell us anything about the degree of satisfaction of the application. A may be a better service than B but the application may not be happy at all with a delay of 1000 ms. Similarly, A may be better than B but B may already be sufficient for the application. This would allow A to be allocated to some other user who needs it more.

For emerging network applications with strict QoS requirements, the use of utility functions to simply order and rank services is inadequate. It is incapable of capturing and modelling expectations of user requirements. In situations like these, it is more appropriate to use utility to represent user well-being. Information such as Value-Based Utility could be useful in managing resources, especially in resource-challenged environments or utilisation-conscious systems, because it identifies users who can possibly share some of their resources. In the succeeding sections, we develop these ideas.

2 Abstract Framework for Levels of Satisfaction

Quality of Service (QoS) requirements are often expressed as either a deterministic or a statistical bound [3] [4]. An example of a deterministic requirement is when the voice application in Section 1 requires that all packets should not be delayed by more than 1000 ms. Given the service choices A and B, neither would have been capable of delivering the desired service. A statistical bound is generally less restrictive. It is similar to a deterministic bound except that it has one additional parameter p, where p is the percentage of packets required to meet the bound. In a deterministic bound, this p is implied to be equal to one. For our voice example, instead of requiring all packets to meet the target, suppose we require that $p = 0.95$. This condition would result in service bundle A meeting the target while B still fails to meet expectations. Note that in both deterministic and statistical QoS representations, the service either succeeds in meeting the requirements or it does not. Unfortunately, questions like "How bad was the service for the deterministic case?" or "How good was the service for the statistical case?" cannot be answered.

To answer this, we first define the user expectation range to be some value between $happiness_{min}$ and $happiness_{max}$. These two points represent the level of user satisfaction given that the received or promised service is at least equal to the minimum requirements. A utility function U_i, which we formally define in Section 3, maps a user's received service to some value hopefully within the expected range. Whenever a user's utility $U_i = happiness_{max}$ then we say that the user has received the best possible service. If $U_i = happiness_{min}$ then the user's requirements have been minimally met. The worst that a user can be within this range is to be in a state of happiness. From a management perspective, it would be sufficient to operate the system at slightly above $happiness_{min}$ levels especially in times of high resource demands. There are no benefits for the network to expend resources that will not improve a user's state. This is because the user's expectation has already been achieved and the user is already happy. From the network's perspective, the users are indifferent to services evaluated within this happiness range.

In cases where some services fail to meet user expectations, applications will become unhappy. Similarly, as with satisfaction, there are varying degrees of unhappiness. We represent unhappiness and its levels as a range called the dissatisfaction levels. This area lies just below the $happiness_{min}$ value and is delimited by the point called $unhappiness_{max}$. Notice that $happiness_{min}$ is a threshold value because it is where the state of utility changes (from satisfaction to dissatisfaction or vice-versa). Service that is evaluated below this value can only make a user dissatisfied. If a user flow's utility $U_i = unhappiness_{max}$, then the user is unhappy and is the recipient of the worst possible service. Note that the expectation range is equivalent to satisfaction levels. This is because a user is not expecting to be unhappy.

3 General Form of the Utility Function

In this section, we formally define Value-Based Utility and develop a function to express satisfaction. We also highlight the important characteristics of the utility function.

3.1 Formulation

Let us assume that some percentage p from a flow of packets belonging to application i must meet some QoS target bound b in an interval Δt. To find a utility function U, we first define the user expectation range to be in $[0, 1]$. This range gives us the two points $happiness_{max} = 1$ and $happiness_{min} = 0$. We shall later see that the definition of $unhappiness_{max}$ is dependent on these two points and is a function of p. We next partition all packets N transmitted in a time interval Δt into two sets; one set S that meets the requirements and another set Q that does not. We can then associate the following ratios $P(S) = \frac{G}{N}$ and $P(Q) = \frac{N-G}{N}$ to these sets, where G is equal to the number of packets meeting the bound. The value $P(S) - P(Q)$ can be considered as the relative bias of a service either towards meeting targets when positive or to not meeting them when it is negative. However, this relation does not characterise how well performance has met expectation (p, b). We accomplish this by multiplying a factor α, which should be a function of p, to $P(Q)$ and then subtracting it from $P(S)$. Intuitively, we associate some benefit with $P(S)$ while $P(Q)*\alpha$ is the rate of how fast the benefit from $P(S)$ diminishes. Given the two points of $happiness_{max}$ and $happiness_{min}$, we find a suitable expression for α is given by $\frac{p}{1-p}$. We can also think of this ratio as the penalty factor for not meeting expectation p. Thus, $P(S) - P(Q) * \alpha$ gives us a utility function U for describing both user satisfaction and dissatisfaction within any specified time interval Δt.

Definition 1. *Value-Based Utility is an expression of user well-being. It uses a utility function to represent both the state and degree of user satisfaction (dissatisfaction). The expression*[1] *for the Value-Based Utility function is given by:*

$$U_{i,QoS,m,\Delta t}(p, b) = \frac{G}{N} - \frac{N-G}{N} * \frac{p}{q} \tag{1}$$

where

$U_{i,QoS,m,\Delta t}$ is flow i's utility for the specified QoS at point m during the time interval Δt,
p is the target percentage of packets that should meet QoS requirement,
b is the target QoS bound,
G is the number of packets meeting flow i's requirements,
N is the total number of packets seen, and
q is equal to $1 - p$.

A related work by Cao and Zegura [1] has extended max-min fairness to include utilities. Their approach is to maximise the minimum utilities of flows. We differ with their approach in that they do not distinguish between satisfaction and dissatisfaction. In addition to this, we also recognise that class differences

[1] This expression only represents the nondeterministic case. The reader is referred to [5] for an equivalent equation for the deterministic case.

affect the values of utilities. A third difference is in the way utilities are formulated. In their formulation, the utilities are functions of QoS while in our case utilities are functions of target QoS and bound (expectation).

3.2 Analysis

To verify that Equation 1 represents the state of user well-being, we consider the best, the minimal, and worst possible service. The best possible service occurs when $G = N$, which means that all the packets were serviced according to expectation (p, b). We see that the second term disappears and the equation simply evaluates to one or $happiness_{max}$. The second term also becomes zero when there is no expectation ($p = 0$). In this case utility will always be greater or equal to $happiness_{min}$.

When the requirement is exactly achieved, that is $\frac{G}{N} = p$, utility is equal to zero or $happiness_{min}$. An application will be satisfied if the utility from the service is above or equal to this level. A service that performs less than the expectation will have a utility value less than $happiness_{min}$, which is a negative utility U. The range of unhappiness begins at a point below the $happiness_{min}$ level and is bounded by $unhappiness_{max}$. We find the expression $unhappiness_{max}$ is equal to $-\frac{p}{q}$ (see Figure 1). This occurs when service to all packets fail to meet objectives ($G = 0$). Note that $unhappiness_{max}$ is not assigned a fixed point because of its dependence on user expectation p. It is also interesting to see that $unhappiness_{max} = -\alpha$. This should not be surprising since α is the total penalty with the negative sign indicating dissatisfaction. We note that for the deterministic case, the best service $G = N$ is just equivalent to the required service $p = 1.0$.

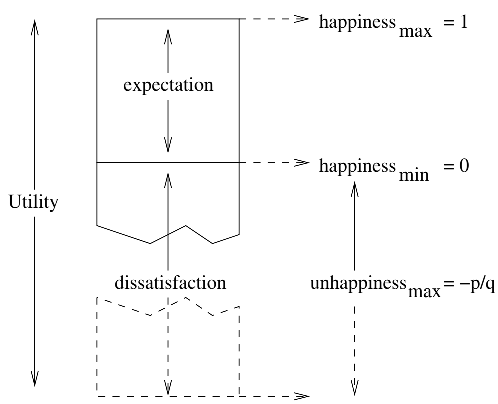


Fig. 1. The $happiness_{max}$, $happiness_{min}$ and $unhappiness_{max}$ are assigned the values 1, 0 and -p/q respectively

3.3 Key Terms and Definitions

From the analysis of section 3.2 we can infer from utility the success or failure of the service in meeting requirements. More importantly, from utility we can deduce the level of user satisfaction (dissatisfaction). This allows us to determine how far above or below users are from the happiness threshold, a measure that can be used for management. We now summarise some of the key terms and definitions we used in the previous sections for future reference. These are given below

Definition 2. State of Satisfaction. *A user can either be in a State of Happiness or in a State of Unhappiness depending on whether utility is negative or not.*

Definition 3. State of Happiness. *A user is in a state of happiness or simply happy if utility is either positive or zero* $(U \geq 0)$*. This implies user expectations were achieved.*

Definition 4. State of Unhappiness. *A user is in a state of unhappiness or simply unhappy if utility is less than zero* $(U < 0)$*. This implies user expectations were not achieved.*

Definition 5. Degree of Satisfaction. *The degree of satisfaction describes the level of user happiness or unhappiness. It is dependent on the magnitude of utility.*

Definition 6. Maximum Happiness. $Happiness_{max}$ (H_{max}) *is equal to one and occurs when G=N for* $0 \leq p < 1$*. It does not exist for p=1.0.*

Definition 7. Minimum Happiness. $Happiness_{min}$ (H_{min}) *is equal to zero and occurs when G/N=p for* $0 \leq p < 1$ *and G=N for p=1.0.*

Definition 8. Maximum Unhappiness. $Unhappiness_{max}$ (UH_{max}) *is equal to* $-p/q$ *when* $0 < p \leq 1$*. It does not exist for p=0.*

Definition 9. User Sensitivity. *Given two users A and B with p's* p_1 *and* p_2 *respectively, we say that user A: a) is more sensitive than B if* $p_1 > p_2$*; b) as sensitive as B if* $p_1 = p_2$*; and c) is less sensitive than B if* $p_1 < p_2$*. Generally, it is more difficult to satisfy a sensitive user than a less-sensitive user.*

4 A FIFO Scheme Using Value-Based Utility

In this scheme, a flow[2] is assigned a utility threshold based on its sensitivity. Higher expectation flows were assigned larger utility thresholds than lower expectation flows. When utility congestion occurs, the router attempts to keep

[2] This may be extended to classes by considering a group of flows.

the flow's level of satisfaction below this threshold value. Utility congestion is the condition where some flows are satisfied while others are not. This scheme prevents utility congestion from deteriorating by dropping packets from flows who have exceeded their threshold. Usually the packets belonging to a flow with lower expectations are the first to be dropped because they are considered less sensitive and given lower thresholds. It is expected that with this sacrifice, buffer space will become available for packets associated with unhappy flows when they arrive at the router. Normally, when all flows are satisfied, this scheme does not drop packets. We note that this scheme does not require per flow queuing and the number of operations is constant. Checking if all the flows are satisfied can be done in O(1) complexity.

4.1 Performance of VBU Under Different Utility Thresholds

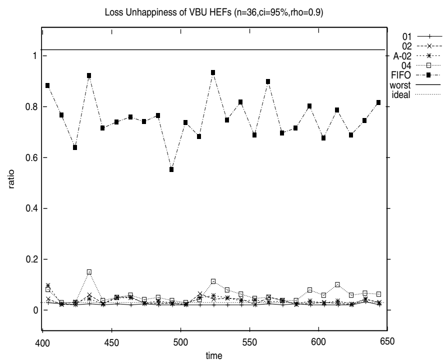
The performance of this scheme is similar to that of FIFO when the utility thresholds for all the flows are set to one. This is because a utility threshold of one means that no packet should ever be dropped, unless there is no physical space available. This section examines how much the performance can be improved when the thresholds are varied. In this paper, we have 36 sources which can be classified into three types of flows based on their expectations. These are 1) the High Expectation Flows (HEFS) which can tolerate 1% packet loss; 2) the Medium Expectation Flows (MEFS) which can lose up to 10% of their packets; and the Low Expectation Flows (LEFS) which can afford 20% packet loss. Exponential sources with mean 1 are used to generate packets which are fixed at 40 bytes. Each traffic source is connected to the router by an infinite bandwidth link[3]. A specified host is assigned to be a sink or receiver where measurements are taken. Traffic flows in one direction, from the source to the router and then finally to the receiver. The router has 400 bytes worth of buffer and a service rate of 160,000 Bytes per second *Bps*.

We now present some results from our experiments on VBU. Figure 2 shows the results of a select group of thresholds in terms of HEFS loss unhappiness (Figure 2(a)) and average utilities for the three different flow groups (Figures 2(b), 2(c) and 2(d)). The HEFS loss unhappiness (Figure 2(a)) shows the ratio of HEFS which are not satisfied. Figures 2(b), 2(c) and 2(d) displays the average utilities for satisfied and dissatified flows for each category. Observe that there are no unhappy MEFS and LEFS for all cases.

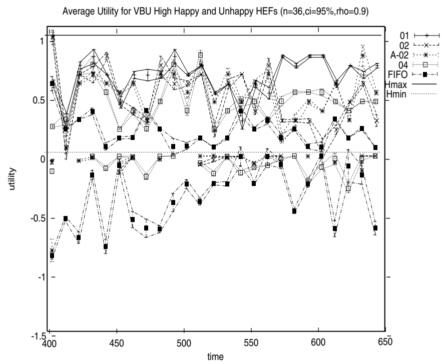
The group shown in Figure 2 used only two thresholds, one for the high expectation flows and another for both the medium and low expectation flows. The HEFS are assigned a threshold of 1.0. The sensitivity of the HEFS is the reason that we were unable to use threshold values lower than 1.0. The MEFS and LEFS were either assigned *0.10, 0.20* or *0.40* utility thresholds.

In Figure 2(a), we see that the combination of protecting the HEFS and decreasing the thresholds associated with the MEFS and LEFS lowers the number of unhappy HEFS. In the case of a *0.10* threshold, all the HEFS were satisfied.

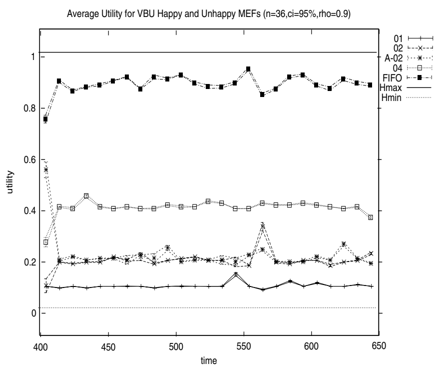
[3] The resulting delay is essentially zero. A similar assumption was used in [6].



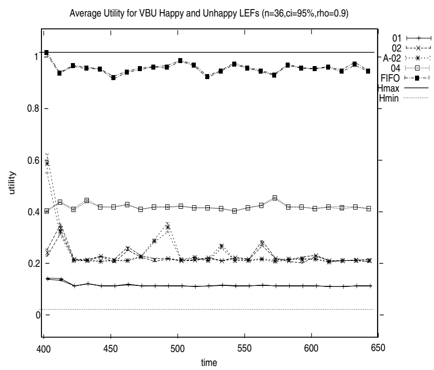
(a) HEFS Loss Unhappiness



(b) HEFS Average Utility



(c) MEFS Average Utility



(d) LEFS Average Utility

Fig. 2. VBU HEFS Loss Unhappiness and Average Loss Utility

The same trends can also be seen in Figure 2(b) where the average utilities increased as the thresholds of the MEFS and LEFS were decreased. The MEFS and LEFS utilities were kept almost constant at their assigned thresholds as shown in Figures 2(c) and 2(d). The occasional values rising above their assigned threshold, for example the MEFS with *0.20* threshold at time 560 seconds, can be attributed to the scheme finding that all flows are satisfied. Under this condition, the scheme allows flows to go above their threshold levels.

The results in this section have clearly shown that keeping less sensitive flows at acceptable and satisfactory levels can benefit the more demanding users such as the HEFS. We have successfully shown that under this policy, the overall happiness can be increased by selectively dropping packets. This indicates that VBU is a viable option for managing loss because it directly exploits knowledge of flow utilities.

5 Conclusions

The potential of using the state and degree of user satisfaction in effectively managing router resources and services has been largely unrealised. In this paper we claim that both types of information could be used locally inside the router for the purposes of buffer management. We have defined a formulation called *Value-Based Utility* (VBU) which is capable of expressing both the state and degree of user satisfaction. We demonstrated this by using the state and degree of satisfaction in managing a FIFO buffer. We assert that not only can the utility function defined in Section 3 provide a measure of user satisfaction given a resource allocation or service, but that it also can be used as a tool for management. VBU is a flexible framework that can be adapted into existing management mechanisms. Its adoption is further motivated by the improved performance of the FIFO VBU-based mechanism over its non-VBU aware counterpart. The uniqueness of our framework, however, is that it offers a new perspective on performance management. By combining both the state and degree of user satisfaction, we revise the definition of acceptable resource allocations.

References

1. Cao, Z., Zegura, E.: Utility max-min: An application oriented bandwidth allocation scheme. Proceedings of IEEE INFOCOM 99. (1999)
2. Cocchi, R., Shenker, S., Estrin D., Zhang, L.: Pricing in computer networks: motivation, formulation, and example. IEEE/ACM Transactions on Networking. **1:6** (1993) 614–627
3. Ferrari, D.: Client requirements for real-time communication services. RFC 1193. (1990)
4. Ferrari, D., Verma, D.: A scheme for real-time channel establishment in wide-area networks. IEEE Journal on Selected Areas in Communications. **8:3** (1990) 368–379
5. Festin, C.: Utility-based buffer management and scheduling for networks. PhD. Thesis. University College London (2002)
6. Jamin, S., Danzig, P., Shenker, S., Zhang, L.: A measurement-based admission control algorithm for integrated service packet networks. IEEE/ACM Transactions in Networking. **5:1** (1997) 56–70
7. Mackie-Mason, J. and Varian, H.: Pricing congestible network resources. IEEE Journal on Selected Areas in Communications. **13:7** (1995) 1141–1149
8. Shenker, S.: Fundamental design issues for the future internet. IEEE Journal on Selected Areas in Communications. **13:7** (1995) 1176–1188

Design and Implementation of a Multifunction, Modular and Extensible Proxy Server

Simone Tellini and Renzo Davoli

Department of Computer Science - University of Bologna,
Mura Anteo Zamboni, 7, I40127 Bologna, Italy
{davoli, tellini}@cs.unibo.it

Abstract. This paper introduces Prometeo[1] a multi-function, modular and extensible proxy server created as part of one the author's thesis work. We will discuss the needs that this project was meant to address: mainly the lack of an application with the aforesaid features, combined with native IPv6 support and ease of administration. Prometeo also provides a C++ framework which simplifies the development of networking applications. The design of Prometeo's will be described, starting with an overview of its components and modules and commenting on the most significant parts of the implementation. Then we will focus on the main issues considered during the development of the project, comparing the adopted solutions with those of other state-of-the-art packages like Squid [1]. Finally we will discuss new ways of improving Prometeo's performances and scalability.[2]

1 Introduction

Proxies are important components of large, heterogeneous networks: they're often found on the frontier of private LAN's or corporate networks to allow their users to access Internet resources in a controlled manner - for instance, forcing them to obey to corporate policy. Caching proxies also help to optimize the available resources, reducing the traffic generated by the users. Another class of proxies enables interoperability between applications, translating on the fly from one protocol to another (for example, from NNTP to POP3). Proxies can also be used for special purposes, for instance to allow visually challenged people to browse the web [2] or to improve the management of networked games [3].

2 Motivations

There are plenty of proxies for almost every service, thus one may wonder where the need of another product comes from. Problem is, the vast majority of the

[1] Available under the GPL license on sourceforge. See http://prometeo-proxy.sourceforge.net/

[2] This work has been partially supported by the FIRB project named WebMinds of the Italian Ministry for Education, University and Research and by the 6NET E.U. project.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 527–534, 2005.

existing packages were aimed at solving a specific goal such as providing a certain service (e.g. http) or adding a special or missing feature to an existing client-server setup (e.g. stripping banners from web pages before feeding them to the browser, or adding TLS/SSL encryption). This implies that a system administrator who need to setup different proxy services is bound to install several packages, each one of them with its own management rules and its idiosyncrasies.

Prometeo main idea was to provide the administrator with an equivalent of the inetd daemon for proxies: a single application able to serve different kind of services through the use of plug-ins. This solutions gives several benefits:

- it simplifies the administration and maintenance of the proxies: services can be started, stopped, (un)installed or updated independently using the same tool;
- resource optimizations: many common functions are included in the framework which is shared by every module implementing a single service;
- Prometeo's framework lets the developer to focus on the logic of the service she wants to implement, making it quite easy to add new services to the package.

Moreover, we wanted an application which supported IPv6 networks natively and that could help to interconnect IPv4 and IPv6 networks allowing IPv4-only clients to access IPv6 servers or vice-versa.

Table 1 shows a comparison of Prometeo's main features against those of other available applications: as you can see, we tried to gather the most important features of several packages into a single, integrated package.

Table 1. Prometeo features compared to those of other existing applications

Feature		Prometeo	Squid	WWWOFFLE	Apache	Ziproxy	Tinyproxy	SuSE Proxy-Suite	Frox	DeleGate	Stunnel	TLSWRAP	WinGate
Open Source/Free Software		Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	
Modular Design		Y			Y								Y
Easy to extend		Y											n/a
Easy to use		Y				Y	Y		Y			Y	Y
IPv6 support		Y		Y		Y							?
Transparent proxy support		Y	Y				Y	Y	Y				Y
Support for multiple protocols		Y								Y	Y		Y
Remote Administration		Y		Y						Y			Y
DNS cache		Y	Y							Y			Y
FTP proxy	RFC-959 compliance	Y				n/a	n/a	Y	Y	Y	n/a	Y	Y
	SSL/TLS wrapper	Y				n/a	n/a			Y	n/a	Y	
HTTP proxy	HTTP/1.1 support	Y	Y	Y	Y			n/a	n/a	Y	n/a	n/a	Y
	Cache	Y	Y	Y	Y			n/a	n/a	Y	n/a	n/a	Y
	Gzip/deflate compression	Y		Y		Y		n/a	n/a	Y	n/a	n/a	
	Connection cache towards origin servers	Y	Y					n/a	n/a		n/a	n/a	
	Filters	Y	Y	Y				n/a	n/a		n/a	n/a	Y
	Host remapping	Y			Y			n/a	n/a	Y	n/a	n/a	
POP3 proxy	POP3 service	Y								Y			Y
	SpamAssassin support	Y											
SSL Tunneling		Y								Y	Y		Y
TCP Tunneling		Y								Y			Y

Two are the products which mostly share Prometeo's philosophy and goals:

– WinGate, a commercial proxy suite for Microsoft Windows platform developed by Qbik and marketed by Deerfield.com. It's main drawback is that it's a closed-source application and that seemed an unacceptable constraint to us.
– DeleGate, an open-source project started in 1984 by Yutaka Sato and grown thanks to the contribution of many programmers around the world. It too provides a lot of different proxy services in a single product, although it has a monolithic design rather than a modular one. Moreover, it's doesn't support IPv6 at all.

Among the others, we can't help mentioning Squid [1], perhaps the most well- known and widely used Web caching proxy for POSIX systems. Squid has become an industry standard used by many corporations or Internet Service Providers thanks to its robustness and scalability.

3 Design Overview

The core of the system is formed by the Prometeo main executable, which contains the framework shared by the modules: other than the normal housekeeping functions, it offers a centralised access to the configuration, logging, access control, storage access and so on.

The core uses a plugin architecture to offer its services: every plugin implements a proxy for a different protocol or implements additional features - mod_cfg, for instance, offers a web-based control panel, providing a comfortable way of configuring and managing the whole application, other plugins included.

Prometeo provides the asynchronous I/O file and network functions. It adds an abstraction layer between the application logic and the standard BSD-socket interface, encapsulating all the required functions in a set of C++ classes. prometeolib supports IPv6 in a fairly transparent way: the programmer won't have to worry about the differences between IPv4 and IPv6 unless he requires so. This library can also be used in other projects without many changes, making life easier expecially whenever asynchronous operations are required.

prometeoctl is an utility program which allows to administer the system from command-line or from scripts. For a detailed description of the implementation please refer to the technical reports provided in the Prometeo web site.

4 Modularity

All the services offered by Prometeo are implemented as modules. As already mentioned, modules can be independently configured, loaded, unloaded or upgraded. It's also possible to provide specialised services using the same plugin with different configurations on different ports.

In the following sections we'll give a brief description of the most interesting aspects of the currently available modules.

mod_http

Most of the time spent developing Prometeo has been dedicated to this module, since HTTP is indeed the most used protocol [7]. The module implements an HTTP 1.1 [10] caching proxy. It has been designed keeping the following points in mind:

– asynchronous operations: everything is performed in the same process, in an effort to minimize latency and to simplify the access to common resources, such as the cache, since no locking mechanism is required;
– low latency: to guarantee a prompt reply to client requests, the full index of the cache is kept in memory, while the actual cache objects are loaded only when needed;
– bandwidth optimization: if the server or the client supports either gzip or deflate content encoding data is transferred in a compressed format; moreover, concurrent requests for the same resource are satisfied using a single server connection, thus avoiding unnecessary transfers;
– standard compliance: mod_http tries to comply with all the recommendations found in [10] regarding proxy servers: for instance it correctly handles persistent connections separately with its clients and the origin servers, it understand the cache control headers and so on.

Data compression is especially useful when clients are connected to the proxy using a slow link, such as a dialup connection or GPRS [8].

mod_ftp

This module provides an FTP [11] proxy whose strength points are:

– IPv6 support: not only it understands the IPv6 protocol extensions [12], it allows IPv4 clients to access IPv6 servers as well.
– SSL/TLS support [13]: mod_ftp is optionally able to secure communications with origin servers, even if the client doesn't support SSL/TLS. It's also possible to disallow access to unsecured origin servers.

mod_cfg

Unlike the other modules, mod_cfg doesn't implement a proxy. Instead, it offers a web-based configuration interface which can be used to configure the whole Prometeo application, including the other modules.

In fact, every module provides an XML fragment describing its configuration options. mod_cfg exploits these information to build an user-friendly interface on-the-fly.

5 Extensibility

A key aspect of Prometeo's design is extensibility: adding a new service is a very simple job. The framework already deals with most of the low-level or tedious parts which are needed by any proxy, such as logging, authentication or caring about IPv4/IPv6 differences.

The programmer should only need to focus on the logic of the proxy she needs. Looking at the source code will show how the code of the modules is simple. For instance, mod_ftp is just about 1500 lines of code (comments included), against the 4600 lines of SuSE Proxy Suite.

mod_http is just 4500 lines of code, which is not bad considering that WWWOffle [15] is made up of 29000 lines of code offering more or less a comparable range of features of Prometeo (although Prometeo is more scalable). As a side note, Squid is about 56000 lines of code, but it offers a number of features (SNMP, ICP...) still not comparable to mod_http.

6 Resource Optimization

Caching the connections
: mod_http is able to cache connections: HTTP/1.1 persistent connections are always requested when dealing with an origin server. After receiving the requested resource, the idle connection is kept in a connection cache for a limited amount of time (15 seconds in the current implementation, as the default timeout for idle connections in the Apache web server). During this time span, when a client requests another resource from the same server, it will be served using an already established connection. As noted in [6], connection caches can be very useful to reduce latency.

Requests aggregation
: When mod_http receives a new request for a resource that is already been transfered from an origin server to a client, it will serve it without establishing a second connection to the server: it will immediately send the data available to the client and as soon as new data arrivers, it will be forwarded to all the registered clients. The implementation in Prometeo is similar to that of Squid. Adding requests aggregation to an HTTP proxy proved to be a simple task, yet very useful to save on connections especially when there's a peak of requests for a popular resource.

Processes cache
: Prometeo uses for many of its modules (such as mod_ftp, mod_ssl, mod_pop3...) a cache of child processes which can be used to process an incoming request. The idea has been inspired by the behaviour of the Apache server [14], although Prometeo's framework offers a generalized implementation which helps to create custom process caches very easily. A processes cache helps to reduce the overhead associated with the creation of a new child and the setup of an IPC channel with the parent and it's particularly effective when the proxy is been stressed. Another interesting aspect of using a process cache is that if a child process crashes, it's damages to the rest of the system will be limited, in the general case. A crashed process will be replaced with a newly spawned one as soon as it's needed.

7 About Scalability

Initial tests have shown that mod_http performs as well as Squid in terms of latency and throughput (requests per second). More accurate benchmarks are required to give a final judgement. Still, we've noted some interesting points:

- logging requests to file inflicts a considerable penalty both in terms of latency and throughput. Disabling the log, Prometeo has been able to serve more than twice the number of requests served when logging is active with less than half of the latency.
- the use of persistent connections and the connection cache seems to have a great impact on throughput, while it is negligible on latency. The tests seem to be in line with [6] findings, showing a 4x increase in the number of requests satisfied per second.

Exploiting multiple CPU's
: Currently, both Prometeo's mod_http and Squid operate using a single process. The main advantage of this solution is that it doesn't require locking mechanisms to access the cache, thus simplifying the code. On the other hand, a single process can't receive any benefit being run on a SMP machine. Apart from mod_http, the other modules of Prometeo are implemented using a process per client. This way, each concurrent process can be scheduled on a different processor.

Hierarchical caching
: One of the biggest scalability advantages of Squid over Prometeo is the support for hierarchical caches: if your proxy does not have an object on disk, its default action is to connect to the origin web server and retrieve the page. In a hierarchy, your proxy can communicate with other proxies (in the hope that one of these servers will have the relevant page). In large networks, a single proxy server may not be enough to satisfy all the requests. Adding more servers helps to keep the individual load under control, while increasing the overall number of clients that can be served at once. Squid efficiently implements different inter-cache communication protocols, thus being able to maintain latency at a low level even with big cache hierarchies. Recently new approaches to the scalability problem of web proxies have been proposed [9]. It would be worth to test how well these proposals behave in a real environment.

8 Use Cases

In this section we'll see a couple of example showing how Prometeo is currently being used reporting the experiences of its users.

Interconnecting IPv4 and IPv6 networks
: At the CS Department of University of Bologna, Prometeo has been used for more than 5 months as default proxy for the IPv6 research project, proving

to be robust and reliable. The proxy is connected to the Internet and to the 6bone network. Its main goal is to make accessible any server, be it on the IPv4 or IPv6 network, to any client. Also, if there were the need of setting up some web servers on native IPv6 hosts, Prometeo could be used to make them accessible from the Internet acting as a front-end server and mapping the requests to the correct machine.

Making use of special features
A feature that has encountered a good success among the users is the support for Spam Assassin which has been integrated into mod_pop3. The module implements a simple POP3 proxy which optionally can filter emails through Spam Assassin's spamd daemon: this feature is mostly useful to those users which cannot change the mail server setup because it's not under their control and thus cannot filter spam as it arrives. Also, the support for transparent proxy[3] makes mod_pop3 use very comfortable, since it doesn't require changes in the clients' configuration. An ISP is currently evaluating the use of this module to offer a spam filtering service to some of its customers without having to modify the rest of its mail servers setup.

9 Improvement Ideas

We reckon that some aspects of Prometeo are not optimal, mostly due to the strict time constraint which have condition the development process.

We propose some ideas which would improve Prometeo's performances on large networks or add interesting features.

Replacement policies
As described in [5], LRU is not the optimal replacement policy for a caching web proxy. It should be fairly easy to implement the proposed LRV policy in mod_http.

RAM-based cache
Nowadays servers can be fitted with huge amounts of memory for a relatively small price. For instance, a dedicated proxy machine with 4 GB of memory would gain a lot in terms of performances if it could avoid using the disks. At the same time, 4 GB should be more than enough to host a web caching proxy for a large network or several kind of proxy for a smaller one. Prometeo can be easily modified to use memory for all its storage needs: it's only necessary to rewrite the Storage and StoreObj classes. The only drawback with placing the cache in memory is that a machine reboot would reset the cache. We believe that the benefits are worth the risk though, also considering that reboots should be rare and mostly due to hardware problems, given that Prometeo has reached a satisfying level of stability.

[3] At the moment of writing, transparent proxy and IPv6 support are mutually exclusive on GNU/Linux.

Differentiated/dynamic filters

mod_http could enable different filters according to the type of the client which submits the requests. For instance, if the client is using a narrow link, as in the GPRS case, it could scale down or resample images to save bandwidth. This could be implemented using proxy-authentication and a database of users' preferences.

Load balancing module

An interesting module that could be added to Prometeo is a software load balancer. It could be useful to use Prometeo as a front-end to a group of servers which need to sustain very high loads.

References

1. Squid Internet Object Cache, [online] http://www.squid-cache.org/
2. Hironobu Takagi, Chieko Asakawa, "Transcoding proxy for non visual web access", *Proceedings of the fourth international ACM conference on Assistive Technologies*, 2000, pages 164-171
3. Martin Mauve, Stefan Fischer, Jörg Widmer, "A generic proxy system for networked computer games", *Proceedings of the first workshop on Network and system support for games*, 2002, pages 25-28
4. J. C. Mogul, "Speedier Squid: A case study of an Internet server performance problem", *Login: The USENIX Association Magazine*, 1999, vol. 24, no. 1, pages 50-58
5. Luigi Rizzo, Lorenzo Vicisiano, "Replacement Policies for a Proxy Cache", *IEEE Transactions on networking*, April 2000, vol. 8, no. 2, pages 158-170
6. Ramón Cáceres, Fred Douglis, Anja Feldmann, Gideon Glass, Michael Rabinovich, "Web proxy caching: the devil is in the details", *ACM SIGMETRICS Performance Evaluation Review*, December 1998, vol. 26, issue 3
7. Martin Arlitt, Rich Friedrich, Tai Jin, "Workload Characterization of a Web Proxy in a Cable Modem Environment", *Proceedings of the eleventh international conference on World Wide Web*, May 2002, pages 25-36
8. Bruce Zenel, Dan Duchamp, "A General Purpose Proxy Filtering Mechanism Applied to the Mobile Environment", *Proceedings of the third annual ACM/IEEE international conference on Mobile computing and networking*, 1997, pages 248-259
9. Markus J. Kaiser, Kwok Ching Tsui, Jiming Liu, "Self-organized Autonomous Web Proxies", AAMAS'02, 2002, pages 1397-1404
10. R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, "Hypertext Transfer Protocol - HTTP/1.1", *RFC 2616*, June 1999
11. J. Postel, J. Reynolds, "File Transfer Protocol (FTP)", *RFC 959*, October 1985
12. M. Allman, S. Ostermann, C. Metz, "FTP Extensions for IPv6 and NATs", *RFC 2428*, September 1998
13. Ford-Hutchinson, Carpenter, Hudson, Murray & Wiegand, "Securing FTP with TLS", Draft 09, April 2002, [online] http://www.ford-hutchinson.com/~fh-1-pfh/ftps-ext.html
14. Apache Software Foundation, [online] http://httpd.apache.org
15. Andrew M. Bishop, [online] http://www.gedanken.demon.co.uk/wwwoffle/index.html

Pass Down Class-LRU Caching Algorithm for WWW Proxies

Rachid El Abdouni Khayari

University of the Armed Forces Munich, Germany
Department of Computer Science

Abstract. Caching has been recognized as one of the most important techniques to reduce Internet bandwidth consumption caused by the tremendous growth of the WWW. Class-based LRU (C-LRU) delivers better results for the hit rate than most of the existing strategies, sharing the best and second place with GDS-Hit which provides better results for small cache sizes, since in this case cache places may still be unused. We study an extension of the C-LRU caching algorithm, namely Pass-Down-C-LRU (PD-C-LRU), to exploit these unused cache places for small cache sizes. We have found that the filling degrees of the classes affect the performance results for both hit rate as for the byte hit rate.

Keywords: Caching, Class-based LRU, World Wide Web, Proxy Server.

1 Introduction

Today, the largest share of traffic in the Internet originates from WWW requests. The increasing use of WWW-based services has not only led to high frequented web servers but also to heavily-used components of the Internet. Fortunately, it is well known that there are popular and frequently requested sites, so that object caching can be employed to reduce Internet network traffic [6] and to decrease the perceived end-to-end delays. When a request is satisfied by a cache, the content no longer has to travel across the Internet from the origin web server, saving bandwidth for the cache owner as well as the originating server. Web caching is similar to memory system caching, in that a cache stores web pages in anticipation of future requests. However, significant differences between memory system and web caching result from the non-uniformity of web object sizes, retrieval costs, and cacheability. In case the considered objects have the same size it is simple to find the optimal caching algorithm if knowledge of the future is available. The Belady's method can be used to choose the optimal algorithm [4]. Unfortunately, in web caching the object sizes vary, so that the problem to determine the optimal caching strategy becomes NP-hard [7].

Over the last few years, many well-known caching strategies have been evaluated [1, 5, 11, 14]. Aim of these strategies has been to improve the cache hit rate (defined as the percentage of requests that could be served from the cache), the cache byte hit rate (defined as the percentage of bytes that could be served

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 535–543, 2005.

from the cache), or, even better, both. At the center of all the approaches is the question which object has to be replaced when a new object has to be stored (and the cached is already completely filled). There are three properties of WWW requests (i.e. references), namely frequency of reference, recency of reference and the referenced object size, that are typically exploited in diverse ways to determine the document popularity. These algorithms have been developed for specific contexts (e.g., for memory, web, disk or database caching) and it has been shown that an algorithm that is optimal for one context may fail to provide good results in another context [1, 5]. This is due to the fact that the caching algorithms rate some object characteristics more important than others.

The caching strategy *class-based LRU* (C-LRU) is a refinement of standard LRU. Its justification lies in the fact that object-size distributions in the WWW are heavy-tailed, that is, although small objects are more popular and are requested more frequently, large objects occur more often than it has been expected in the past, and therefore have a great impact on the perceived performance. The C-LRU caching algorithm works size-based as well as frequency based. In most caching methods, the object sizes are completely ignored, or either small or large objects are favored. However, since caching large objects increases the byte hit rate and decreases the hit rate (and vice versa for small objects), both a high byte hit rate and a high hit rate can only be attained by creating a proper balance between large and small objects in the cache. With C-LRU, this is achieved by proportioning the cache into portions reserved for objects of a specific size, as follows: The available memory for the cache is divided into I partitions where each partition i (for $i = 1, \ldots, I$) takes a specific fraction p_i of the cache ($0 < p_i < 1, \sum_i p_i = 1$). The partition i caches objects belonging to class i, where class i is defined to encompass all objects of size s with $r_{i-1} \leq s < r_i$ ($0 = r_0 < r_1 < \ldots < r_{I-1} < r_I = \infty$). Each partition in itself is managed with the LRU strategy. Thus, when an object has to be cached, its class has to be determined before it is passed to the corresponding partition. For this strategy to work, we need an approach to determine the values $p_1, \ldots, p_I$ and $r_1, \ldots, r_I$. This will be addressed in the next section.

This paper is organized as follows. In Section 2, we will first present the rationale behind class-based LRU caching algorithm, present the found results, and discuss the issue of potentially unused caches by small cache sizes. In Section 3, we introduce and validate an extension of the C-LRU, namely the PD-C-LRU, to avoid unused cache places. Finally, the paper is concluded in Section 4.

2 C-LRU: Characteristics and Application

As has been shown in [8], the object-size distribution of objects requested at proxy servers, can very well be described as a hyper-exponential distribution; the parameters of such a hyper-exponential distribution can be estimated easily with the EM-algorithm. This implies that the object-sizes density $f(x)$ takes the form of a probabilistic mixture of exponential terms: $f(x) = \sum_{i=1}^{I} c_i \lambda_i e^{-\lambda_i x}$, with $\sum_{i=1}^{I} c_i = 1$, and $0 \leq c_i \leq 1$, for $i = 1, \ldots, I$. This can now be interpreted as follows: the weights c_i indicate the frequency of occurrence for objects of class i and the average size of objects in class i is given by $1/\lambda_i$. For the fraction p_i, we propose two possible values: (a) to optimize the hit rate, we take the partition size p_i proportional to the probability that a request refers to an object from class i, that is: $p_i = c_i$; or (b) to optimize the byte hit rate, we take into account the expected amount of bytes "encompassed by" class i in relation to the overall expected amount of bytes. Since the average object size in class i is $1/\lambda_i$, we set: $p_i = \frac{c_i/\lambda_i}{\sum_{j=1}^{I} c_j/\lambda_j}$. The range boundaries r_i are computed using Bayesian decision (see [9]): $r_i = \frac{\ln(c_i\lambda_i)-\ln(c_{i+1}\lambda_{i+1})}{\lambda_i-\lambda_{i+1}}$ for $i = 1, \ldots, I-1$.

Since the C-LRU approach exploits characteristics of the requested objects, it is important to have an accurate characterization of the requested objects. Hence, next to the question how one should characterize the objects, we have to determine how often one has to adapt the characterization. There are three possibilities [9]: Once-only determination, periodical application and (c) application on-demand. In our analysis here, we will focus on the the first method, namely the once-only determination. A full investigation of the above adaptation strategies and their performance implications goes beyond the scope of the current work.

Application and evaluation: To evaluate and compare the performance of C-LRU, trace-driven simulations have been performed. The RWTH trace has been collected in early 2000 and consists of the logged requests to the proxy-server of the Technical University of Aachen, Germany. First, we will start with a detailed analysis of the trace, before we continue with a detailed comparison study. We finish with a discussion of the complexity of the caching algorithm.

Statistics of the trace: In our analysis, we only considered static (cacheable) objects, requests to dynamic objects were removed as far as identified. Table 1 presents some important statistics for the RWTH trace. The heavy-tailedness of the object-size distribution is clearly visible: high squared coefficients of variation and very small medians (compared to the means). The maximum reachable hit

Table 1. Statistics for the RWTH trace

total #requests	32,341,063
total #bytes	353.27 GB
#*cacheable* request	26,329,276
#*cacheable* bytes	277.25 GB
fraction #*cacheable* requests	81.4 %
total #*cacheable* bytes	78.5 %
average object size	10,529 Bytes
squared coeff. of variation	373.54
median	3,761 Bytes
smallest object	118 Bytes
largest object	228.9 MB
unique objects	8,398,821
total size of unique objects	157.31 GB
HR∞	30.46 %
BHR∞	16.01 %
original size of trace file	2 GB
size after preprocessing	340 MB

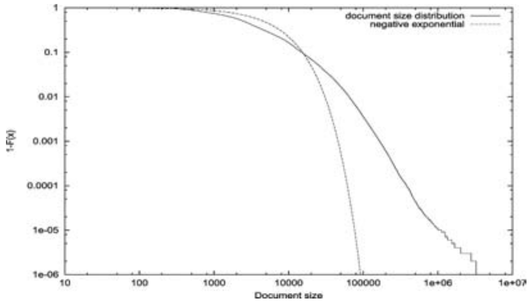


Fig. 1. Complementary log-log plot of document size distribution

rate (denoted as HR_∞) and the maximum reachable byte hit rate (BHR_∞) have been computed using a trace-based simulation with infinite cache.

Distribution of the object sizes: Figure 1 shows the complementary log-log plot of the object-size distribution. As can be seen, this distribution decays more slowly than an exponential distribution, thus showing heavy-tailedness. This becomes even more clear from the histogram of object sizes in Figure 2.

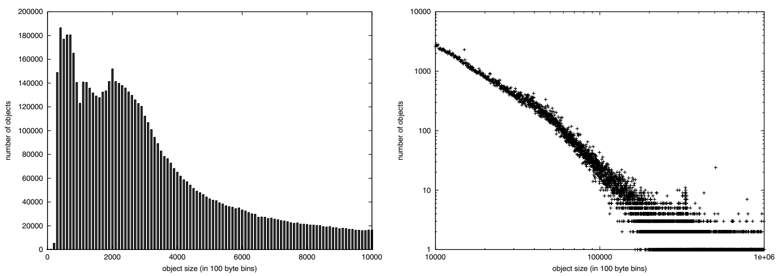


Fig. 2. Number of objects as function of objects size: (left) linear scale for objects smaller than 10 KB; (right) log-log scale for objects larger than 10 KB

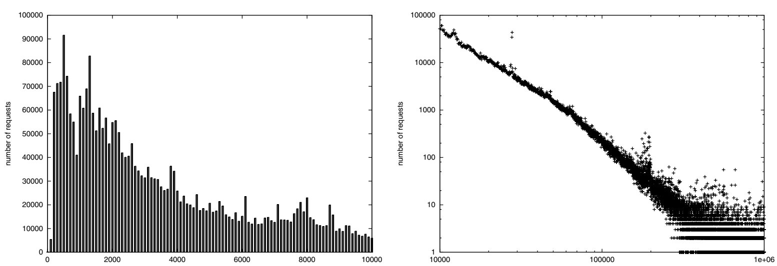


Fig. 3. Number of requests by object size: (left) linear scale for objects smaller than 10 KB; (right) log-log scale for objects larger than 10 KB

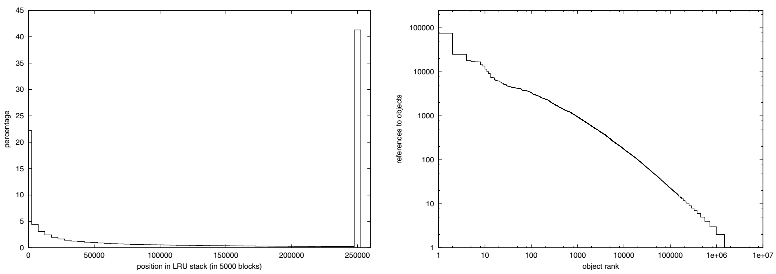


Fig. 4. Analysis of the trace: (left) temporal locality characteristics (LRU stack-depth); (right) frequency of reference as a function of object rank (Zipf's law)

The heavy-tailedness is also present when looking at the request frequency as a function of the object size (Figure 3). It shows that small objects are not only more numerous but also that they are requested more often than large objects (this inverse correlation between file size and file popularity has also been stated in [13]). Thus, caching strategies which favor small objects are supposed to perform better. However, the figure also shows that large objects cannot be neglected.

Recency of reference (temporal locality): Another way to determine the popularity of objects is the temporal locality of their references [2]. However, recent tests have pointed out that this property decreases [3], possibly due to client caching. We performed the common LRU stack-depth [2] method to analyze the temporal locality of references. The results are given in Figure 4. The positions of the requested objects within the LRU stack are combined in 5000 blocks. The figure shows that about 20% of all requests have a strong temporal locality, thus suggesting the use of a recency-based caching strategy.

Frequency of reference: Object which have been often requested in the past, are probably popular for the future too. This is explained by Zipf's law: if one ranks the popularity of words in a given text (denoted ρ) by their frequency of use (denoted P), then it holds $P \sim 1/\rho$. Studies have shown that Zipf's law also holds for WWW objects. Figure 4 shows a log-log plot of all 8.3 million requested objects of the trace. As can be seen the slope of the log-log plot is nearly -1, as predicted by Zipf's law, suggesting the use of frequency-based strategies. It should be mentioned that there are many objects which have been requested only once, namely 67.5% of all objects. Frequency-based strategies have the advantage that "one timers" are poorly valued, so that frequently requested objects stay longer in the cache and cache pollution can be avoided.

We performed the trace-driven simulations using our own simulator, written in C++. To obtain reasonable results for the hit rate and the byte hit rate, the simulator has to run for a certain amount of time without hits or misses being counted. The so-called *warm-up* phase was set to 8% of all requests, which corresponds to two million requests and a time period of approximately four days. In the simulator, well-known caching algorithms have been implemented;

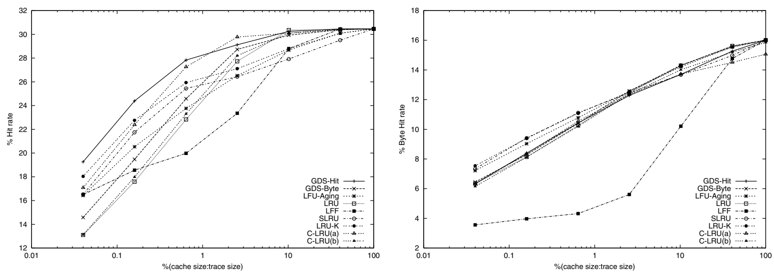


Fig. 5. Hit rate and byte hit rate comparison of the caching strategies

the description of these used caching algorithms can be found in [9, 10, 12]. The cache size is a decisive factor for the performance of the cache, hence, we want to choose the caching strategy that provides the best result for a given cache size. To compare the caching strategies, we have performed the evaluation with different cache sizes. In Figure 5, we show the simulation results of the RWTH trace for the different caching strategies with respect to the hit rate, respectively, the byte hit rate. With respect to the hit rate, the simulations show that GDS-Hit provides the best performance for smaller cache sizes. However, for larger cache sizes, it is outperformed by C-LRU(a). In practical use, the problem of small cache sizes does not pose a problem since typical caches nowadays are larger than 1 GBytes and, indeed, C-LRU performs well for those cache sizes. For the byte hit rate, one observes that the performance of all strategies is nearly equal, except for LFF which yields the worst results.

3 Pass-Down C-LRU

As already stated above, C-LRU does not yield the best performance results for small cache sizes (compared to GDS-Hit). The reason for this is the fact that the reserved cache for large documents was not large enough to encompass the assigned files, when we are dealing with small caches (64 MB, 256 MB). For example, at an overall cache size of 64 MB the reserved cache for the fourth class is $0.2\% \cdot 64$ MB= 1.28 MB. Since the fourth class is used to store documents larger than 377 KBytes, many of these documents will not fit at all. This implies that the fourth class might be empty most of the time. An example for a similar situation is given in Figure 6. A new document should be stored in the cache, in the first class. Unfortunately, the first class is completely filled, so that one document has to be removed from it, although there is enough place in the last two classes, class 3 and 4.

The idea of the PD-C-LRU caching strategy can be explained as follows (Figure 7): each partition is divided in a “valid” (dark dashed) and “invalid“ region (whitely dashed). The “valid” part has been used by C-LRU, and caches the documents with appropriate sizes and the “invalid” domain of a class might include documents from all other (higher) classes. Instead of removing the least-

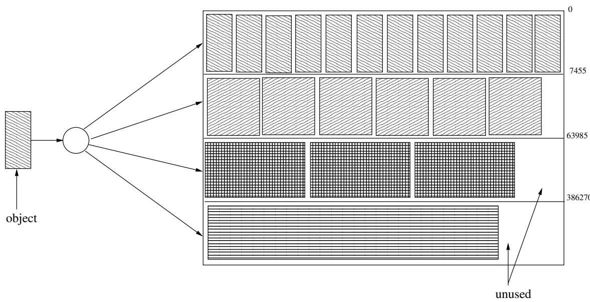


Fig. 6. Unused cache of Class-LRU

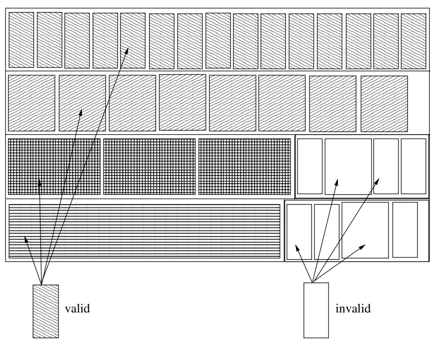


Fig. 7. Pass-down Class-LRU

recently used document from a class, we try to place it in the "invalid" domain of the next class (pass-down); if the cache is completely filled and a new document has to be cached, first, documents from the invalid regions have to be removed, beginning with the last class, bottom-up.

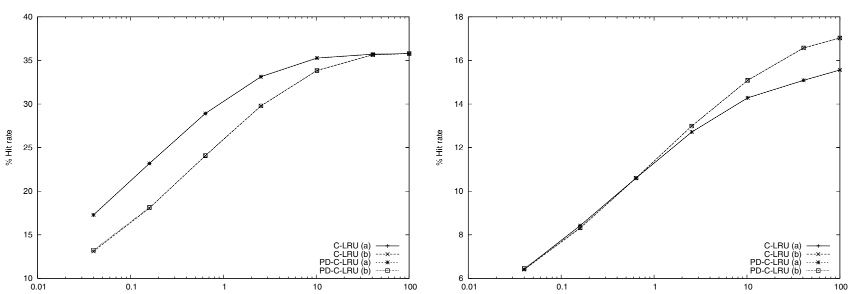


Fig. 8. Comparison of the hit rate and the byte hit rate for the caching algorithms C-LRU and PD-C-LRU

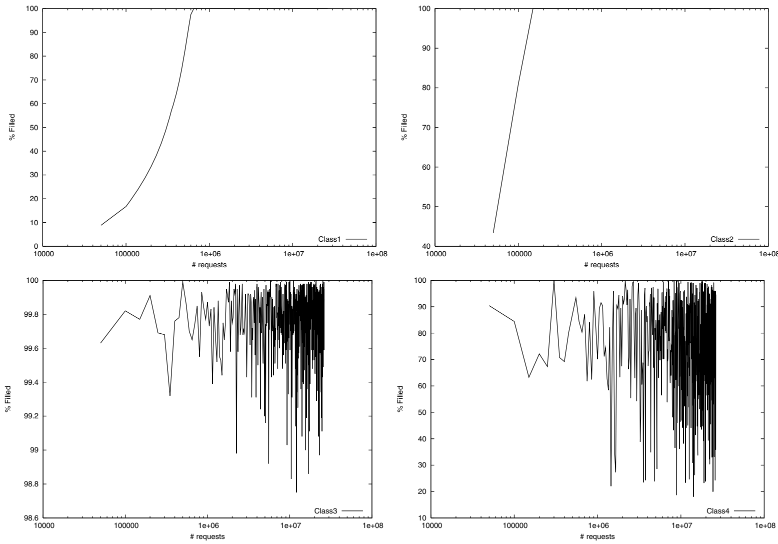


Fig. 9. Filling degree of the classes by the use of C-LRU(a) and a 1GB cache size

We have analyzed PD-C-LRU for the RWTH trace. The obtained results are shown in Figure 8 for the hit rate and the byte hit rate. As we can see, the difference between the two caching algorithms C-LRU and PD-C-LRU is negligible, for both the hit rate and the byte hit rate. The cause of this is the "filling degree" of each class, as can be observed in Figure 9, where the filling degree of class i is defined as the percentage of that class that is yet occupied by documents of suitable sizes. Figure 9 shows the results obtained by the use of the C-LRU(a) and a 1GB cache size. As we can see, after the classes 1 and 2 have been total filled (degree $\sim 100\%$), they will retain this state unchanged. Classes 3 and 4 change their filling degrees so fast that it seems to be unprofitable to use their invalid class region; quickly after documents from other classes have been placed in the invalid domains of the classes, these objects (smaller documents) have to removed already. Similar results have been found for C-LRU(b) and a 4GB cache size; here the filling degrees were higher and roughly 100% for all classes.

4 Conclusions

For the performance of C-LRU, we can make two statements: considering the byte hit rate, its performance is comparable to existing strategies, but when looking at the hit rate, C-LRU is clearly better than most other strategies, sharing the first place with GDS-Hit, depending on cache size. We have also studied an extension of the C-LRU caching algorithm, PD-C-LRU to exploit unused cache places. Experimental work, however, showed that PD-C-LRU does not yield better results than C-LRU. The raison for that is the fast changing of the filling degrees for each class. The benefit that can be attained by PD-C-LRU is at once away, since objects cached ininvalid regions have to be removed from there (in particularly for large caches). This show that the use of frequency based replacement strategies is mandatory in class based caching algorithms. We are investigating the possibility to extend the 'classical' C-LRU to consider the frequency of the requested documents.

References

1. M. F. Arlitt, R. Friedrich, and T. Jin. Workload Characterization of a Web Proxy in a Cable Modem Environment. In *Proceedings of ACM SIGMETRICS*, volume 27, pages 25–36, 1999.
2. M. F. Arlitt and C. L. Williamson. Internet Web Servers: Workload Characterization and Performance Implications. *IEEE/ACM Transactions on Networking*, 5(5):631–645, October 1997.
3. P. Barford, A. Bestavros, A. Bradley, and M. Crovella. Changes in Web Client Access Patterns. *WWW Journal*, 2(1):3–16, 1999.
4. L. A. Belady. A Study of Replacement Algorithms for Virtual Storage Computers. *IBM Systems Journal*, 5:78–101, 1966.
5. P. Cao and S. Irani. Cost-aware WWW Proxy Caching Algorithms. In *Proceedings of USENIX*, pages 193–206, Monterey, CA, December 1997.
6. J. Gettys, T. Berners-Lee, and H. F. Nielsen. Replication and Caching Position Statement. http://www.w3.org/Propagation/activity.html, August 1997.
7. S. Irani. Page Replacement with Multi-Size Pages and Applications to Web Caching. In *Proceedings of the 29th Annual ACM Symposium on the Theory of Computing*, pages 701–710, El Paso, Texas, United States, May 1997. ACM Press, New York, NY, USA.
8. R. El Abdouni Khayari, R. Sadre, and B. Haverkort. Fitting World-Wide Web Request Traces with the EM-Algorithm. *Performance Evaluation*, 52(2–3):175–191, April 2003.
9. R. El Abdouni Khayari, R. Sadre, and B.R. Haverkort. A Class-Based Least-Recently Used Caching Algori thm for WWW Proxies. In *Proceedings of the 2nd Polish-German Teletraffic Symposium*, pages 295–306, Gdansk, Poland, September 2002.
10. Rachid El Abdouni Khayari. *Workload-Driven Design and Evaluation of Web-Based Systems*. PhD thesis, RWTH Aachen, Technical University of Aachen, Germany, February 2003.
11. P. Lorenzetti and L. Rizzo. Replacement Policies for a Proxy Cache. *IEEE/ACM Transactions on Networking*, 8(2):158–170, 2000.
12. S. V. Nagaraj. *Web Caching and Its Applications*. Kluver Academic Publishers, 2004.
13. J. Robinson and M. Devrakonda. Data Cache Management Using Frequency-Based Replacement. In *Proceedings of ACM SIGMETRICS*, pages 134–142, Boulder, May 1990.
14. S. Williams, M. Abrams, C. R. Standridge, G. Abdulla, and E. A. Fox. Removal Policies in Network Caches for World-Wide Web Documents. In *Proceedings of the ACM SIGCOMM Conference*, pages 293–305. ACM Press, August 1996.

Delay Estimation Method for N-tier Architecture

Shinji Kikuchi, Ken Yokoyama, and Akira Takeyama

Fujitsu Laboratories Limited, 4-1-1 Kamikodanaka, Nakahara-ku,
Kawasaki, Kanagawa 211-8588, Japan
{skikuchi, ken-yokoyama, takeyama}@jp.fujitsu.com

Abstract. These days, the majority of large systems serving large numbers of users prefer N-tier architecture because of its scalability and flexibility. To maintain the quality of service in these systems, we need to understand how much delay is generated in each tier. However, the structure and the behavior of this architecture is so complicated that it is difficult to analyze these delays without installing special software or hardware – improvements that might change the server's behavior or result in significant additional costs. To solve these problems, we developed a practical method for estimating the delays generated in each tier of N-tier architecture that uses only easily obtainable parameters. In this paper, we first discuss what these easily obtainable parameters are. We then construct a performance model of the N-tier architecture using these parameters and analyze the delays in the model. Finally, we describe the experiments we conducted to evaluate our approach.

1 Introduction

Because Internet use continues to grow rapidly, systems providing services to huge numbers of users must be able to handle the ever-increasing and varied user requests. To cope with such requests in the shortest amount of time, most large-scale systems, such as major Internet data centers or the internal networks of large companies, use N-tier architecture.

One of the most common examples of N-tier architecture is the 3-tier architecture described in Fig. 1. It consists of a Web server tier that serves as a front-end to users, an application server tier that processes dynamic requests from users, and a database server tier that stores data to be accessed by users.

In N-tier architecture, the servers that share the same role make up a tier and user requests are distributed to them in order to achieve scalability. Tiers that have individual roles are connected and cooperate with each other to handle complicated user requests. Although N-tier architecture is scalable and flexible, it is difficult to readily analyze its performance or behavior because such systems are spread across multiple servers.

For example, the response time for user requests and delays generated in each tier are very important factors for determining service quality. The two most common methods for analyzing the delays generated in each tier are (1) installing special agents in the servers, and (2) inserting packet capture machinery in the networks to

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 544–553, 2005.

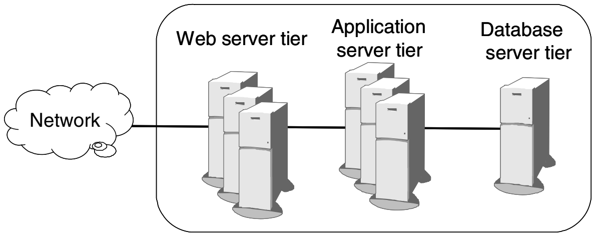


Fig. 1. 3-tier architecture

monitor the packet flow. However, there are many cases where we are unable to apply these methods due to the high cost of packet capture machinery or when special agents cause a change in the behavior of the server. For these reasons, many network managers are extremely reluctant to apply them in their servers or networks.

To solve this problem, we propose a practical delay estimation method that uses very handy and easily obtainable parameters. This method enables us to estimate the average delay generated in each tier of N-tier architecture without installing any special software or hardware in the target node or network.

The rest of paper is organized as follows. In Section 2, we will identify the easily obtainable parameters and select the parameters most appropriate for our approach. In Section 3, we construct a model that can be used to represent the performance of N-tier architecture using the parameters selected in the previous section, and then determine how to estimate delays by analyzing the performance model. In Section 4, we conduct an experiment to evaluate the accuracy of our approach, before concluding in Section 5.

2 What Easily Obtainable Parameters?

There are many possible parameters that can be used to determine a server's behavior. However, the parameters that are most suitable for use in analysis depend on the methods available for collecting information from the system. Since our goal is the development of a practical approach, we need to carefully select the parameters to be used for analysis and the methods for collecting them. Here, we will show and discuss the practicability of four common methods of collecting data representing the servers' behavior.

(1) Modify the application on the servers or install special monitoring agents

One of the most powerful methods of obtaining information from servers we wish to monitor, because it enables us to collect any data we want, is to directly modify the application installed on the servers or install a special monitoring agent. For example, when IBM WebSphere [1] is installed on servers, it places identification tags on data as it is transmitted and calculates delays directly from the behavior of the tagged data.

(2) Capture packets traveling the network
Another approach for analyzing the response time is packet capturing, which records all packets transmitted by each server. Like ENMA [2], if you can capture all packets transmitted by a server, you can estimate any delays occurring in the server without changing the behavior of the system.

(3) Use very common tools
For example, a server like UNIX has many common and useful tools for monitoring the server's resource utilization such as `sar`, `mpstat`, and `iostat`. We can collect basic information, such as CPU or I/O utilization rates using these tools. You can use them even if you are not an administrator. Therefore, these tools are very handy.

(4) Analyze application log files
Many applications such as the Apache web server [3] record information, such as access records, in their log files. We might be able to better comprehend server behavior by analyzing these files.

From the viewpoint of delay measurement accuracy in N-tier architecture, methods (1) and (2) seem to have the strongest advantage since they can measure delays directly. However, from the viewpoint of practicability, there are many cases where we cannot install special software or hardware in the networks we monitor because it might change the system behavior and the owners or the managers of the networks are unwilling to do so. In addition, method (2) requires a packet capture machine. Because this machine needs high-performance processors to process vast numbers of packets in short time periods, as well as enormous storage media to hold the captured data, it is sure to be very expensive. Accordingly, we conclude that direct measurement methods, such as methods (1) and (2) are not always practical. On the other hand, unlike methods (1) and (2), we can collect data using methods (3) and (4), without installing special agents or nodes in the network, – but we cannot measure the delays directly from the parameters collected by these methods. However, if we were able to estimate the N-tier architecture delays from these parameters, it would be a handy and practical analysis method.

Based on this reasoning, we decided to develop a method of estimating the delays generated in each tier in N-tier architecture from easily available information, such as (1) and (2), without installing special agents that might change the behavior of the servers or result in excessive add-on costs.

3 N-tier Architecture Performance Model

3.1 Performance Model Construction

To analyze the performance of a system, we need to construct a model that can represent its characteristics. Since previous research efforts, [4] and [5], showed that a server system could usually be modeled using a queuing system, we decided to use a queuing model as well.

Based on the discussion in the previous section, we constructed the model shown in Fig. 2 to analyze the response time in N-tier architecture, using parameters that can be collected with basic tools and by analyzing the log files. We constructed this model, based on the following assumptions:

- The system consists of tiers with different roles.
  (e.g. Web server tier, application server tier, and database server tier)
- Each tier has one or more servers that have the same role, and user requests are distributed to them equally.
- Each server has one or more CPUs.
- There are some requests at certain rates that leave the system and respond to the users after processing by the server in the n-th tier. For example, requests for static files stored on the web server are processed only by the server in the web server tier and leave the system without being processed by the servers in the other tiers.
- The user requests follow a Poisson distribution that allows us to model the behavior of all servers as M/M/s queuing systems.
- We can collect the CPU utilization data easily using common tools such as `sar` or `mpstat`.
- We can collect the request frequency-per-second data from log files generated by servers, such as Web servers.

Next, the model parameters are explained below:

(1) Static parameters representing an element in the system

N : The number of tiers in the entire system

M_n : The number of servers in the n-th tier

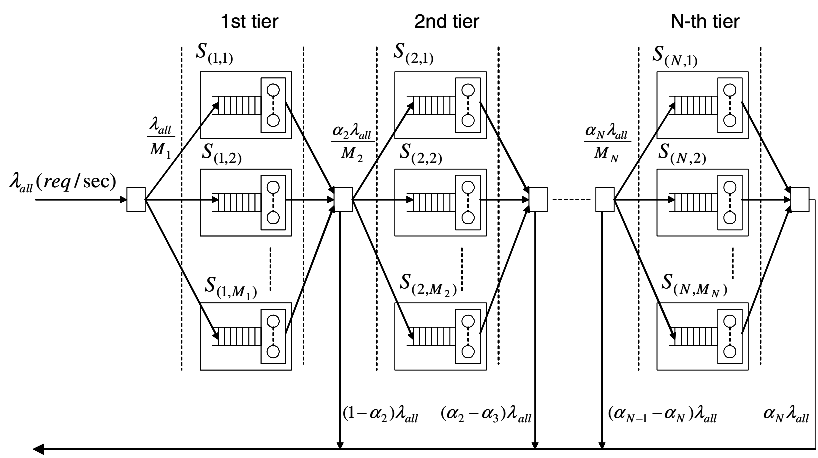


Fig. 2. N-tier architecture model

$S_{(n,m)}$: The index representing the m-th server in the n-th tier $(1 \le n \le N, 1 \le m \le M_n)$
$C_{(n,m)}$: The number of CPUs the $S_{(n,m)}$ has.

(2) Variable parameters that change depending on user requests
λ_{all} : Request rate from users (req/sec)
α_n : The fraction of requests that reach the n-th tier
$(\alpha_1 > \alpha_2 > ... > \alpha_N > \alpha_{N+1}$, and $\alpha_1 = 1, \alpha_{N+1} = 0)$
$\rho_{(n,m)}$: Average CPU utilization rate of the $S_{(n,m)}$ (%)

3.2 Model-Based Delay Analysis

By analyzing this model, we can derive the N-tier architecture delays. First, we derive the delay in a server. Next, we analyze the delay in a tier. Finally, we derive the response time for the entire system.

(1) Server delay analysis

Here, we assume that the server S in Fig. 3 has C CPUs and suppose the request rate to the server is λ, and the average utilization rate of the CPUs is ρ. From the analysis results of the M/M/s queuing model [6], we can derive the average server delay $T(C,\lambda,\rho)$ for user requests as follows:

$$T(C,\lambda,\rho) = F(\lambda,\rho)\, G(C,\rho) \tag{1}$$

$$F(\lambda,\rho) = \frac{\rho}{\lambda} \tag{2}$$

$$G(C,\rho) = \left(\left(\frac{1-\rho}{C^C \rho^C} C! \sum_{i=0}^{C-1} \frac{C^i \rho^i}{i!} + 1\right)(1-\rho) \right)^{-1} + C \tag{3}$$

Equation (2) represents the average CPU time consumed by one request. We suppose this parameter is constant, settled by the clock-speed performance of the server's

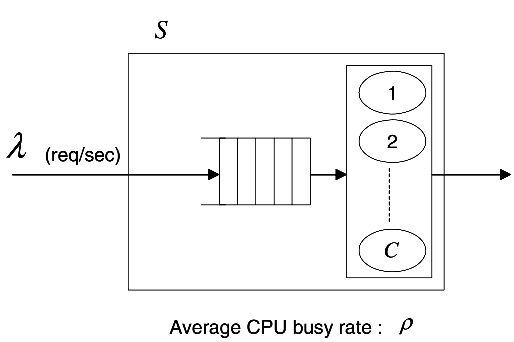


Fig. 3. Single-server model

CPUs. On the other hand, equation (3) represents the degree of increase of the delay, which changes depending on the CPU load.

From equations (1), (2), and (3), we can easily derive equation (4).

$$T(C,\alpha\lambda,\rho)=\frac{1}{\alpha}T(C,\lambda,\rho) \tag{4}$$

(2) Tier delay analysis

We can describe the delay of the m-th server in the n-th tier $S(n,m)$ by $T(C_{(n,m)},\lambda_{(n,m)},\rho_{(n,m)})$. Based on our assumptions that the rate of requests reaching the n-th tier is $\alpha_n\lambda_{all}$ (req/sec) and, the fact that they are distributed to each server in the tier equally, we can say $\lambda_{(n,m)}=\frac{\alpha_n}{M_n}\lambda_{all}$. We can thus derive the average delay W_n that the n-th tier gives to each request going through this tier as

$$W_n=\frac{1}{M_n}\sum_{i=1}^{M_n}T(C_{(n,i)},\lambda_{(n,i)},\rho_{(n,i)})\,. \tag{5}$$

From equations (4) and (5), we can calculate W_n as

$$\begin{aligned}W_n&=\frac{1}{M_n}\sum_{i=1}^{M_n}T(C_{(n,i)},\lambda_{(n,i)},\rho_{(n,i)})\\&=\frac{1}{M_n}\sum_{i=1}^{M_n}T(C_{(n,i)},\frac{\alpha_n}{M_n}\lambda_{all},\rho_{(n,i)})\\&=\frac{1}{\alpha_n}\sum_{i=1}^{M_n}T(C_{(n,i)},\lambda_{all},\rho_{(n,i)}).\end{aligned} \tag{6}$$

For simplicity, we define parameter D_n, representing the delay as

$$D_n=\sum_{i=1}^{M_n}T(C_{(n,i)},\lambda_{all},\rho_{(n,i)})\,. \tag{7}$$

(3) Entire system response time analysis

We can describe the number of requests R_n that leave the system and respond to the users after being processed by the servers in the n-th tier as

$$R_n=(\alpha_n-\alpha_{n+1})\lambda_{all}\qquad(\alpha_1=1,\,\alpha_{N+1}=0)\,. \tag{8}$$

We can also describe the average response time L_n of these requests as

$$L_n=\sum_{i=1}^{n}W_i\,. \tag{9}$$

From equation (9), we derive

$$L_n-L_{n-1}=W_n\,. \tag{10}$$

From these equations, we can calculate the average response time $\hat{X}$ by averaging the response times of all requests entering the system.

$$\begin{aligned}
\hat{X} &= \frac{1}{\lambda_{all}} \sum_{i=1}^{N} R_i L_i \\
&= \sum_{i=1}^{N} (\alpha_i - \alpha_{i+1}) L_i \\
&= (\alpha_1 - \alpha_2) L_1 + (\alpha_2 - \alpha_3) L_2 + \ldots + (\alpha_N - \alpha_{N+1}) L_N \\
&= \alpha_1 L_1 + \alpha_2 (L_2 - L_1) + \ldots + \alpha_N (L_N - L_{N-1}) - \alpha_{N+1} L_N \\
&= \alpha_1 W_1 + \alpha_2 W_2 + \ldots + \alpha_N W_N \\
&= \sum_{n=1}^{N} D_n
\end{aligned} \tag{11}$$

Therefore, from these results, D_n represents the fraction of the average delays generated by the n-th tier. Then, summing them up results in the average response time of the entire system.

One of the noteworthy points of these results is that we can estimate the average response time of all user requests and the magnitude of the effects caused by delays generated in each tier, even if we don't know the values of α_n s, which represent the fraction of the requests that reach the n-th tier.

4 Experiment

4.1 Setup

We evaluated our approach in the experimental network shown in Fig. 4. This is a test-bed system for a personnel service system that can process procedures, such as leave applications or pay statement inquiries. This system is comprised of a web server tier that consistsing of two Web servers using one CPU each, an application server tier with one application server having four CPUs, and a database server tier with one database server having four CPUs. This system is connected to the client using a load balancer in front of the web server tier. We installed Apache in the Web

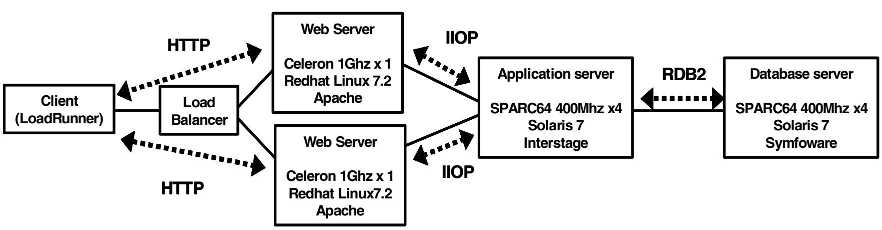


Fig. 4. Experimental network

servers, Fujitsu Interstage [7] in the application server, and Fujitsu Symfoware [8] in the database server.

In this system, when users send requests by HyperText Transfer Protocol (HTTP), the Web servers receive them and send messages by Internet Inter-ORB Protocol (IIOP) to the application server to facilitate cooperation with the server. The application server then accesses the database server using the RDB2 protocol, which is the original protocol for Fujitsu's database system.

4.2 Experiment

We repeatedly sent requests for leave applications from clients using the LoadRunner load generator [9] to the system for 3 minutes. At the same time, we recorded the CPU utilization rate of each server using `sar` command every 10 seconds. In addition, for reference, we captured all packets transmitted by these PCs using an installed `tcpdump` [10] to measure the actual delays generated in each server. After the experiment, we checked the log files of the Apache web servers and calculated the average request rate. From these data and CPU utilization rates, we estimated the delays generated in each tier and compared them with the actual delays calculated from the data collected by packet capture. We performed the same experiment, changing the concurrency of the pseudo clients generated by LoadRunner to 8, 13, 16, and 32.

4.3 Results

Table 1 shows the average CPU utilization rate measured using `sar` command and request rate derived from the Apache web server log files in each experiment. To estimate the delay $T(C_{(n,i)}, \lambda_{all}, \rho_{(n,i)})$ generated by each PC from this data, we first need to calculate $F(\lambda_{all}, \rho_{(n,i)})$ and $G(C_{(n,i)}, \rho_{(n,i)})$. Here, we suppose that $F(\lambda_{all}, \rho_{(n,i)})$ is a static value determined by the clock speed of the PC's CPUs. Therefore, to avoid the fluctuations of $F(\lambda_{all}, \rho_{(n,i)})$ caused by the measured data fluctuations, we derive $F(\lambda_{all}, \rho_{(n,i)})$ using the least mean square approximation to equation (2). As a result, the $F(\lambda_{all}, \rho_{(n,i)})$s values of Web server 1, Web server 2, the application server, and the database server were 0.0417, 0.0337, 0.0174, and 0.0149, respectively. By multiplying these values by $G(C_{(n,i)}, \rho_{(n,i)})$, we calculated the estimated delay in each server $T(C_{(n,i)}, \lambda_{all}, \rho_{(n,i)})$. Finally, by summing them up we estimated the delays D_n generated in each tier.

Table 2 shows a comparison between the delays estimated by our approach and the actual delays directly calculated from the packet capture data collected by `tcpdump`. From the results, we determined that our approach could estimate the delay in each tier correctly, because the differences between the estimated delays and the actual delays are quite small. However, the difference between the estimated delays and the actual delays for the 32 clients, a heavily loaded example, is relatively larger than the differences between these delays in other cases. We need to investigate the cause of this phenomenon and what it means in our future work.

Table 1. Measured request rate and CPU utilization rate

Clients	Request rate (req/sec)	Average CPU utilization rate			
		Web (1)	Web (2)	Appli-cation	Data-base
8	16.3	34.8%	27.5%	42.2%	47.4%
13	24.9	58.8%	32.2%	44.3%	51.3%
16	28.3	49.0%	45.1%	46.7%	31.4%
32	34.4	74.8%	67.2%	54.4%	33.8%

Table 2. Comparison between estimated and actual delays in each tier

Clients	Request rate (req/sec)	Delay in web tier			Delay in application tier			Delay in database tier			Total response time		
		Estimate (msec)	Actual (msec)	Diff. (%)	Estimate (msec)	Actual (msec)	Diff. (%)	Estimate (msec)	Actual (msec)	Diff. (%)	Estimate (msec)	Actual (msec)	Diff. (%)
8	16.3	55.2	49.9	10.7%	72.4	74.7	3.1%	64.0	80.3	20.3%	191.7	204.9	6.5%
13	24.9	75.5	72.4	4.3%	73.1	76.1	3.9%	65.9	71.0	7.1%	214.5	219.5	2.3%
16	28.3	71.6	70.4	1.8%	74.1	86.7	14.6%	60.8	76.2	20.2%	206.5	233.3	11.5%
32	34.4	134.1	185.6	27.7%	78.8	140.2	43.8%	61.3	73.6	16.8%	274.3	399.5	31.3%

5 Conclusions

We developed a practical delay estimation method for estimating N-tier architecture that can estimate the delays generated in each tier. To develop this method, we chose suitable parameters and constructed an N-tier architecture performance model based on those parameters. After analyzing the model, we confirmed the validity of our approach through experiments.

In future research, we are first going to investigate the cause of the estimation errors and develop a method to eliminate them. Then, we plan to evaluate our method in more realistic situations. The experiment described in this paper was performed using a very simple test-bed network. Our future plans require a more realistic test bed on which to perform our experiment. Additionally, we are planning to evaluate our approach in the actual N-tier architecture system of a company network used by many users.

Finally, we believe we can use our performance model for delay estimations and for other performance provisioning areas. For example, by using our model, we think we can estimate the answer to a number of questions, such as how much will the delays decrease if we increase the number of Web servers, or how much will delays increase if the user request rate doubles, or how many servers will we need if we want to maintain the service level stipulated by the customer contract. Accordingly, we plan to develop a system performance provisioning method using our performance model.

Acknowledgements

The authors would like to thank our colleagues for their cooperation in our research. We would especially like to thank Hiroshi Otsuka, who worked diligently to develop

the packet analysis tools used for evaluation and to perform the data collection experiments.

References

1. IBM WebSphere, http://www-306.ibm.com/software/info1/websphere/
2. Y. Nakamura, K. Chinen, H. Sunahara, and S. Yamaguchi: "ENMA: Design and Implementation of the WWW Server Performance Measurement System by Packet Monitoring" (in Japanese), Trans. on communications, D-I, p329-338 (2000)
3. Apache HTTP Server Project, http://www.apache.org/
4. L.P. Slothouber: "A Model of Web Server Performance", in Proc. of 5th Conf. on WWW (1996)
5. Y. Fujita, M. Murata, and H. Miyahara: "Performance Modeling and Evaluation of Web Server Systems", Trans. on communications, B-I, Vol. J82-B, No. 3, p. 347-357 (1999)
6. Y. Yoshioka: "Queuing system and probability distribution", Morikita Shuppan (2004)
7. Fujitsu Interstage, http://interstage.fujitsu.com/
8. Fujitsu Symfoware, http://www.fujitsu.com/services/software/symfoware/
9. Mercury LoadRunner,http://www.mercury.com/us/products/performance-center/loadrunner/
10. tcpdump, http://www.tcpdump.org/

A New Price Mechanism Inducing Peers to Achieve Optimal Welfare*

Ke Zhu, Pei-dong Zhu, and Xi-cheng Lu

School of Computer, National University of Defense Technology, Changsha 410073, China
pigbajie_vivi@hotmail.com

Abstract. Today's Internet is a loose federation of independent network providers, each acting in their own self interest. With the ISPs forming with peer relationship under this economic reality, [1] proves that the total cost of "hot potato" routing is much worse than the optimal cost and then gives a price mechanism—one ISP charges the other a price per unit flow, to prevent this phenomenon. However, with its mechanism the global welfare loss may be arbitrarily high. In this paper we propose a new price mechanism—one ISP charges the other a given fee for different flow scale. With our mechanism, we show that if both ISPs agree on splitting the flow according to the max global welfare the charging ISP will get more profit and the charged ISP will achieve the least cost. And our new mechanism can almost eliminate the welfare loss. Finally, some instances are given and the results show that our mechanism is much more effective than that in [1].

1 Introduction

Traditional analyses of routing in data networks have assumed that the network is owned by a single operator. Typically, the network operator attempts to achieve some overall performance objective—e.g., low average delay or low packet loss rates. Today, it is a network owned by a loosely connected federation of independent network providers. Fundamentally, the objectives of each provider are not necessarily aligned with any global performance objective; rather, each network provider will typically be interested in maximizing their own monetary profits.

To understand the economic incentives driving the actions of network providers, we must first understand the structure of the interconnections they form with each other. Most relationships between two providers may be classified into one of two types: *transit*, and *peer*. Transit is the business relationship whereby one ISP provides (usually sells) access to all destinations in its routing table. Peer is the business relationship whereby ISPs reciprocally provide access to each others' customers [2].

Nowadays, Researchers are keen on peer relationship because it is one of the most important and effective ways for ISPs to improve the efficiency of their operation [2].

* This work is supported by National Sciences Foundations of China (NSFC), under agreement no. 90412011 and 90204005 respectively, National Basic Research Priorities Programme of China, under agreement no.2003CB314802.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 554–560, 2005.

More and more researches have been taken into action [3, 4, 5], but for the first time, [1] quantifies the shortfall in efficiency of the "hot potato" routing deploying in peer relationship. To prevent this shortfall [1] proposes a price mechanism—one ISP charges the other a price per unit flow, to encourage the providers to use network resources efficiently. However, with this mechanism the global welfare loss may be arbitrarily high.

[1] shows that with peer ISPs' cooperation there exists an optimal flow splitting which can minimize the welfare loss to 0. Considering the economic reality, we conclude that a good algorithm should induce both ISPs agree on this flow splitting. In this paper we propose such a new algorithm—damping price mechanism, to prevent the phenomenon of "hot potato" routing. With our mechanism—one ISP charging the other a given fee for different flow scale, our analysis shows that both peer ISPs prefer the optimal flow splitting for more profit. Hence, the mechanism proposed here almost eliminates the global welfare loss.

The rest of the paper is organized as follows. In Section 2, we give the problem arising due to the phenomenon of "hot potato" routing and the price mechanism of [1] in details. In Section 3, we introduce our new highly efficient algorithm—damping price mechanism. The comparison of these two mechanisms is given in Section 4. Finally, we conclude with future works in Section 5.

2 Hot Potato Routing Versus Pricing Routing

Consider a situation where providers S and R are peers. Each of these providers will typically have some amount of traffic to send to each other. However, for the purposes of this paper, we will separate the roles of the two providers as sender and receiver; this will allow us to focus on the different incentives that exist in each role. In particular, we suppose that provider S has some amount of traffic to send to destinations in provider R's network.

2.1 Hot Potato Routing

We assume the only costs incurred are network routing costs, then because the peering relationship includes no transfer of currency, provider *S* has an incentive to force traffic into provider *R* as quickly and cheaply as possible. This phenomenon is known as "nearest exit" or "hot potato" routing (see Marcus [6]).

The optimal routing is to achieve the minimization of the *sum* of the routing costs experienced by the sender and the receiver. When sender and receiver act independently, there is no reason to expect them to arrive at the globally optimal solution, and indeed, this is generally not the case.

In Fig 1, the dashed line depicts the "hot potato" routing passing through Php (which is the nearest point in S to R from s). The solid black line represents the optimal routing passing through Popt.

[1] proves that the "hot potato" routing cost to be no worse than three times the optimal routing cost.

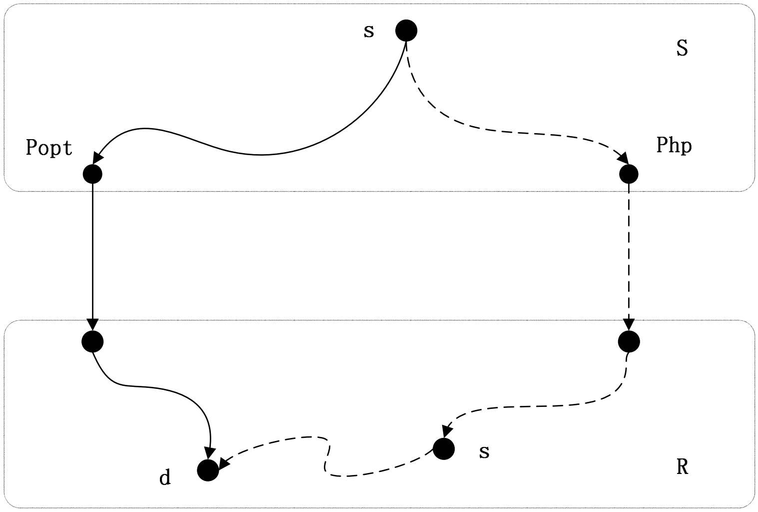


Fig. 1. Hot potato routing cost is at most three times optimal routing cost

2.2 Pricing Routing

[1] investigates the applicability of pricing mechanisms to the peering problem. In Fig 2, [1] assumes that both *S* and *R* consist of a single link connecting two nodes, *s* and *d*. *S* has a total amount of flow *Xs* to send from point $s \in S$ to $d \in R$. Peering points have already been placed at both *s* and *d*. As a result, two routes exist: *S* may choose to either send flow out at *s* to *R*, then use provider *R*'s link to destination *d*; or *S* may use its own link to $d \in S$, then use the peering point at *d* to send traffic to $d \in R$.

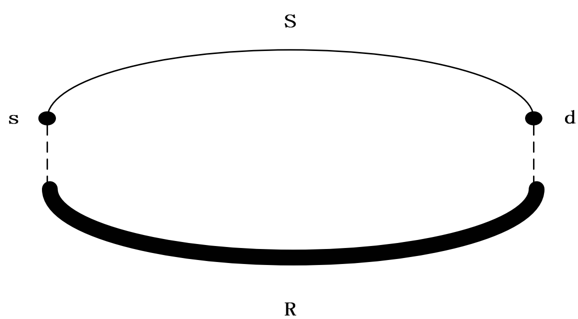


Fig. 2. Provider *S* pays a price per unit flow sent across the link owned by provider *R*

Let f_S and f_R denote the total flow carried by provider *S* and provider *R*, respectively. Assume that *S* has a cost function for the flow on its link, given by $C_S(f_S)$; C_S is assumed strictly convex and strictly increasing with $C_S(0)=0$. C_S has a convex and strictly increasing derivative C'_S with $C'_S(0)=0$. R has a cost function $C_R(f_R)$, which is assumed convex and nondecreasing with $C_R(0)=0$; the derivative C_R is convex and nondecreasing as well, with $C'_R(0)=0$.

The globally optimal routing solution aims to minimize $C_S(f_S) + C_R(f_R)$, subject to $f_S+f_R = X_S$. A simple differentiation establishes that the unique solution to this problem occurs with $C'_S(f_S) = C'_R(f_R)$. Such a point exists since that $C'_S(X_S) > 0 = C'_R(0)$. Denote the globally optimal amounts of flow by f_S* *and* f_R*.

The price mechanism in [1] is setting a price p per unit of flow sent on R's link, then provider S makes a routing decision about how the X_S units of flow will be split between R and S. With price $P(f_R)$, provider S will solve the following problem:

$$\begin{aligned} &\min \quad C_S(f_S) + P(f_R)(f_R) \\ &\textit{subject to} \quad f_S + f_R = X_S \end{aligned}$$

Working with this problem we have

$$P(f_R) = C'_S(X_S - f_R) \tag{1}$$

And the profit maximization problem facing provider R *is* :

$$\max_{f_R \in (0, X_S)} \quad P(f_R) f_R - C_R(f_R) \tag{2}$$

Resolve (2) subject to (1), the solution to (2) is f_R^M and [1] proves that $f_R^M < f_R$*.

Defining $C_S(X_S) - C_S(f_S) - C_R(f_R)$ as welfare. $C_S(X_S) - C_S(f_S*) - C_R(f_R*)$ is defined as the total welfare because the point (f_S*, f_R*) maximize the welfare. [1] gives the bound of the welfare loss to total welfare:

$$\frac{L}{W} \le \frac{\log(\frac{f_R^*}{f_R^M})}{\log(\frac{f_R^*}{f_R^M}) + 1} \tag{3}$$

In the worst case the efficiency loss can be arbitrarily large.

3 Damping Price Mechanism

Considering the analysis of [1], we get two hints:

1. A simple price per unit of flow model can not be high efficient.
2. A good mechanism should induce both peers to split flows as f_R* and f_S*.

With the two hints we develop our Damping price mechanism as follows:

1. While $f_R <= f_R$*, R charges S with the Damping price (*Dp*);
2. While $f_R > f_R$*, R charges S with $k*C_S(X_S)$ $(k > 1)$.

We define the *Dp* as:

$C_S(X_S) - C_S(X_S\text{-}f_R*) - s$ ($s >= 0$,named as seductive coefficient)

Then we will analyze the advantages of this mechanism to S, R and welfare loss, respectively.

As to S, the total cost is

$$C(f)=\begin{cases} C_S(X_S) & f_R=0 \\ kC_S(X_S) & f_R>f_R* \\ Dp+C_S(X_S-f_R)=C_S(X_S)-s & 0<f_R\le f_R* \end{cases}$$

Because $C'(f) = -C'_S(X_S-f_R)<0$ (C_S is a strictly increasing function), $C(f)$ achieve its minimization at the point f_R*, when $f_R \in (0, f_R*]$. Furthermore $C(f_R*)- C_S(X_S) =- s<0$, so S surely wants to send f_R* through R to minimize its cost.

As to R, the total profit is

$$W(f_R)=\begin{cases} 0 & f_R=0 \\ kC_S(X_S)-C_R(f_R) & f_R>f_R* \\ Dp-C_R(f_R) & 0<f_R\le f_R* \end{cases}$$

Because S prefers $f_R<f_R$*, the max profit for R only can be $Dp-C_R(f_R)$. We compare this profit with that $(W_1(f_R)= C'_S(X_S-f_R)f_R-C_R(f_R))$ in[1]:

$$W(f_R)-W_1(f_R) \tag{4}$$

$$\begin{aligned}
&=C_S(X_S)-C_S(X_S-f_R*)-s-C'_S(X_S-f_R)f_R \\
&>C_S(X_S)-C_S(X_S-f_R)-s-C'_S(X_S-f_R)f_R \\
&=C'_S(\zeta_1)f_R-C'_S(X_S-f_R)f_R-s && X_S-f_R<\zeta_1<X_S \\
&=C''_S(\zeta_2)[\zeta_1-(X_S-f_R)]f_R-s && X_S-f_R<\zeta_2<\zeta_1
\end{aligned}$$

Because C_S is a convex function, $C''_S(\zeta_2)>0$, with an appropriate s we can guarantee (4)>0. In other words, with our mechanism R always gets more profit than with that in [1].

As to the welfare loss, our damping price mechanism can induce both S and R prefer the global optimal flow splitting as f_S* and f_R*, so we can almost eliminate it.

4 Mechanisms Comparing

We have shown that our mechanism is much better than [1]'s in welfare loss. In this section by setting the C_S, C_R and X_S we give two instances to compare the R's profit in our mechanism with that in [1].

4.1 Identical Cost Functions

Let $C_S(f_R)=C_R(f_R)=f_R^2$, $X_S=10$. As in section 4 we get $f_R*=5$, $W(f_R*)=50$ and $maxW_1(f_R)=33.33$. In Fig 3 the dashed line depicts the profit $W(f_R)$. The solid black line represents the profit $W_1(f_R)$.

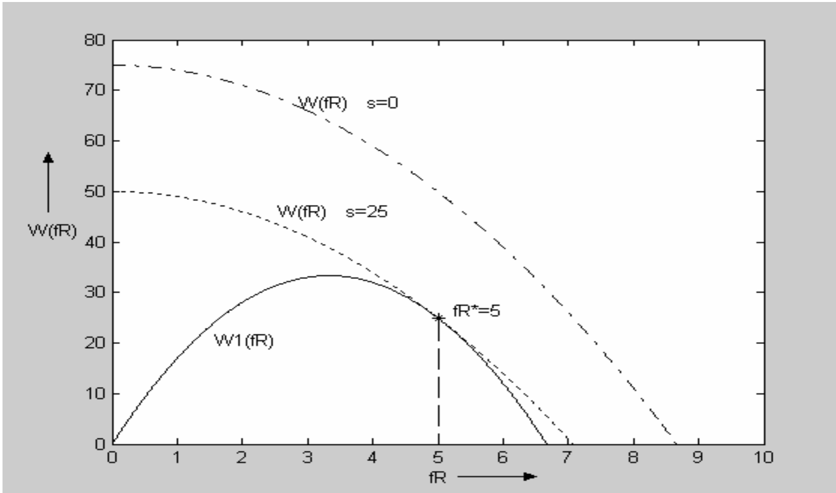


Fig. 3. With an appropriate s $W(f_R)$ is much higher than $W_1(f_R)$

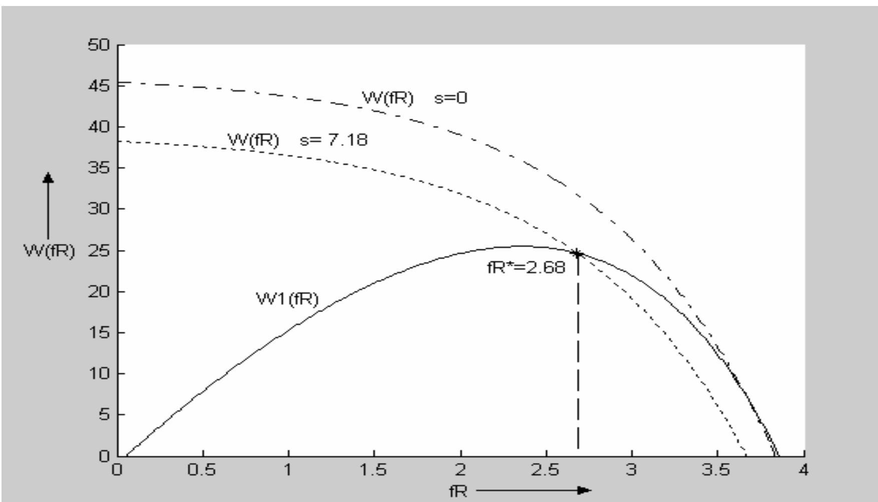


Fig. 4. With an appropriate s $W(f_R)$ is much higher than $W_1(f_R)$

4.2 Different Cost Functions

Let $C_S(f_R)=f_R^2$, $C_R(f_R)=exp(f_R)$, $X_S=10$. As in section 4 we get $f_R*=2.68$, $W(f_R*)=45.42$ and $maxW_1(f_R)=25.47$. In Fig 4 the dashed line depicts the profit $W(f_R)$. The solid black line represents the profit $W_1(f_R)$.

5 Conclusion

Instead of blindly give a price mechanism, our basic approach is to induce both peers to split traffic flow as f_R* and f_S* to eliminate the welfare loss. Considering the ISP's self interest our approach should also satisfy their individual demand. With our damping price mechanism we have successfully achieved our goals.

Future work aims at combining our mechanism with the router hops to solve the route inflation [7, 8] problem.

References

1. Johari, R., and Tsitsiklis, J.N. (2003). Routing and peering in a competitive Internet. LIDS Publication 2570.
2. W. B. Norton. Internet service providers and peering. In Proc. NANOG, June 2000.
3. Awduche, D.O., Agogbua, J., and McManus, J. (1998). An approach to optimal peering between autonomous systems in the Internet. In Proceedings of the 7th International Conference on Computer Communication and Networks, pp. 346-351.
4. Feigenbaum, J., Papadimitriou, C., Sami, R., and Shenker, S. (2002). A BGP-based mechanism for lowest-cost routing. In Proceedings of the 21st Symposium on Principles of Distributed Computing, ACM Press, pp. 173-182.
5. Gopalakrishnan, G., and Hajek, B. (2002). Do greedy autonomous systems make for a sensible Internet? Presented at the Conference on Stochastic Networks, June 19-24, 2002.
6. Marcus, J.S. (1999). Designing Wide Area Networks and Internetworks: A Practical Guide. AddisonWesley.
7. H. Tangmunarunkit, R. Govindan, S. Shenker. Internet Path Inflation Due to Policy Routing. Proceeding of SPIE ITCom 2001, Denver, CO, 19-24, August 2001.
8. Hongsuda Tangmunarunkit, et al. The Impact of Policy on Internet Paths. Proc. IEEE Infocom 2001, Anchorage, AK.

A Study of Reconnecting the Partitioned Wireless Sensor Networks

Qing Ye and Liang Cheng

Laboratory of Networking Group (LONGLAB)
http://long.cse.lehigh.edu
Lehigh University, Department of Computer Science and Engineering
19 Memorial Drive, West, Bethlehem, PA 18015, USA
qiy3@lehigh.edu, cheng@cse.lehigh.edu

Abstract. Most wireless sensor networks (WSN) researches assume that the network is connected and there is always a path connected by wireless links between a source and a destination. In this paper, we argue that network partitioning is not an uncommon phenomenon for practical WSN applications, especially when the sensor nodes are deployed in a hash working environment. We first present a simple distributed method to detect the occurrence of network partitions. Some critical survival nodes would be selected to re-connect to the sink after certain network disruption happens. Two reconnection approaches, Transmission Range Adjustment (TRA) and Message Ferry (MF), are then proposed. We study their performances in terms of power consumption by taking the specifications of the commercial sensor nodes into account. Simulation results show that TRA is more appropriate for the current implementation of WSN. However, MF has the potential to be more energy-efficient if there is a powerful wireless transmitter and a larger amount of buffers at each sensor node.

1 Introduction and Related Works

Wireless sensor networks (WSN) consist of a set of small, inexpensive, and ad hoc deployed MEMS nodes that are capable of sensing, computing and transmitting environmental data. These advantages make WSN envisioned as an attractive new approach for a broad range of applications, including national security, public safety, health care, environment control, and manufacturing [1]. In these applications, a sensor network is usually assumed to be connected after the system initialization, i.e., there is always a path connected by wireless links from each node to the sink, which collects the sensed data from the network. However, in many practical cases, the network would possibly be partitioned or disconnected during its operating time, due to node failures caused by environmental disaster, malicious attack, electromagnetic interference, component malfunction or physical damage, especially when sensor nodes are deployed in a harsh working environment.

Several existing researches study the connectivity issue when a WSN is first deployed. Theoretical analyzes about how to preserve network connectivity while achieving the maximum sensing coverage in a fixed area is discussed in [2]. [3]

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 561–568, 2005.

presents the similar discussion but focus on how to solve the problem with minimum number of nodes, so that the rest of deployed sensors can be turned into sleep mode to save more energy. Network connectivity of the grid network, an unreliable deployment of WSN, is analyzed in [4]. It indicts that if more nodes are deployed inside a unit area, the successful connectivity rate of the network would be higher, even if the node failure rate is large.

We are considering the network connectivity from another side of view. We assume a WSN is initially connected at the beginning, but it would be partitioned into several isolated information islands when many nodes fail due to certain disaster. Under this circumstance, how to reconnect the network to recover the disrupted data transmissions becomes an issue. Similar problems are studied in Delay-tolerant Network (DTN) [5]. In this research, we first show that network partition is not an uncommon phenomenon in the uniformly random deployed WSN. Two network reconnection approaches are investigated, and we compare their performances in terms of power consumptions because energy consumption is always a big concern for WSN. The first approach is to reconnect the network by adjusting the transmission ranges of survival sensor nodes after a network partition happens. The second approach is to select certain nodes to be the message ferries that move in the network and get data relayed. Both approaches have their own benefits and working domains. Our simulation results show that the first method is more appropriate for the current commercial implementations of WSN.

The rest of the paper is organized as follows. Section 2 discusses how to detect network partition and how to select the *critical node* to reconnect the network. Section 3 proposes two reconnection approaches. Their performances are compared by simulations in Section 4. Finally, section 5 concludes this paper.

2 Network Connectivity and Partition Detection

2.1 A Simple Study of Network Connectivity

A straight forward way to construct WSN is to uniformly random deploy sensor nodes, i.e., each node would be positioned independently. We study the network connectivity problem as the following: given an open area *A*, let *G(n, r)* denote the network consist of *n* nodes randomly deployed in *A* with the same transmission range of *r*, what is the possibility *P(n, r)* of that *G(n, r)* is successfully connected. Obviously *P*(1, *r)* is always equal to1 when *n*=1 and the following recursive equation holds:

$$P(n,r) = \sum_{i \in I} \left(\frac{C(n-1,r)_i}{S} \times P(n-1,r)_i \right) \tag{1}$$

where *S* is the area of *A*, $C(n\text{-}1,r)_i$ is the covered area of the ith case of the previous deployed (n-1) nodes which are successfully connected. $C(n\text{-}1,r)_i/S$ represents the possibility of that the *n*th node is dropped into the previous connected network constructed by (n-1) sensor nodes. And *I* is the set of all the possible deployments of putting (*n*-1) nodes in *A*. To give a clear view of this problem, a simulation result of randomly placing 1 to 100 sensor nodes into a 50×50m^2 area is shown in Fig. 1(a). We observe that in general, $P(n, r)$ will increases if we put more nodes in A.

However, even with a large number of deployed nodes, if the sensed area becomes unsafe so that sensor nodes begin to fail, the connectivity of the network would not be guaranteed anymore. Network partition can be observed often with high node failure rate. In Fig. 1 (b), we perform three experiments with 50, 100, and 200 nodes deployed in the same area with the same transmission range. We find that more sensor nodes die in the network, the rest WSN would be more possibly disconnected. Thus, network partition is actually not an uncommon phenomenon as long as a WSN is deployed in a harsh working environment.

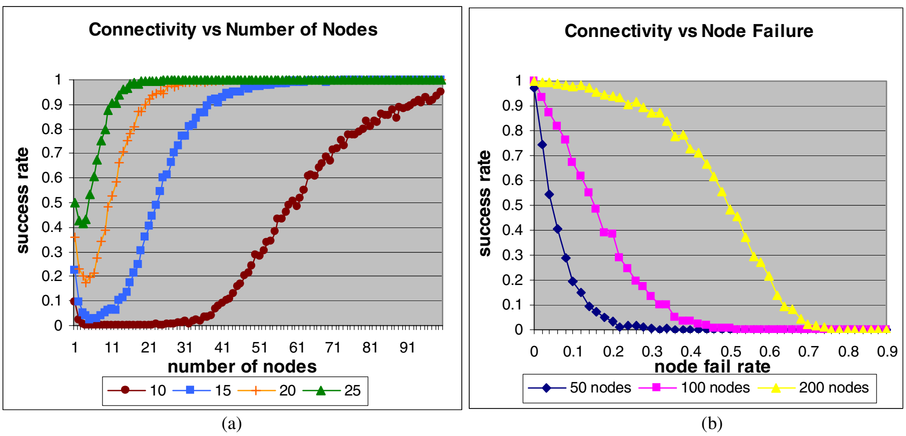


Fig. 1. (a) Network connectivity success rate when randomly putting 1 to 100 sensor nodes in a 50×50m^2 area with transmission range increasing from 10 to 25. (b) Network connectivity success rate of the survival nodes when 10% to 90% nodes fail in the network. The original connected network consists of 50 to 200 nodes

2.2 Network Partition Detection

At the system initial time, the original network hierarchy and topology can be discovered by the following *level discovery approach* performed in a flooding way. Assume after the first deployment, the network is initially connected. The sink is assigned to be level 0 and broadcasts a *level discovery packet*. Those nodes who receive this packet will be assigned a level 1 and take the sink as their parent. Then they continue to broadcast their level assignments to the other nodes. A node may receive several such packets and it only takes the smallest level value it received plus 1 as its own level, and it continues the broadcasting process. Finally when the level discovery period ends, each sensor node discovers its level and creates a parent list and a children list during the message exchanges. A node in a smaller level implies that it is closer to the sink in terms of number of hops. This node is responsible for relaying data from its children to the sink. In fact, each parent of a node represents a possible path connected to the sink. After the network hierarchy is constructed, keep-alive messages would be periodically exchanged between any two levels. Network partition then can be easily detected if one node can't receive any keep-alive message from its parents in a certain period of time. It indicts that this node loses all the chances to get connected to the sink. We name such node as a *potential node*, which is a candidate *critical node* to reconnect the partitioned network.

3 Reconnection Approaches in DTSN

3.1 Critical Node Selection Algorithm

After network partition is detected, some *critical nodes* would be selected to take in charge of reconstructing a path to the data sink. As a result of network partition, the original sensor network would be divided into several isolated clusters, and each *potential node* is located in at least one cluster. Thus, critical node selection method is basically a cluster header selection algorithm. We solve this problem in a very simple way. If there is only one potential node in a cluster, it would be automatically chosen as the *critical node*. If there is more than one potential node in a cluster, then the one with the smallest level value would be selected, because it is closer to the sink. If there are two potential nodes having the same smallest level in one cluster, then simply the one with smaller node ID is selected.

3.2 Network Reconnection by Transmission Range Adjustment (TRA) Approach

One way to reconnect the partitioned network is to increase the transmission range of the critical nodes until they can reach one survival node still connected to the sink. In this case, the network connectivity problem becomes: given an open area A, let $G(n, r(n))$ denote the network with n nodes randomly deployed in A, what is the assignment of the transmission range of each node in $r(n)$, so that $P(n, r(n)) = 1$ or the G is a connected graph. [6] gives a mathematical solution for this problem if the overall topology information is known. We can achieve the same goal by simply asking the critical node increase its transmission power step by step and broadcast *reconnection request packet,* until it receives a *reconnection accept packet* from one survival node which is in a smaller level. If every critical node works in this way, the partitioned network would be recursively reconnected. Note that a loop will never happen because a critical node is only allowed to reconnect with nodes that are closer to the sink than itself. TRA approach is very simple and doesn't require any mobility. However, the network may not be reconnected if a critical node can not receive any *reconnection accept response* even when it works in its maximum transmission range. To tolerate m hops of network disconnection, the initial communication range of each node could be set at most as r_{max}/m, where r_{max} is the maximum transmission range of a sensor node. This approach has to set up WSN in a high dense manner.

3.3 Network Reconnection by Message Ferry (MF)

Another reconnection approach is to ask the critical nodes to move and keep broadcasting *reconnection request packets* during its movement, until they receive a *reconnection accept packet* from a node in a smaller level. The critical nodes then record the position and move back to their clusters. They become message ferries that carry the sensed data of their cluster members in its buffer and move back-and-forth to relay the communications. There could be many movement patterns for these critical nodes to find a survival node linked with the sink. [7] presents a method to decide the route of a message ferry if the position of each cluster is known. In this paper, we always make the critical nodes move towards the sink. This isn't the optimal solution because we may require more nodes to move and make them move more distances. But clearly this movement pattern can guarantee100% network reconnection success rate since even in the worst case the message ferries can reach the sink. Also, it's simple to be implemented and it doesn't require any supreme knowledge of other nodes' behaviors. And compare to TRA, this approach need fewer nodes to cover an area in a sparse manner.

3.4 Performance Comparison

We compare the performance of these two approaches in terms of their power consumption. For wireless communication in sensor network, the transmission power can be model as Eq. 2, where α_1 is the energy/bit consumed by the transmitter electronics, α_2 is the energy dissipation coefficient, d is the transmission distance and r is the number of bits of the transmitted message. The typical values of Eq. 2 is α_1=180nJ/bit and α_2=0.001pJ/bit/m^4 [8].

$$P_{tran} = (\alpha_1 + \alpha_2 d^4) \times r \tag{2}$$

The movement power consumption of a message ferry is the energy spent to overcome the friction between its wheels and the ground. It can be simply modeled as Eq 3, where $m \times g$ is the weight of the node, μ is the friction coefficient, and d is the moving distance. For rubber wheel rolling over the pavement, the typical μ is 0.8:

$$P_{move} = m \times g \times \mu \times d \tag{3}$$

Notice that, if by TRA approach a critical node must increase its transmission range to be R, clearly this node doesn't have to move R distance by MF. It only needs to move into the transmission range of another survival node. For MF approach, its power consumption can be modeled as Eq. 4:

$$P_{MF} = m \times g \times \mu \times d_1 + (\alpha_1 + \alpha_2 d_2^4) \times r \tag{4}$$

If we take the CotsBots mobile sensor node as an example, the weight of a node is 0.6kg (including the weight of the batteries) [9]. With these energy models in mind, we can compare the power consumption of TRA and MF approaches with simply connecting two disconnected sensor nodes, if we assume the maximum transmission range is 1000m. The result is shown in Fig.2. It's easy to see that there exists a turning point at which the MF approach would consume less energy than TRA.

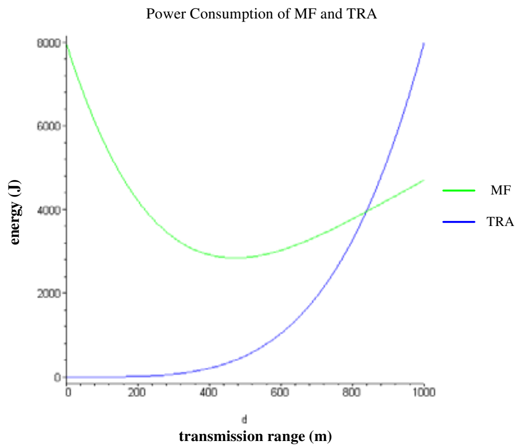


Fig. 2. Power consumption of MF and TRA of connecting two disconnected nodes, 1Mbytes data are transmitted. If the distance between these two nodes is more than 836.16m, then MF approach consumes less power than TRA. It becomes 531.1m for sending 10Mbytes data, and 500m for sending 1Gbytes data

4 Simulation Results

To study MF and TRA in multi-hop sensor network scenarios, we implement both approaches and compare their performances by simulation. We take the specifications of the commercial xbow's MICA2 motes as the parameters of our sensor nodes. The maximum transmission range is then limited as 1000ft (about 300m). And we take the specifications of CotsBots mobile sensors to define the mobility of each node. In our simulation, 300 nodes are randomly dropped in a 500×500m^2 area with the initial transmission range of 50m. Then the original connected network is partitioned due to node failures. Each node has the same possibility to be inactive. We study the power consumption of both methods by adjusting the node failure rate, the transmitted data size and the buffer size in the system. Fig.3 depicts the simulation results. Overall, MF approach would consume more energy than TRA for reconnecting the partitioned network because the maximum transmission range of current practical nodes is only 300m, which is far away from the turning point shown in Fig. 2.

From Fig.3 (a), we can find that when more nodes die in the network, the message ferries selected by MF approach have to move longer distance to find a still-alive node that is closer to the sink. More movement energies are consumed. However, with less active nodes left in the network, the total energy consumption of TAR is decreased (in Fig.3 (b)). Fig.3 (c) and (d) show that both approaches would consume more energy when there are more data need to be transmitted in the network. In this case, the transmission energy consumption becomes the significant part for both MF and TRA methods. When the buffer size in sensor nodes increases, the message ferries can carry more data and don't have to move back-and-forth very often. Thus, the more buffer size the less power consumption of MF approach. However, the additional buffers won't bring any benefit for TRA. Simulation results in Fig.3 (e) and (f) prove this point.

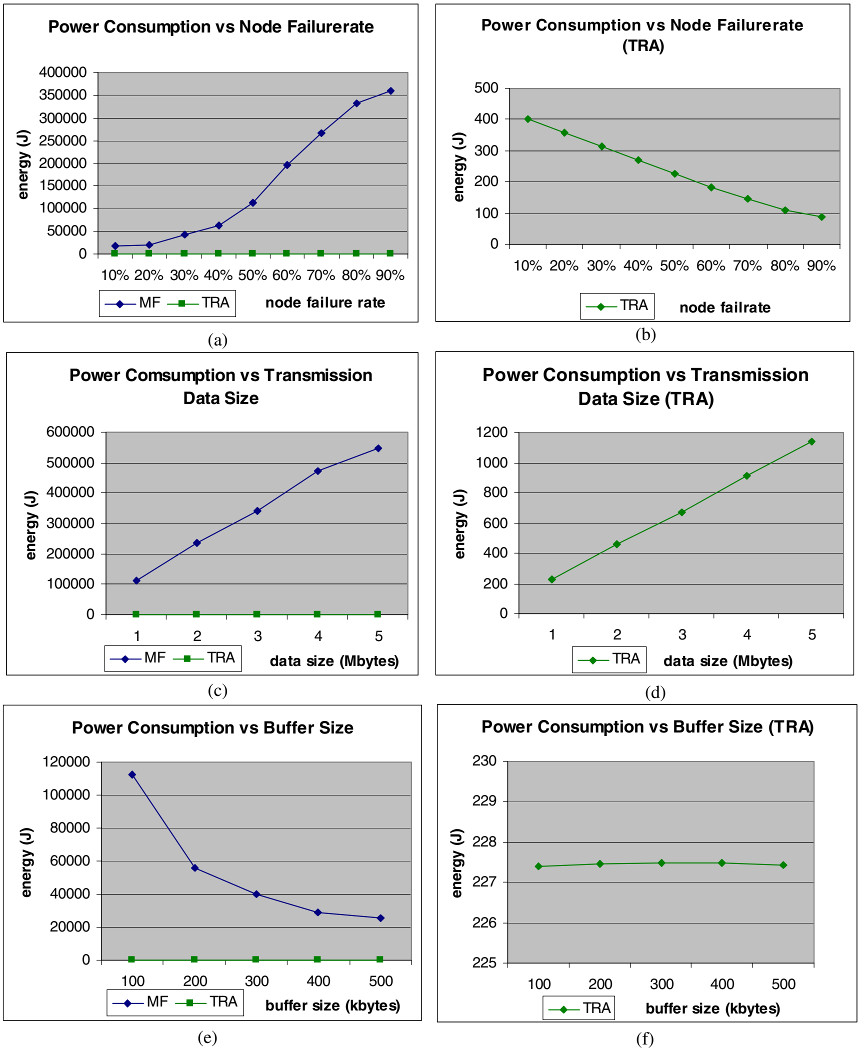


Fig. 3. Compare the power consumptions of MF and TRA in different scenarios: (a) node failure rate increases from 10% to 90% (c) transmitted data size increases from 1Mbytes to 5Mbytes, with node failure rate is fixed at 50% (e) buffer size in a node increases from 100k to 500k, with 50% nodes die in the original network. (b),(d), and (f) show the performance of MF clearly in each case

5 Conclusion

In lots of practical WSN applications, network partition is a common phenomenon especially when sensor nodes are deployed in a hash working environment. In this paper, we present TRA and MF methods for reconnecting the partitioned sensor networks and study their performances in terms of power consumptions. In both approaches, some survival *critical nodes* are selected to re-link to the sink, by either adjusting their transmission ranges or moving around the network to relay the messages. From simulation results we observe that TRA consumes less energy than MF, under the limitations of the current commercial implementation of sensor nodes. However, MF has the potential to be more energy-efficient when mobile nodes have a more powerful wireless transceiver and larger buffers to carry more data.

Acknowledgement

This research has been partly financed by a grant from the Commonwealth of Pennsylvania, Department of Community and Economic Development, through the Pennsylvania Infrastructure Technology Alliance (PITA). The authors also wish to thank Defense Advanced Research Projects Agency (DARPA) for the support

References

1. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, A survey on sensor networks, IEEE Communications Magazine, Vol. 40, No. 8, pp. 102-116, August 2002.
2. Wang, Guoliang Xing, Yuanfang Zhang, Chenyang Lu, Robert Pless, Christopher Gill, Integrated Coverage and Connectivity Configuration in Wireless Sensor Networks, In Proc. of the First ACM Conference on Embedded Networked Sensor Systems (SenSys'03), Los Angeles, CA, November 2003.
3. Di Tian and Nicolas D. Georganas, Connectivity Maintenance and Coverage Preservation in Wireless Sensor Networks. In Proc. of the 2004 IEEE Canadian Conference on Electrical and Computer Engineering, 2004.
4. Sanjay Shakkottai, R. Srikant and Ness Shroff, Unreliable Sensor Grids: Coverage, Connectivity and Diameter. In Proc. of the 2003 IEEE Infocom, Hong Kong, China, 2003.
5. V. Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall, and H. Weiss, Delay tolerant network architecture, http://www.dtnrg.org/specs/draft-irtf-dtnrg-arch-02.txt, Oct 2003.
6. Piyush Gupta, P. R. Kumar, Critical Power for Asymptotic Connectivity In Wireless Networks. pp. 547-566, in Stochastic Analysis, Control, Optimization and Applications: A Volume in Honor of W.H. Fleming. Edited by W.M.McEneany, G. Yin, and Q. Zhang, Birkhauser, Boston, 1998.
7. Tammara Massey, Message Ferry Architecture and Implementation, http://www.cc.gatech.edu/grads/t/tammy/MF_Master_Proj.pdf
8. L. C. Zhong, J. M. Rabaey and A. Wolisz, "An integrated data-link energy model for wireless sensor networks", ICC 2004, Paris, France, June 20-24, 2004.
9. CotsBots. http://www-bsac.eecs.berkeley.edu/projects/cotsbots/

Application-Driven Node Management in Multihop Wireless Sensor Networks

Flávia Delicato[1,3], Fabio Protti[2], José Ferreira de Rezende[3], Luiz Rust[1], and Luci Pirmez[1]

[1] Núcleo de Computação Eletrônica,
[2] Computer Science Department,
[3] Grupo de Teleinformática e Automação - Federal University of Rio de Janeiro,
P.O Box 2324, Rio de Janeiro, RJ, 20001-970, Brazil
{fdelicato, fabiop}@nce.ufrj.br
rezende@gta.ufrj.br
{luci, rust}@nce.ufrj.br

Abstract. A strategy for energy saving in wireless sensor networks is to manage the duty cycle of sensors, by dynamically selecting a different set of nodes to be active in every moment. We propose a strategy for node selection in multihop sensor networks that prioritizes nodes with larger residual energy and relevance for the application. The proposed scheme is based on an implementation of the knapsack algorithm and it seeks to maximize the network lifetime, while assuring the application QoS. An environmental monitoring application was simulated and huge energy savings were achieved with the proposed scheduling algorithm.

1 Introduction

WSN applications often request the deployment of sensors in hard access areas, turning battery recharge or sensor replacement so difficult that it is important to keep sensor nodes alive as long as possible. Therefore, the network operational lifetime is severely constrained by the battery capacity of its nodes. Energy saving becomes a paramount concern in WSNs, particularly for long running applications [5].

WSNs often have a large density of nodes, generating redundant data. Recent works [8,9,10] argue that, instead of providing such unnecessary redundancy to the application, the large density of nodes can be exploited to achieve significant energy savings by dynamically activating a reduced set of sensors (i.e. some nodes are assigned “to sleep”).

This work analyses the potentiality of adopting an enhanced sensor management in multihop WSNs, based on the strategy of turning off redundant sensors to extend the network lifetime while satisfying application requirements. The fundamental problem concerns the election of nodes that should remain active. Basically, the election process is formulated as an optimization problem, which is solved by the knapsack algorithm [2]. The major goal is to maximize relevance (for the application) and residual energy of active nodes, constrained by connectivity, coverage and energy issues.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 569–576, 2005.

Several researchers have been investigating the problem of WSN management in the last years, most of them seeking to achieve high levels of energy efficiency and considering the guarantee of coverage and connectivity as the unique QoS requirement for WSNs. In [1] techniques of linear programming are used to select the minimum set of active nodes able to maintain the sensing coverage of the network. Application specific requirements were not considered in these works. In [9] and [8] the problem of maximizing the lifetime of a WSN while guaranteeing a minimum level of quality at the application level is addressed. In those works, node selection and data routing are jointly addressed, and solved as a problem of generalized maximum flow. They present both an optimal and a heuristic solution with a totally centralized approach.

In contrast, our work addresses the active node selection as a problem independent from the network routing protocol. The proposed scheme for node selection considers as QoS requirements, besides coverage and connectivity requirements, network-related and application-related parameters, such as network lifetime and data accuracy. Furthermore, differently of approaches based on computational intensive techniques of linear programming, which are restricted to run off-line, the proposed approach is light enough to be executed inside the sensor network.

The rest of this paper is organized as follows. In Section 2 we present the problem description and formulation. Section 3 describes the performed simulations and results. Finally, Section 4 presents our conclusions.

2 Node Election in MultiHop Wireless Sensor Networks

Given an application submitting a sensing task to the WSN, the node election algorithm decides which sensors should be active for the task execution. In the proposed algorithm, time is divided in rounds. During each round r the subset of selected nodes and the role of each node (sensor/router) do not change. A task launching at the round initiation can last a time interval equal to an integer number multiple of p, where p is the round extent.

The algorithm of node election is firstly executed when interests from a new application are submitted to the network. Application interests consist of the task descriptor and QoS requirements. The first round of a task starts just after the election algorithm is concluded. The algorithm is executed again in the following cases: (i) on-demand by the application to change some QoS parameter; (ii) proactively by the network, for purposes of energy savings; or (iii) reactively by the network whenever some QoS violation is detected.

2.1 Network and Application Models

A WSN is usually composed of hundreds of sensor nodes and one or more sink nodes. Sink nodes are entry points of application requests and gathering points of sensor-collected data. The data communication in WSNs is accomplished through multiple hops from data sources to sink nodes.

The energy model assumes that sensors are capable to operate in a sleep/inactive mode or according to K predefined active modes. Two main roles are assumed by active nodes: (i) source, for nodes placed inside the target area; (ii) router, for nodes outside the target area, responsible for forwarding their neighbors data. Furthermore, a sensor can play both roles, simultaneously. In each mode, a sensor spends a different amount of energy [3].

An application of environmental monitoring (continuous measurements about a given physical phenomenon) was chosen as the target of our work. The application defines a data-sending rate, a geographical area of interest, monitoring time and, optionally, one or more data aggregation functions. Furthermore, the application defines a minimum value for the accuracy and for the spatial precision of the sensor-collected data.

2.2 Problem Formulation

The proposed scheme for node selection aims to maximize the lifetime of a network containing N multi-mode sensors while guaranteeing a required level of application quality. The adopted algorithm seeks out the best set of sensors to be activated for accomplishing a specific sensing task. Two strategies can be used to extend WSN lifetime: (i) to minimize the network energy consumption by choosing the smallest possible number of nodes capable of providing the requested level of QoS; and (ii) to maximize the residual energy of the selected nodes, that is, to consume energy in a uniform way among sensors along time, thus avoiding the premature collapse of excessively used nodes. Both strategies are used in the proposed algorithm. Further, the algorithm takes into account the potential relevance of data reported by each sensor, from the application point of view.

The proposed scheme for node selection was modeled as a knapsack problem [2], with some additional constraints. With the knapsack algorithm applied to the problem of active node selection, the sum of the utilities of nodes placed in the knapsack is optimized under the constraint of the energy budget considered. The algorithm seeks to maximize the relevance R_i and the final residual energy U_i of the selected nodes. The objective function of the problem is given below:

$$\begin{aligned} &\text{Max } \Sigma\, x_i\,(\propto R_i + \beta\,(U_i - w_i)) \qquad (1)\\ &\text{st. } \Sigma\, x_i\, w_i \le M, \text{ where } x_i \in \{0,1\} \end{aligned}$$

A value 1 for x_i indicates that sensor i is selected to participate of task T. The term $U_i - w_i$ denotes the final energy of sensor i, if it was chosen to participate of the task (initial residual energy U_i minus the energy spent for the sensor in the task, w_i). The coefficients $\propto$ and β are used to balance the priorities given for each term of the equation, and they depend on the application QoS requirements. In the general case, $\propto = \beta = 1$.

The relevance of a node i depends on its physical and topology characteristics, given by its nominal precision (NP_i); the environmental noise of its measurements (F_i); its set of sensing neighboring nodes (N_i) and its proximity of the target area defined by the application (A_i). Each parameter contributes with a different balancing factor for the computation of R_i. The value of NP_i is a physical feature of each sensor. We assumed that NP_i have the smallest balancing factor among all terms for computing R_i. The parameter F_i is mainly influenced by the physical characteristics of the place where i was deployed. The parameter F_i is in fact a normalized value that depends upon the actual level S_i of environmental noise, where S_i ranges from 0 to 100. We applied the formula $F_i = 1 - S_i /100$ (2).

The largest balancing factors were assigned to the parameters A_i and N_i. The values of those two parameters are highly correlated. The value of N_i is inversely proportional to the amount of neighbors of the sensor. The importance of the value measured by a node in a location X,Y is proportional to the contribution of that sensor for sensing such location.

For calculating the value of A_i, sensors with distances d_i from the target area larger than the radio range Rr are automatically excluded from selection. Since it is desired to assign a smaller value of relevance for sensors located at larger distances from the target area, we applied the formula $A_i = 1 - d_i/Rr$ (3).

From the observed correlation between A_i and N_i, and considering the different balancing factors of each parameter in the calculation of R_i, the following equation is used:

$$R_i = \delta\ NP_i + \phi\ F_i + \gamma\left(\frac{1}{\mathrm{Ai\ \ Ni}}\right) \qquad (2)$$

where ϕ, δ and γ are coefficients that represent the balancing factors of each parameter, and $\delta < \phi < \gamma$.

2.2.1 Including QoS Profiles

Applications can choose to prioritize the lifetime in favor of the accuracy, or to prioritize the accuracy in favor of the monitoring period, or they can choose to balance both parameters. In the present work, the application QoS requirements, along with the parameter that it chooses to prioritize, compose a QoS profile. There are 3 possible QoS profiles: (i) precision-based, which prioritizes the data accuracy or precision; (ii) lifetime-based, which prioritizes the network lifetime; and (iii) ratio-based, that seeks the best tradeoff between energy consumption and data accuracy.

Considering the QoS profiles above, the original objective function (4) is modified to include different weights according to the priority given by the application to the different QoS parameters. For precision-based profiles, larger values are assigned to the coefficient α; for lifetime-based profiles, larger values are assigned to the coefficient β; and finally, for ratio-based profiles, equal values are assigned to both the coefficients.

2.3 Constraints

The choice of active nodes in a WSN is subject to a set of constraints, which should be taken into account by any scheme for node selection.

2.3.1 Energy Constraints

The first constraint to be considered (R1) is the finite amount of energy of the network. At each round j, the energy spent by the selected set of sensors cannot be larger than the budget of energy of the network for that round. The constraint R1 is already taken into account by the knapsack algorithm, since the value (capacity) of the knapsack is the total budget of the network in each given round.

A second energy-related constraint (R2) considers that a sensor node is only eligible to remain active in a round j if it has energy enough to remain alive up to the end of the round. To satisfy that constraint, we defined a minimum energy threshold, L, which a node should have to be eligible for selection. For establishing such threshold we assumed that, if the node is inside the target area, it should have at least energy enough for sensing at the defined rate and to transmit its data. Otherwise, it should have at least energy to forward its neighbors' data. The constraint R2 can be defined as follows: $x_i \leq U_i/L$ (R2)

Since x_i is a binary variable, if the residual energy U_i of sensor i is smaller than the threshold L, x_i is set to 0 (the sensor cannot be selected). Otherwise, if $U_i \geq L$, then the variable x_i may or may not be set to 1 (that is, the sensor i is eligible).

The constraint R2 can be included in the knapsack algorithm, by including an additional if, or it can be solved through a previously executed procedure.

2.3.2 Coverage and Connectivity Constraints

Since the primary goal of a WSN is to monitor the environment, it has to maintain a full sensing coverage, even when it operates in a power save mode. Besides, a successful WSN operation must also provide satisfactory connectivity so that all active nodes can communicate for data fusion and report to sink nodes.

In this work, a point p is assumed to be covered by a node i if the Euclidian distance between them is smaller than the sensing range of i, denoted by Sr. Another assumption is that the covering area CA of sensor i is the circular area with center in X,Y, where X,Y are the geographical coordinates of i, and whose ray is Sr. A convex area A is defined as having a degree of coverage K (that is, A is K-covered) if every point p inside A is covered by at least K nodes [10]. In addition, we assumed that any two nodes i and j can directly communicate if the Euclidian distance between them is smaller than the radio range of the nodes, Rr, i.e., $d(i,j) < Rr$.

The coverage and connectivity constraint R3 can be formulated as follows. Given a convex area A and a coverage degree K specified by the application, the number of inactive nodes should be maximized under the constraint that (i) active nodes guarantee that A is at least K-covered and (ii) all active nodes are connected. That is, for every point p of A:

$$\sum_{i \in A} x_i \geq K \text{ (coverage degree requested by the application)} \qquad \text{(R3)}$$

To satisfy such constraint, a procedure based on the disk-covering algorithm [6] was employed before executing the knapsack algorithm. That procedure consists of two stages, the first one aiming to guarantee the coverage of the target area and the second one to guarantee the network connectivity. In the first stage, the target area (a rectangular area defined by the application) is totally covered by disks whose diameter is defined as the spatial precision requested by the application. Afterwards, the procedure heuristically selects K nodes that must remain active inside each disk. That selection can be totally random or it can take into account the residual energy of the nodes. In the second stage, the sensor field is totally covered by disks whose ray is equal to the radio range *Rc*. To assure the network connectivity, the procedure should guarantee that in each disk there is at least one active node.

3 Simulations

We ran simulations in the JIST simulator [7] to demonstrate the benefits of using our scheme for node selection in WSNs. A greedy heuristic for solving the knapsack algorithm was implemented [2]. The algorithm runs in an unconstrained sink node.

An application of environmental monitoring was simulated. The requested sensing task was to monitor the temperature of a target area during a period of time. The application was interested on raw data values (with no aggregation), with the following requirements: (i) a spatial resolution of $40m^2$ with a 1-coverage degree (at least 1 sensor at each $40m^2$); (ii) an acquisition rate of 10 seconds, and (iii) a data accuracy above a predefined threshold. Data accuracy is given by the Mean Square Error (MSE) value. The MSE is calculated as the difference between a set of values assumed as “real” values and the set of values generated by sensors, considering their nominal precisions and the environmental noise. The WSN lifetime has to be long enough to guarantee that data will be collected during all period of time requested by the application and respecting QoS requirements.

A sensor field was created with 300 nodes randomly distributed in a square area with 200m x 200m. Each node had a radio range of 40m and a sensing range of 20m. The energy dissipation model is as described in [3]. Sensors that generate data (sources) were randomly selected from nodes in a 100m by 100m square (target area) within the sensor field. The sink node was located in the right upper bound of the field. Since we were not interested in simulating any specific routing or MAC protocol, we assumed hypothetical protocols, delivering data generated from sources to the sink node through the shortest path (in terms of geographical distance), without data loss. Each simulation runs for 1000 seconds, divided in 10 or more rounds, at the end of each the network residual energy and the MSE are computed. “Real” values of temperature data were randomly generated at every round ranging from 20 to 40 degrees Celsius. The size of data packets in all transmissions is fixed and equal to 100 bytes. All results correspond to the average of 10 simulation runs.

In the first simulations, we compare the results of scheduling different percentages of active nodes, in terms of final residual energy of the network and data accuracy. Our goal is to show that activating only a subset of nodes can satisfy the QoS requested by the application, leaving WSN resources for new tasks and applications. The network energy budget (knapsack capacity) is specified as a percentage of active nodes, which varies from 30% to 100%. A budget given in percentage means that the knapsack capacity is set to the sum of the weights of the respective percentage of nodes. In the greedy approach, we assume that the weights of all nodes were the same and equal to their initial energy. All sensors have an initial energy randomly chosen between 15 and 20J. Before running the selection procedure, nodes were sorted according to their relevance and residual energy, so that the procedure prioritizes the selection of nodes with larger values for these parameters. After running the procedure, routes from sources to the sink were established and kept unchanged until the end of the rounds. The monitoring time requested by the application corresponds to 9 rounds and the maximum tolerated MSE was 0.3.

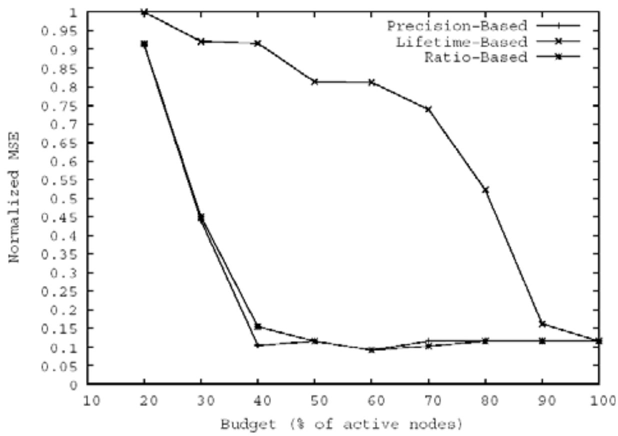


Fig. 1. Normalized MSE at the 10th round, for the different budgets, considering the three QoS profiles

Results shown that a gain of 1000% in the final energy at round 9 is obtained when only 30% of nodes are activated, in contrast with activating 100% of nodes. We observed that from the 8th round the MSE starts increasing for all budgets. This is due to a large number of sensors being short of energy. Lifetime expiration of source nodes or nodes located in the path from sources to the sink prevents data delivery. Although the MSE increases, up to the 9th round it is still below the point tolerated by the application, for all budgets. From the next rounds, MSE increases to a value above the desired threshold, meaning that the application QoS is not being satisfied anymore. Since the monitoring time was requested as 9 rounds, results prove that with only 30% of nodes the application QoS was met, with a huge energy saving. Next, we varied the number of sensors while keeping the size of the sensor field, to analyze the effect of node density. Similar results were achieved for 400 nodes. For 200 nodes a smaller although significant energy saving of 300% was obtained. These results indicate that schemes for node scheduling are more suitable for high density WSNs.

All the previous simulations assumed a ratio-based QoS profile. Next, we evaluate the effect of using the different profiles considered in this work. For the precision-based profile the value of the coefficient $\propto$ was set to 50, while β was set to 1. For the lifetime-based profile the value of the coefficient $\propto$ was set to 1, while β was set to 50. Results show that the final energy does not significantly change among the different profiles. This result is due to the fact that the selection algorithm runs before the first round, when the residual energy of all nodes is very similar. A different result would probably be achieved if nodes were assigned energy values with larger ranges.

On the other hand, the values of relevance vary a lot among different nodes. Results shown in Fig.1 corroborate this fact. When the application decides to prioritize the relevance (precision-based profile) the final value of error was up to 90% smaller then when the network lifetime is prioritized.

4 Conclusions

We presented a scheme for node selection in multihop WSNs whose primary goal is maximizing residual energy and application relevance of active nodes. We formalized the problem of node selection as an optimization problem, and we adopted the knapsack algorithm for solving it. An application of monitoring environment was chosen to derive some specific requirements. We adopted a non-optimal, greedy approach for solving the knapsack problem, whose complexity is low enough to allow an online, in-network execution of the algorithm. Simulation results are very encouragers, and huge energy savings can be achieved while preserving application QoS requirements.

References

1. Chakrabarty, K. et al.: Grid coverage for surveillance and target location in distributed sensor networks. IEEE Transactions on Computers, 51(12), pp. 1448-1453 (2002)
2. Cormen, T. H. et al.: Introduction to Algorithms. MIT Press (2001)
3. Estrin, D., Sayeed, A., Srivastava, M.: Wireless Sensor Networks. Mobicom2002 Tutorial. Available in: http://www.sigmobile.org/mobicom/2002/ program/ tutorial.html
4. Frolik, J.: QoS Control for Random Access Wireless Sensor Networks. In Proc. of IEEE WCNC2004, Atlanta (2004)
5. Mainwaring, A. et al.: Wireless sensor network for habitat monitoring. In Proc. of WSNA2002, Atlanta (2002)
6. Pach, J., Agarwal, P.K.: Combinatorial Geometry. Wiley Pubs. New York (1995)
7. JIST: Java in Simulation Time. Available in http://jist.ece.cornell.edu/
8. Perillo, M., Heinzelman, W.: Optimal sensor management under energy and reliability constraints. In Proc. of the IEEE WCNC2003, New Orleans (2003)
9. Perillo, M., Heinzelman, W.: Sensor Management Policies to Provide Application QoS. Elsevier AdHoc Networks Journal, Special Issue on Sensor Network Applications and Protocols, 1 (2-3), pp 235-246 (2003)
10. Wang, X. et al.: Integrated Coverage and Connectivity Configuration in Wireless Sensor Networks. In Proc. of ACM SenSys03, Los Angeles (2003)

Power Management Protocol for Regular Wireless Sensor Networks*

Chih-Pin Liao[1], Jang-Ping Sheu[1], and Chih-Shun Hsu[2]

[1] Department of Computer Science and Information Engineering,
National Central University, Chung-Li, 320, Taiwan
[2] Department of Information Management,
Nanya Institute of Technology, Chung-Li, 320, Taiwan

Abstract. Most of the existing power saving protocols are designed for irregular networks. These protocols can also be applied to regular networks, but these protocols do not consider the characteristics of regular networks and thus are more complicated and less efficient than the protocols designed for regular networks. Therefore, we propose a novel power management protocol for regular WSNs. Gathering information to the base station is an important operation for WSN. Hence, even some nodes switch to PS mode, the network still needs to be connected so that the sensed information can be sent to the base station through the active nodes. The goal of our protocols is to choose several different connected dominating sets, so that these connected dominating sets can switch to active mode in turn to serve other nodes in PS mode. Simulation results show that our power management protocol can conserve lots of power and greatly extend the lifetime of the WSN with a reasonable extra transmission delay.

1 Introduction

The wireless sensor network (WSN) is a network which consists of thousands of wireless sensor nodes. The wireless sensor node is a low-cost, small size, and power-limited electronic device, which consists of three components: the sensor, the general purpose signal processing engine, and the radio circuit. Among the three components of the wireless sensor node, the amount of power consumed by the radio frequency circuit is the most. Therefore, we should try to reduce the amount of power consumed by the radio frequency circuit so that the lifetime of the network can be extended.

One of the best solutions for saving power is to let the wireless sensor node switch to PS mode by turning off its radio circuit when it has no information to transmit or receive. Many power management protocols for WSNs have been proposed [1], [2], [3], [4], [5], [6]. These power management protocols are designed for irregular networks and can also be applied to regular networks. However,

* This work was supported by the National Science Council of the Republic of China under grant NSC 92-2213-E-008-006.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 577–584, 2005.

these protocols do not consider the characteristics of regular networks and thus are more complicated and less efficient than the protocols designed for regular networks.

As we know that the power management protocols for regular WSNs have not been proposed before. Therefore, we propose a novel power management protocol for regular WSNs. The goal of our protocol is to let as many sensor nodes as possible switch to PS mode while still maintaining the connectivity of the network so that if any emergency occurs, the sensor node, which sense the event, may transmit this information to the base stations through the active sensor nodes without need to wake up any node in PS mode. Besides, each sensor node should switch to PS mode in turn, so that the power consumption of each node can be balanced. Although based on the concept of connected dominating set to design power saving protocol has been addressed in [7], [8], yet, how to choose several connected dominating sets and balance the power consumption of each dominating set is still an open question.

Our protocol works as follows: first, choose several different connected dominating sets according to the network topology and assign an *id* to each of the connected dominating set, and then the nodes in each connected dominating set will switch to active mode to serve the other nodes in PS mode according to which dominating set they belong to in a round robin manner. Each node can decide which connected dominating set it belongs to according to its own *id*. Our protocol can still work even there are faulty nodes. Performance analysis shows that the ratio of active nodes of our protocol is near optimal and much lower than those of GAF [8] and SPAN [7], those are designed for high density irregular networks. Simulation results show that our power management protocol can conserve lots of power and greatly extend the lifetime of the network with a reasonable extra transmission delay.

The rest of this paper is organized as follow. Section 2 describes the system models. Section 3 presents the novel power management protocol. Performance analysis is shown in Sect. 4. Simulation results are shown in Sect. 5. Conclusions are made in Sect. 6.

2 System Models

The $CSMA/CA$ like protocol is adopted as our MAC protocol and the First Order Radio Model [9] is adopted to evaluate the power consumption of each sensor node. We assume that the base station can directly transmit messages to all the nodes in the WSN and there are 4 base stations located in the corners of the WSN. when a sensor node switch to PS mode, it only turn off the power of its radio circuit. Therefore, it can still monitor the change of the environment. When a sensor node detects any emergency, it will turn on its radio circuit and transmit the sensed information through the active nodes to the base station. To guarantee each node in the same connected dominating set sleeps and wakes up at about the same time, we have to synchronize these nodes. We can synchronize the WSN according to the protocols proposed in [10], [11]. The size of the mesh is assumed to be $m \times n$, where m and n are positive integers. The node (x, y) is located in the xth column and yth row of the mesh, where $1 \leq x \leq m$ and $1 \leq y \leq n$. There are c different connected dominating sets and c time slots in each frame, the ith connected dominating set is denoted as CDS_i.

3 Power Management Protocol

The wireless sensor nodes have no plug-in power. Therefore, how to conserve the battery power of wireless sensor nodes so that the network lifetime can be extended is a critical issue for the WSNs. One of the best ways to conserve power is to let the wireless sensor nodes switch to PS mode. When a wireless sensor node switches to PS mode, it turns off its radio circuit and shall keep its sensor active. There are two goals that must be achieved when designing our power management protocols: first, every wireless sensor node should have almost the equal chance to switch to PS mode so that the power consumption can be balanced. Second, the wireless sensor nodes in active mode should be connected and dominate all the nodes in the network so that any sensed information can be transmitted to the base stations through the active nodes.

Our protocol works as follows: first, choose several different connected dominating sets according to the network topology and assign an *id* to each of the connected dominating set, and then the nodes in each connected dominating set will keep active to serve the other nodes based on the dominating set they belong to in a round robin manner. In the following, we first show our power management protocol and then we will discuss the fault tolerant issue of our power management protocol.

3.1 Power Mode Switch

The connected dominating sets are chosen according to the following guidelines: first, choose nodes from certain columns, or rows to form several different *basic dominating sets(BDS)*. These basic dominating sets are the bases of the connected dominating sets. We can choose some nodes to join each BDS, so that each BDS can be connected and form one or several connected dominating sets.

When the connected dominating sets are chosen and the nodes in the current active connected dominating set are synchronized, each node switches its power mode according to the following rules:

R1 Any node that belongs to CDS_i shall wake up and serve the other nodes in the ith time slot of each frame.

R2 The other nodes, which do not belong to CDS_i and have no message to transmit, will switch to PS mode.

R3 The nodes, which belong to $CDS_{i \bmod c+1}$, are the successor of the nodes in CDS_i.

R4 If the node in CDS_i is going to sleep and still have messages to transmit, it will pass these messages to any of its neighbors in $CDS_{i \bmod c+1}$ and then switch to PS mode.

3.2 Choose CDS for Regular Mesh Topologies

In this subsection, we will show how to choose the connected dominating sets in 2D mesh with 4 neighbors. With similar manner, we can also choose the connected dominating sets in other regular mesh topologies. In 2D mesh with 4 neighbors, each node can dominate 4 of its neighbors. Therefore, in the ideal case, only $\frac{1}{5}$ of the nodes need to be chosen as the members of the dominating set. However, only $\frac{1}{5}$ of the nodes in the dominating set are not enough to form a connected dominating set. To form a connected dominating set, two neighbors of the members in the dominating set also need to join the dominating set.

For the simplicity of choosing the connected dominating set, once a node becomes a member of the connected dominating set, the nodes in the same column (or the same row) will also become the members of the connected dominating set. As Fig. 1(a) shows, we can choose the nodes in columns $1, 4, 7, \ldots, 3k+1$, where $3k+1 \leq m$, to form the first basic dominating set(denoted as BDS_1). Similarly, as Fig. 1(b) shows, we can choose the nodes in columns $2, 5, 8, \ldots, 3k+2$, where $3k+2 \leq m$, to form the second basic dominating set(denoted as BDS_2). However, we will not choose the nodes in columns $3, 6, 9, \ldots, 3k+3$, where $3k+3 \leq m$, to form the third basic dominating set(denoted as BDS_3). If they become the third BDS, the nodes in the first column are not dominated by any nodes. Thus, nodes in the first column also need to join BDS_3. Then, the nodes in the first column need to keep active in BDS_1 and BDS_3 for every three time slots, which is not a good approach for power saving protocol. To balance the power consumption, the nodes in the $(3k+3)$th column will not become the third BDS. Instead, we will choose the nodes in the $(3l+1)$th and $(3l+2)$th rows, where $l \geq 0$ and $3l+2 \leq n$ to form the third (BDS_3) and fourth (BDS_4) basic dominating sets, respectively, as shown in Fig. 1 (c) and Fig. 1 (d). With the similar reason as the $(3k+3)$th column, we will not choose the nodes in the $(3l+3)$th row to form a basic dominating set, for $3l+3 \leq n$. The nodes in BDS_1, BDS_2, BDS_3 and BDS_4 will switch to active mode to serve other hosts in the $(4k+1)$th, $(4k+2)$th, $(4k+3)$th and $(4k+4)$th time slots of each frame, respectively, where k is an nonnegative integer.

Since BSD_1 and BSD_2 are not connected dominating sets, we need to choose a row to join BDS_1 and BDS_2 whenever any of them becomes active, so that the union of the basic dominating sets and the nodes in the chosen row can form one or several connected dominating sets. Similarly, we need to choose a column to join BDS_3 and BDS_4 whenever any of them becomes active. Since most of the nodes in the $(3k+3)$th column and $(3l+3)$th row do not belong to any of the four basic dominating sets, we will choose the nodes in the $(3l+3)$th row to join BDS_1 and BDS_2 and the nodes in $(3k+3)$th column to join BDS_3 and BDS_4. Therefore, the nodes in the $(6l+3)$th row will join BDS_1 in the $(4l+1)$th time slot and the nodes in the $(6l+6)$th row will join BDS_2 in the $(4l+2)$-th time slot. Similarly, the nodes in the $(6k+3)$th column will join BDS_3 in the $(4k+3)$th time slot and the nodes in the $(6k+6)$th column will join BDS_4 in the $(4k+4)$-th time slot.

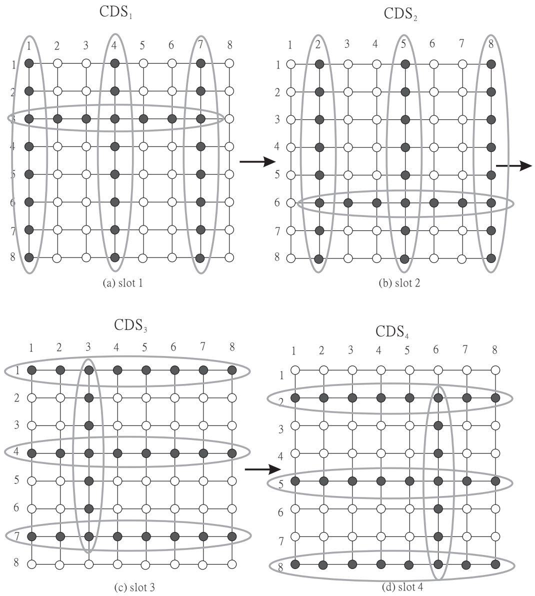


Fig. 1. The power mode switch with four connected dominating sets in an 8×8 2D mesh with 4 neighbors

Fig. 1 shows the power mode switch with four connected dominating sets in an 8×8 2D mesh with 4 neighbors. We choose the nodes in columns 1, 4, and 7 to form BDS_1, the nodes in columns 2, 5, and 8 to form BDS_2, the nodes in rows 1, 4, and 7 to form BDS_3, and the nodes in rows 2, 5, and 8 to from BDS_4. The union of BDS_1 and the nodes in row 3 form CDS_1, the union of BDS_2 and the nodes in row 6 form CDS_2, the union of BDS_3 and the nodes in column 3 form CDS_3, and the union of BDS_4 and the nodes in column 6 form CDS_4. The frame length is $4 \times T_a$ and the nodes in CDS_1, CDS_2, CDS_3, and CDS_4 will switch to active mode to serve other hosts in the first, second, third, and fourth time slots of each frame, respectively.

Note that, the protocol proposed above can work properly only when $(m \bmod 3) = 2$ and $(n \bmod 3) = 2$. When $(m \bmod 3) = 1$ and the nodes in BDS_2 wake up to serve other hosts, the nodes in column m will not be dominated by any node. Therefore, the nodes in column m also need to join BDS_2 when $(m \bmod 3) = 1$. When $(m \bmod 3) = 0$ and the nodes in BDS_1 wake up to serve other hosts, the nodes in column m will not be dominated by any node. Therefore, the nodes in column m also need to join BDS_1 when $(m \bmod 3) = 0$. Similarly, when $(n \bmod 3) = 1$, the nodes in row n also need to join BDS_4. When $(n \bmod 3) = 0$, the nodes in row n also need to join BDS_3. Overall, in case of $(m \bmod 3) = 2$ and $(n \bmod 3) = 2$, no extra nodes need to be active and thus conserve most power.

3.3 Fault Tolerance

Fault tolerance is an important issue for our power management protocol because the network may not be so regular and some nodes may become faulty and thus the connected dominating set may be broken. When a sensor node, say node a, has detected that its next hop node is faulty or out of its location. It will try to establish a route to the node which belongs to current active CDS and is nearest to node a. Among all such nodes, the node which is nearest to the base station and can connect to the base station will be chosen. When the route has been established, the nodes in the route will join current active CDS. For example, in Fig. 1 (a), node $(4,5)$ realizes that node $(4,4)$ is faulty, node $(4,5)$ will try to establish a new route to node $(3,3)$. Since nodes $(3,4)$ and $(3,5)$ are in the new established route, they will join CDS_1. If the faulty node is an intersection, node a will try to establish routes to connect the neighbors, which belong to current active CDS, of the faulty node. The nodes belong to the routes will join the current active CDS. For example, in Fig. 1 (a), node $(2,3)$ realizes that node $(1,3)$ is faulty, node $(2,3)$ will try to establish new routes to nodes $(1,2)$ and $(1,4)$. Since nodes $(2,2)$ and $(2,4)$ are in the new established routes, they will join CDS_1.

4 Performance Analysis

We evaluate the ratio of active nodes for our power management protocol in this section. The ratio of active nodes is defined as the average number of active nodes in a time slot over the total number of nodes in the WSN. With the ratio of active nodes, we can estimate the total amount of power that can be conserved in the WSN. The lower the ratio is, the better the performance is.

In 2D mesh with 4 neighbors, each node can dominate 4 neighbors. Therefore, without considering connection, only $\frac{1}{5}$ of the nodes need to be chosen as the members of the dominating set. However, only $\frac{1}{5}$ of the nodes in the dominating set are not enough to form a connected dominating set. To form a connected dominating set, at least two neighbors of the members in the dominating set also need to join the dominating set. Therefore, each dominating node can only dominate two non-dominating nodes. In the ideal case, at least $\frac{1}{3}$ of the nodes need to join the connected dominating set. In our protocol, since $\frac{1}{3}$ of the columns (or rows) will be chosen to join the connected dominating set, the ratio of active nodes of our protocol is quite close to the ideal case, except that we need to pick an extra row (or column) to join the connected dominating set and connect the separated columns (or rows).

According to the above analysis, the ratio of active nodes of our protocol is quite close to that of the ideal case. Our protocol also performs better than GAF [8] and SPAN [7], those are designed for irregular networks. In GAF, the host density should be no less than $\frac{5}{R^2}$, where R is the communication range of the node, otherwise, some grids would be empty. Since the host densities of 2D mesh with 4 neighbors is $\frac{1}{R^2}$, which is less than $\frac{5}{R^2}$, all the hosts need to be active all the time and thus can not conserve energy. In SPAN, a node should become a coordinator if it discovers, using local information, that two of its neighbors cannot reach each other either directly or via one or two coordinators. In 2D mesh with 4 neighbors, each node can discover that two of its neighbors cannot reach each other either directly or via one or two coordinators and thus all the nodes need to become coordinator. Since all the nodes are coordinators, all the nodes need to active all the time.

5 Simulation Results

To evaluate the performance of the proposed power management protocols, we have developed a simulator using C. The distance between each sensor node is 1 meter, the transmission rate is 8K bits/sec, the battery power of each sensor node is 10 Joules, the packet size is 1K bytes, the length of a time slot is 10 seconds. We will randomly choose a sensor node to transmit a packet to the base station every 10 seconds. To show the efficiency of our protocols, we will compare the performance of our protocols with that of the always active scheme. In the always active scheme, every node in the WSN shall keep active all the time until it run out of its battery.

Two performance metrics are used in the simulations:

- the network life time: the time from the WSN starts operation to the time the first sensor node runs out of its battery.
- transmission delay: the time from the sensor start transmits the sensed information to the time a base station receiving the information. Here, we use hop counts to represent the transmission delay.

According to the analysis in Sect. 4, our protocol performs much better than GAF and SPAN. Therefore, we will not simulates the two protocols.

The network life time of the always active scheme and our protocol are shown in Table 1. As we can see that our protocol can greatly extend the network life time.

When the message is transmitted along the connected dominating set to the base station, it may not go through the shortest path. Therefore, our protocol may cause some extra transmission delays. As Table 2 shows, our protocol only causes 11% extra transmission delay.

Table 1. The network life time (minutes) of the always active scheme and our protocol

Number of nodes	Always active	Our protocol	Improved rate
529	323	641	98%

Table 2. The transmission delay (hops) of the always active scheme and our protocols

Number of nodes	Always active	Our protocol	Extra delay rate
3136	27.2	30.2	11%

6 Conclusions

In this paper, we have proposed a power management protocol based on the idea of connected dominating set for regular WSNs. Different from previous works (SPAN [7]and GAF [8]), we choose several different connected dominating sets for regular WSN topologies and balance the power consumption of each node. The nodes in each of the different connected dominating sets will switch to active mode in turn to serve other nodes in power saving mode according to their own *id*s. Performance analysis has shown that the ratio of active nodes of our protocol is near optimal and much lower than those of GAF and SPAN, those are designed for high density irregular networks. Simulation results have shown that our protocol can conserve lots of power and greatly extend the network life time with a reasonable extra transmission delay.

References

1. Schurgers, C., Tsiatsis, V., Ganeriwal, S., Srivastava, M.: Topology management for sensor networks: Exploiting latency and density. Proceedings of ACM international Symposium on Mobile Ad Hoc Networking and Computing (2002) 135–145
2. Slijepcevic, S., Potkonjak, M.: Power-efficient organization of wireless sensor networks. Proceedings of IEEE International Conference on Communications (2001) 472–476
3. Pei, G., Chien, C.: Low power tdma in large wireless sensor networks. Proceedings of Military Communications Conference **1** (2001) 347–351
4. Ye, W., Heidemann, J., Estrin, D.: An energy-efficient mac protocol for wireless sensor networks. Proceedings of IEEE INFOCOM (2002) 1567–1576
5. Rajendran, V., Obraczka, K., Garcia-Luna-Aceves, J.: Energy-efficient, collision-free medium access control for wireless sensor networks. Proceedings of International Conference on Embedded Networked Sensor Systems (2003) 181–192
6. Dam, T., Langendoen, K.: An adaptive energy-efficient mac protocol for wireless sensor networks. Proceedings of International Conference on Embedded Networked Sensor Systems (2003) 171–180
7. Chen, B., Jamieson, K., Balakrishnan, H., Morris, R.: Span: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. Wireless Networks **8** (2002) 281–294
8. Xu, Y., Heidemann, J., Estrin, D.: Geography-informed Energy Conservation for Ad Hoc Routing. Proceedings of International Conference on Mobile Computing and Networking (2001) 70–84
9. Heinzelman, W.R., Chandrakasan, A., Balakrishnan, H.: Energy-efficient communication protocol for wireless microsensor networks. Proceedings of Hawaii International Conference (2000) 3005–3014
10. Elson, J., Girod, L., Estrin, D.: Fine-grain network synchronization using reference broadcasts. Proceedings of symposium on Operating System Design and Implementation (2002)
11. Ganeriwal, S., Kumar, R., Srivastava, M.B.: Timing-sync protocol for sensor networks. Proceedings of International Conference on Embedded Networked Sensor Systems (2003)

Information Fusion for Data Dissemination in Self-Organizing Wireless Sensor Networks*

Eduardo Freire Nakamura[1,2], Carlos Mauricio S. Figueiredo[1,2], and Antonio Alfredo F. Loureiro[1]

[1] Federal University of Minas Gerais – UFMG – Brazil
{nakamura, mauricio, loureiro}@dcc.ufmg.br
[2] Research and Technological Innovation Center – FUCAPI – Brazil
{eduardo.nakamura, mauricio.figueiredo}@fucapi.br

Abstract. Data dissemination is a fundamental task in wireless sensor networks. Because of the radios range limitation and energy consumption constraints, sensor data is commonly disseminated in a multihop fashion (flat networks) through a tree topology. However, to the best of our knowledge none of the current solutions worry about the moment when the dissemination topology needs to be rebuilt. This work addresses such problem introducing the use of information fusion mechanisms, where the traffic is handled as a signal that is filtered and translated into evidences that indicate the likelihood of critical failures occurrence. These evidences are combined by a Dempster-Shafer engine to detect the need for a topology reconstruction. Our solution, called Diffuse, is evaluated through a set of simulations. We conclude that information fusion is a promising approach that can improve the performance of dissemination algorithms for wireless sensor networks by avoiding unnecessary traffic.

1 Introduction

Wireless Sensor Networks (WSNs) [1] define a special class of *ad hoc* network composed of a large number of nodes with sensing capability. Wireless sensor networks are strongly limited regarding power resources and computational capacity. In addition, these networks need to autonomously adapt themselves to eventual changes resulted from external interventions, such as topological changes, reaction to a detected event, or requests performed by an external entity.

The main objective of a WSN is to gather data from the environment and deliver it to a sink node for further processing. Consequently, data dissemination is a fundamental task, which is commonly performed in a multihop fashion in flat networks due to the radios range limitation and energy consumption constraints.

Data dissemination can be performed as a continuous task where the application continuously receives data perceived from the environment [2]. Tree

* This work is partially supported by CNPq, Brazilian Research Council, under process 55 2111/02-3.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 585–593, 2005.

topologies are frequently used to disseminate data in a continuous flat sensor network. Even Directed Diffusion [3] provides a tree-like variant called One-Phase Pull Diffusion [4]. In this algorithm there is no exploratory data; the sink node simply disseminates its interest, and the source nodes send data to their neighbors that firstly sent the interest (therefore, building a dissemination tree). Although the tree topology is explored by different solutions [5, 6], none of them consider when the topology needs to be rebuilt. In [6], Zhou and Krishnamachari suggest to periodically rebuild the network topology to recover from eventual node failures.

The correct moment when the network needs to be rebuilt is the problem addressed in this work. We propose Diffuse, which is a topology building engine that uses information fusion mechanisms to implement a feasible solution. Information fusion is commonly used in detection and classification tasks in robotics and military applications [7]. Lately, these mechanisms have been used in applications such as intrusion [8] and Denial of Service (DoS) detection [9]. Within the WSNs domain, simple fusion technics (e.g., aggregation methods such as *maximum* and *minimum*) have been used to reduce data traffic and save energy [3, 5].

This work provides two major contributions. First, the improvement of the dissemination algorithms reducing unnecessary topology constructions. Second, the expansion of the information fusion applicability. As a side effect, we show how to reason about matching information fusion to a specific problem.

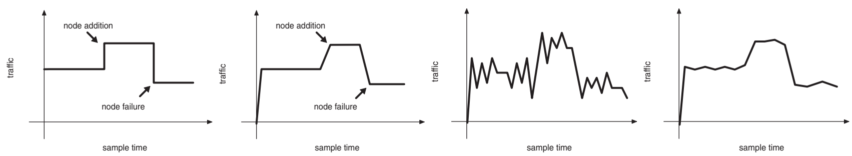
The remaining of this paper is organized as follows. In Section 2 we provide a formal definition for the problem we address, limit its scope, and analyze it. Section 3 presents Diffuse and details how we match the information fusion mechanisms to our specific problem. In Section 4 Diffuse is evaluated through a set of simulation experiments. Finally, in Section 5 we present our conclusions and some future works.

2 Problem Investigation

The traffic can be handled as a discrete signal $\delta(t)$ that computes the amount of packets received during a time interval $\mathbf{S}$ (sample rate). For continuous networks the data rate $\mathbf{R}$ remains the same for all nodes during the network lifetime. Although the measured traffic is still vulnerable to noise (due to packet losses, queue delays, and clock-drifts), making $\mathbf{S} \equiv \mathbf{R}$ should provide a good estimate of the data traffic in continuous networks.

Ideally, in a continuous data gathering scenario, the traffic remains unchanged until new nodes are added – leading to a higher traffic level – or failures happen – leading to a lower traffic level (Fig. 1(a)). Although, the ideal measured traffic (Fig. 1(b)) may not be reached due to the embedded noise (Fig. 1(c)), the raw measure can be filtered to provide a more realistic estimate (Fig. 1(d)).

The impact of failures depends on the activity load of the failing node. The failure of a leaf node is called a *peripheral failure*, while the failure of a relay node is called a *routing failure*. In this case, the greater the disconnected subtree, the



(a) (b) (c) (d)

Fig. 1. Behavior of the traffic signal: (a) Ideal traffic signal, (b) Ideal measured signal, (c) Actual measured signal, and (d) Filtered measured signal

more critical the failure. A routing failure must result in a greater traffic decay than a peripheral failure. In fact, great traffic decays possibly mean that few routing failures or several peripheral failures occurred. Thus, the traffic signal $\delta(t)$ should provide enough information to decide when the network topology needs to be rebuilt.

3 Diffuse: A Topology Building Engine

Diffuse is a topology building engine that uses information fusion mechanisms to combine data and features to detect when it is necessary to rebuild the dissemination topology. Diffuse is composed of the elements described bellow.

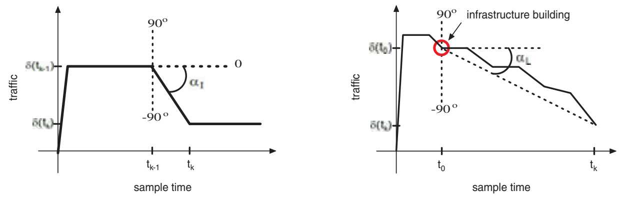
3.1 Signal Processor

The traffic signal behavior (Fig. 1(a)) is very similar to a step function. Consequently, the moving average filter (MAF) is the best choice to clean the traffic signal because MAF is optimal for reducing the random white noise while keeping the sharpest step response [10]. The filter computes the arithmetic mean of a number of input measures to produce each point of the output signal. The filter has one configurable parameter that is the filter's window m, which is the number of input samples that are fused into one output sample. The lower the m, the sharper the step edge; and the greater the m, the cleaner the signal. The window size m must be chosen based on the traffic noise profile and on the desired response time.

3.2 Feature Extractor

Once the traffic measure is filtered, we extract two features from the signal: *instant decay* and *long term decay*. The instant decay evaluates how the traffic changed since the last sample (observation). The long term decay shows how the traffic changed since the last topology reconstruction.

Instant Decay. Fig. 2(a) illustrates how the instant decay is computed. Given two samples in a sequence, t_{k-1} and t_k, we define the **instant decay** as



(a) Instant decay (b) Long term decay

Fig. 2. Measured traffic

$$\phi_i = \frac{\alpha_i}{\alpha_{imax}} = \left(\arctan\frac{\delta(t_{k-1})}{t_k - t_{k-1}}\right) \times \left(\arctan\frac{\delta(t_k) - \delta(t_{k-1})}{t_k - t_{k-1}}\right)^{-1} \qquad (1)$$

where α_i is called **instant decay angle** and α_{imax} is called **maximum instant decay angle**. A traffic increase occurs when $\phi_i > 0$, and traffic decrease occurs when $\phi_i < 0$.

Long Term Decay. Fig. 2(b) depicts how the long term decay is computed. Given the current sample, t_k, we define the **long term decay** as

$$\phi_l = \frac{\alpha_l}{\alpha_{lmax}} = \left(\arctan\frac{\delta(t_k) - \delta(t_0)}{t_k - t_0}\right) \times \left(\arctan\frac{\delta(t_0)}{t_k - t_0}\right)^{-1} \qquad (2)$$

where α_l is called **long term decay angle** and α_{lmax} is called **maximum long term decay angle**. Again, $\phi_l > 0$ means a traffic increase, and $\phi_l < 0$ means a traffic decrease.

3.3 State Estimator

The fusion method used to infer the network state is the Dempster-Shafer Inference [11] because it generalizes the Bayesian theory. Additionally, compared to the Bayesian Inference, Dempster-Shafer is closer to the human perception and reasoning. Important elements of the Dempster-Shafer theory used in this work are: frame of discernment, mass function or basic probability assignment, belief function, plausibility function, and the Dempster-Shafer combination rule. The definitions of these elements can be found in [11].

The network states considered by Diffuse are: NORMAL and CRITICAL. The NORMAL state is used to specify when no failures occur or when only *peripheral failures* occur in the network. The CRITICAL state specifies when a *routing failure* occurs in the network. Thus, our frame of discernment [11] is the set $\Theta = \{\mathsf{NORMAL}, \mathsf{CRITICAL}\}$.

Diffuse translates the traffic features ϕ_i (instant decay) and ϕ_l (long term decay) into evidences. We understand that in the particular case of continuous WSNs, if $-1 \leq \phi_i < 0$, then there is a nonzero probability that a *routing failure* occurred, and if $\phi_i \geq 0$, then we assume that no failure at all occurred. Thus, we define the mass function [11] $m_i : 2^{\Theta} \rightarrow [0, 1]$ as follows:

$$m_i(\mathsf{CRITICAL}) = \begin{cases} 0, & \phi_i \geq 0; \\ |\phi_i|^w, & -1 \leq \phi_i < 0,\ w > 0,\ w \in \mathbb{R}. \end{cases}$$
$$m_i(\mathsf{NORMAL}) = 1 - m_i(\mathsf{CRITICAL})$$

where w is called the decay weight. Assuming that these observations are also valid for the long term decay, we can similarly define the mass function $m_l : 2^{\Theta} \rightarrow [0, 1]$ as

$$m_l(\mathsf{CRITICAL}) = \begin{cases} 0, & \phi_l \geq 0; \\ |\phi_l|^w, & -1 \leq \phi_l < 0,\ w > 0,\ w \in \mathbb{R}. \end{cases}$$
$$m_l(\mathsf{NORMAL}) = 1 - m_l(\mathsf{CRITICAL})$$

The network state is estimated applying the Dempster-Shafer rule [11] to fuse the probabilities assigned by m_i and m_l into $m_i \oplus m_j$. Then, the plausibility [11], and the belief [11] of each hypothesis (NORMAL and CRITICAL) regarding $m_i \oplus m_l$ is computed. The most plausible state is chosen as the actual network state. When both states are equally plausible, the most believable state is chosen. If both states are equally plausible and believable the NORMAL state is chosen.

3.4 Decision Maker

If the network concludes that the system's state is CRITICAL, the sink node rebuilds the network topology trying to find alternate routes to nodes which have stopped delivering data due to a routing failure. Otherwise, if the system is in the NORMAL state, nothing is done.

4 Experiments

This section presents the methodology (scenarios, failure model, and metrics) used to evaluate the use of Diffuse, and the results of our experiments.

4.1 Methodology

Diffuse is evaluated through simulations that compare its behavior with the periodic rebuilding (periodic interest dissemination) of One-Phase Pull Diffusion [4]. We empirically determined $m = 5$ as the best filter window for our traffic profile. We show the behavior of Diffuse with values of $w \in \{\frac{1}{1}, \frac{1}{3}, \frac{1}{5}, \frac{1}{7}\}$, which is the decay weight (Section 3.3). The experiments use the ns-2 simulator [12]. For each experiment, 33 different seeds are used. The graphs shown in this section represent the arithmetic mean and the confidence interval for 95% of confidence.

The simulation parameters are based on the Mica2 sensor node [13]: transmission, reception, and sensing are 45.0mW, 24.0mW and 15.0mW, respectively; the bandwidth is 19200 bps; and the communication radius is 40m. The MAC layer uses the 802.11 standard. In all scenarios the sink is placed in the bottom left corner (0,0) of the sensor field. Both, data packets and control packets, have 20 bytes. The chosen data rate is one packet each 20s. For One-Phase Pull Diffusion (1PP Diffusion), interests are disseminated each 200s. The simulation time is 4000s for all scenarios.

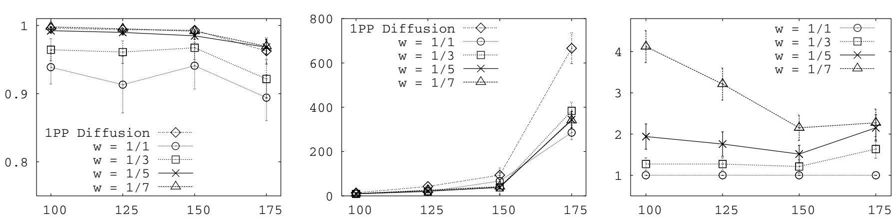
Reliability is evaluated using an independent failures model. In this model failures occur as a Poisson process where the time between successive failures is represented by an independent exponential random variable with constant rate λ (the failure rate measured in failures per seconds).

The metrics chosen to evaluate Diffuse are: delivery ratio, queue drops, and number of constructions. The delivery ratio provides an efficacy measure regarding the network ability to deliver sensed data. Queue drops allow the evaluation of the impact of network constructions in the overall traffic. The number of constructions, associated with the delivery ratio, provides means to evaluate how often we need to rebuild the network topology.

4.2 Results

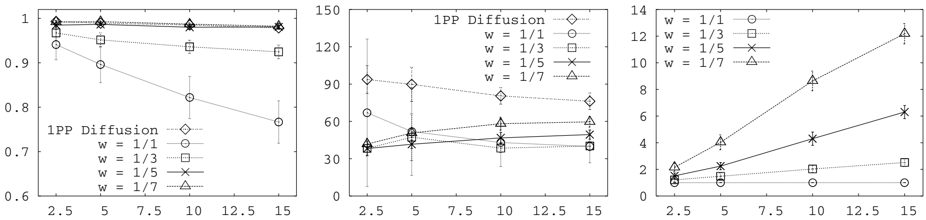
Scalability. is evaluated increasing the network size from 100 to 175 nodes with a constant failure rate ($\lambda = 0.0025$ failures/s), and the results are depicted in Fig. 3. Regarding the delivery ratio (Fig. 3(a)), using Diffuse with $w = 1/5$ and $w = 1/7$ the network delivers nearly as many packets as One-Phase Pull Diffusion (1PP Diffusion), and when the network size is large (175 nodes) One-Phase Pull Diffusion starts to suffer a greater saturation because of the extra topology constructions.

Increasing the network size the overall traffic also increases (specially in the nodes closer to the sink). Consequently, the number of queue drops (Fig. 3(b)) increases, specially when the network size is large (175 nodes). In addition, the impact of the extra topology constructions performed by One-Phase Pull Diffusion can be seen in Fig. 3(b) when the network drops more packets due to queue overflow.



(a) Delivery ratio (y-axis) (b) Queue Drops (y-axis) (c) Constructions (y-axis)

Fig. 3. Scalability using Diffuse in a scenario with independent failures. The x-axis for all graphs is the network size (number of nodes)



(a) Delivery ratio (y-axis) (b) Queue Drops (y-axis) (c) Constructions (y-axis)

Fig. 4. Reliability using Diffuse in a scenario with independent failures. The x-axis for all graphs is the failure rate in 0.001 failures/s

Regarding the network constructions depicted in Fig. 3(c), the number of topology constructions tends to decrease with the network size when $w = 1/5$ and $w = 1/7$. This occurs because when the traffic increases the impact of one failure is reduced demanding less constructions. However, when the network begins to loose more packets in the queues (175 nodes) these drops count as failures demanding more topology constructions (Fig. 3(c)). As a general result, from Fig. 3 we can conclude that for the evaluated scenarios with one additional network construction (when $w = 1/5$ in Fig. 3(c)) the network performs as good as One-Phase Pull Diffusion (Fig. 3(a)) being less affected by the traffic caused by unnecessary constructions (Figure 3(b)). This represents a reduction of nearly 90% in the number of constructions performed by the network

Reliability. is evaluated with failure rates (λ) equal to 0.0025, 0.005, 0.01, and 0.015 failures/s, making the network size constant (150 nodes). The results are shown in Fig. 4. The delivery ratio (Fig. 4(a)), using Diffuse with $w = 1/5$ and with $w = 1/7$ is practically the same of One-Phase Pull Diffusion. Furthermore, Diffuse with $w = 1/5$ and with $w = 1/7$, and One-Phase Pull Diffusion successfully recover from failures making the delivery ratio almost constant independently from the failure rate.

Regarding the queue drops in Fig. 4(b), as a result of the traffic decrease, One-Phase Pull Diffusion begins to drop less packets when the number of failures increases. On the other hand, with $w = 1/5$ and with $w = 1/7$, Diffuse drops more packets as the number of failures increases. This is a result of the number of topology constructions that increases quickly with the number of failures (Fig. 4(c)). However, even with the increasing number of constructions, Diffuse still rebuilds the network topology fewer times than One-Phase Pull Diffusion, which also results in fewer queue drops.

5 Conclusion and Future Work

This work proposes and implements a topology building engine, called Diffuse, that adopts information fusion mechanisms (Moving Average Filter and Dempster-Shafer Inference) to determine when the dissemination infrastructure needs to be rebuilt based only on the measured traffic. This approach showed to be very efficient in avoiding unnecessary topology constructions. We showed that in some cases, only one additional construction is enough to guarantee the data delivery (a reduction of 90% in the number of topology constructions). Other contributions include the illustration of information fusion mechanisms being used in other application domains (dissemination algorithms) and the reasoning about how we can match information fusion mechanisms to the requirements and limitations of a specific problem.

This work evaluates flat sensor networks with continuous data gathering. In the next step, Diffuse will be adapted to aggregating sensor networks, and event-driven sensor networks. New challenges are introduced when data aggregation is performed as the overall traffic is naturally reduced. For the event-driven networks the challenge is even greater since the traffic behavior is more complex – it supposedly increases (as events are detected) and decreases (as events stop being detected), independently from network failures.

References

1. Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cyirci, E.: Wireless sensor networks: A survey. Computer Networks **38** (2002) 393–422
2. Tilak, S., Abu-Ghazaleh, N.B., Heinzelman, W.: A taxonomy of wireless micro-sensor network models. Mobile Computing and Communications Review **6** (2002) 28–36
3. Intanagonwiwat, C., Govindan, R., Estrin, D.: Directed diffusion: A scalable and robust communication paradigm for sensor networks. In: Proc. of the 6th ACM International Conference on Mobile Computing and Networking, Boston, USA (2000) 56–67
4. Heidemann, J., Silva, F., Estrin, D.: Matching data dissemination algorithms to application requirements. In: Proc. of the 1st International Conference on Embedded Networked Sensor Systems, Los Angeles, USA (2003) 218–229
5. Krishanamachari, B., Estrin, D., Wicker, S.: The impact of data aggregation in wireless sensor networks. In: Proc. of the 22nd International Conference on Distributed Computing Systems, Vienna, Austria (2002) 575–578
6. Zhou, C., Krishnamachari, B.: Localized topology generation mechanisms for self-configuring sensor networks. In: Proceedings of the IEEE GLOBECOM 2003. Volume 22., San Francisco, USA (2003) 1269–1273
7. Brooks, R.R., Iyengar, S.S.: Multi-Sensor Fusion: Fundamentals and Applications with Software. Prentice-Hall, Inc., Upper Saddle River, USA (1998)
8. Bass, T.: Intrusion detection systems and multisensor data fusion. Communications of the ACM **43** (2000) 99–105

9. Siaterlis, C., Maglaris, B.: Towards multisensor data fusion for DoS detection. In: Proc. of the 19th ACM Symposium on Applied Computing, Nicosia, Cyprus (2004) 439–446
10. Smith, S.W.: The Scientist and Engineer's Guide to Digital Signal Processing. 2nd edn. California Technical Publishing, San Diego, USA (1999)
11. Yager, R.R., Kacprzyk, J., Fedrizzi, M.: Advances in the Dempster-Shafer Theory of Evidence. John Wiley & Sons, Inc. (1994)
12. NS-2: (The network simulator) http://www.isi.edu/nsnam/ns/.
13. Crossbow: (Mica2) http://www.xbow.com.

An Efficient Protocol for Setting Up a Data Dissemination Path in Wireless Sensor Networks*

Dongkyun Kim[1] and Gi-Chul Yoo[2]

[1] Department of Computer Engineering,
Kyungpook National University, Daegu, Korea
dongkyun@knu.ac.kr
[2] Digital Media Lab., LG Electronics, Seoul, Korea
gcyoo@lge.com

Abstract. Recently, the interest in sensor network has increased with the advanced technologies in the field of digital signal processing, sensing and wireless networking devices. In this paper, an efficient protocol to set up a data dissemination path shared among multiple sink nodes is proposed for a special sensor network, where the source sensor nodes update their sensed data according to various desired update rates requested by sink nodes. We modify and enhance the basic SAFE(Sinks Accessing data From Environments) protocol to minimize the increased amount of data update rate at the intermediate nodes over a dissemination path. By using GloMoSim simulator, we show that our ESAFE (Enhanced SAFE) protocol outperforms the basic SAFE.

1 Introduction

The current advanced technology in the fields of digital signal processing, sensing and wireless networking devices enables sensor applications such as the remote monitoring of an interested event to be used easily within the near future [1]. In the sensor networks, sensor nodes are generally scattered in an interested area to transmit an event or data sensed at a source node to a sink node that is required to process it. Due to the high cost of wiring sensor nodes, they are using wireless links with short-range radio capability. For the propagation of a sensed event, the nodes rely on multi-hop wireless forwarding services. A great deal of research [4] [5] [6] assumed possible sensor applications in which the sensor network consists of the source nodes which can periodically update sensed data and a lot of sink nodes that require data in the network.

Previous research works require intermediate sensor nodes to update data at the same rate at which the source sensor node updates the sensed data. However, we assume that the rates at which the sink nodes want to obtain sensed data,

* This work was partially supported by Winitech Co., Ltd.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 594–602, 2005.

are different among the sink nodes because the interest of one sink node in the sensed data can also be different from others. In other words, we allow each sink node itself to specify the desired data update rate. In this context, the source node will update the sensed data at the maximum rate of the rates requested by the sink nodes. All intermediate sensor nodes, over the path to a sink node, however, do not need to update the data at the maximum rate, due to the existence of different paths toward each sink node with its own desired update rate. Therefore, it is enough that the intermediate nodes set the date update rate to the maximum rate of the rates requested by the sink nodes, which are reachable via themselves.

In particular, we consider a sensor network consisting of stationary source sensor nodes or sensor nodes moving at very low speeds. For the environment, to the best of our knowledge, the SAFE (Sinks Accessing data From Environments) protocol was the first trial as a data dissemination protocol for the application which requires the sink nodes to require the sensed data from a source node at their different interested update rates [2] [3]. The SAFE protocol attempts to utilize the tree structure. Each sink node attaches itself to the best branch of a tree rooted at a source node. Its Query and PathSetup procedures enable energy to be saved through a data delivery path shared among multiple sink nodes with common interests. The procedures allow intermediate nodes over a tree to update the sensed data according to a rate requested by a sink node. However, it is likely that the SAFE protocol forces nodes over a tree to increase their update rates unnecessarily, which results in dissipating a great deal of energy during the updating process. In this paper, we propose an Enhanced SAFE, called ESAFE, to avoid the unnecessary increase of the updating rate at the intermediate tree nodes.

The rest of this paper is organized as follows. In Section 2, we describe the basic SAFE protocol upon which our proposed scheme is based. Section 3 shows our proposed ESAFE scheme in detail, which is followed by the performance evaluation in Section 4. Finally, some concluding remarks are given in Section 5 with future plans.

2 Basic SAFE Protocol

The SAFE protocol consists of two processes: Query Transfer and Dissemination Path Setup. In the Query Transfer process, a sink node with a need to update data from a source sensor node, transfers a query message to its neighbor nodes with its location and the desired data update rate. Basically, the SAFE assumes that all sink nodes know the location of their source sensor nodes. Thus, using the location information, the query message will be forwarded to the nodes which are gradually located nearer to the source sensor node by utilizing SPEED [7] or LAR [8] routing protocol. During the query propagation, the intermediate nodes record a next-hop node, that is, the reverse path information toward the sink node. Consequently, the message reaches the source sensor node or an intermediate node located on an already formed dissemination tree. The source sensor node responds to the query message by unicasting a PathSetup message, which reaches the inquiring sink node using the recorded next-hop node information. Similarly, the intermediate sensor node on the dissemination path sends a JunctionInfo message to the sink node.

In the Dissemination Path Setup process, when a sink node receives a PathSetup or JunctionInfo message, it responds to the message by sending an Ack message to the source sensor node when it has not received any other PathSetup or JunctionInfo message during a timeout interval. The Ack message is used to confirm a successful dissemination path. However, it is highly possible that the sink nodes receive several PathSetup messages traversed over different routes or JunctionInfo messages generated by some intermediate sensor nodes, which are located on the existing dissemination paths during a certain amount of time. In order to minimize message exchanges over the network, the authors maintain that the best subscription locus is one that can update the sink node with the smallest number of extra messages. According to the SAFE protocol, the updating overhead is defined as a subscription cost C of a junction j when a sink node m wants data updates from a source sensor node, s, through a junction node, j, as follows:

$$C(m,s,j) = \begin{cases} d(s,j) * (r_m - r_j) + d(j,m) * r_m & \text{if } r_m > r_j \\ d(j,m) * r_m & \text{otherwise} \end{cases},$$

where $d(a,b)$ quantifies the hop distance from node a to b, and r_a denotes the update rate requested by and thus, available to node a. In this context, the sink node sends a Subscribe message to a junction sensor node or an Ack message to the source sensor node depending on the best subscription point. Refer to the SAFE mechanism for a more detailed description [2] [3].

3 ESAFE: Enhanced SAFE Protocol

3.1 Motivation

The SAFE protocol is known as an efficient data dissemination protocol utilizing a tree structure, which allows a data delivery path to be shared among multiple sinks. The protocol is also suitable for achieving energy efficiency as well as scalability, both of which the authors hold are crucial for large-scale, battery-powered sensor networks. However, during the expansion of a data delivery path, when there exist different update rates demanded by the multiple sink nodes, a possible tree structure can be formed as shown in Figure 1. According to the SAFE protocol applied, we can observe that the intermediate sensor nodes located nearer to the source sensor node update data at a higher rate than those farther away from the source sensor node. At first, a sink node (*node a*) demands a data update rate, $r = 3$ which means that the number of data updates that the sink node requires in a unit time is 3 (Figure 1 (a)). Second, *node b* attaches its branch to the existing dissemination path with a required update rate, 6 and therefore, the update rates from the source node to *noded* are all 6

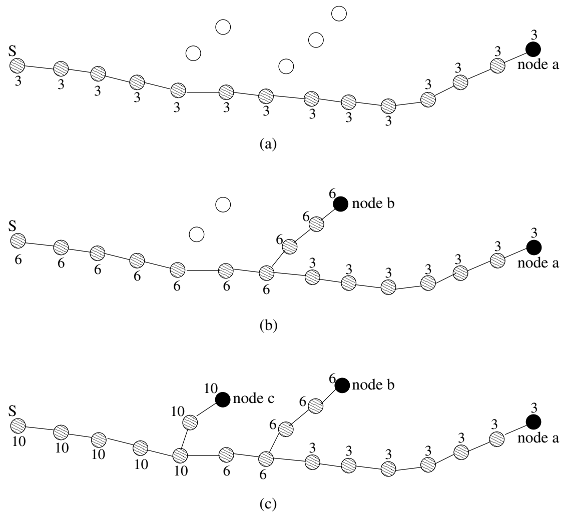


Fig. 1. An Illustrative Example of a Dissemination Path for the SAFE Protocol

(Figure 1 (b)). Finally, when *node c* needs to update sensed data at rate 10, the final dissemination path is shown in Figure 1 (c).

Although it is likely that the intermediate sensor nodes, over a data dissemination path, perform data updates at different rates as shown in Figure 1, the $d(s, j) * (r_m - r_j)$ term in the cost function $C(m, s, j)$ of the SAFE protocol considers the increased amount of the update rate based on the current update rate of only a junction node j, according to the demand by a sink node m. The cost function $C(m, s, j)$ used in the SAFE protocol ignores the fact that the increased amount of an update rate can be different among intermediate nodes over an existing dissemination path.

It is enough that Figure 2 shows the drawback of the SAFE protocol. Suppose that *node m* needs to update its data with a required update rate, $r = 6$. As shown in Figure 2, *node m* receives two JoinInfo messages from the junction nodes, $j1$ and $j2$. Therefore, *node m* should select the best junction point with respect to energy dissipation. According to the SAFE protocol's cost function, $d(s, j1) * (r_m - r_{j1}) + d(j1, m) * r_m = 9 * 3 + 4 * 6 = 51$ and $d(s, j2) * (r_m - r_{j2}) + d(j2, m) * r_m = 6 * 4 + 4 * 6 = 48$. The SAFE protocol, therefore, selects *pathb*.

In order to attach the *node m* to the existing path, however, *path a* is actually better. When *path a* is selected, it is enough to increase additionally the data update rates of nodes $n1, n2$ and $n3$ by 3 (totally, 3+3+3 = 9). In contrast, when *path b* is selected, we should increase additionally the data update rates of nodes $n4$ and $n5$ by 3 and furthermore, those of nodes $n6$ and $n7$ by 4 (totally, 3+3+4+4 = 14). Therefore, the basic SAFE protocol is not suitable for addressing these cases since it adheres to *path b* instead of *path a*.

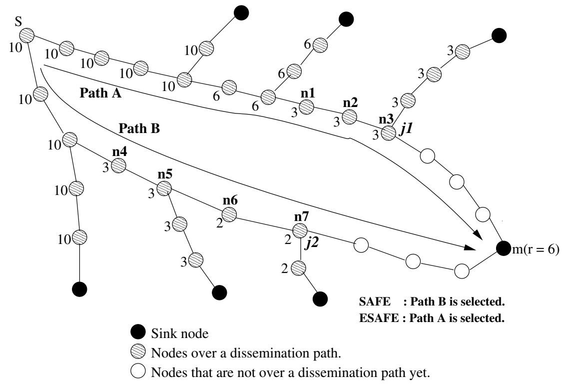


Fig. 2. SAFE's Wrong Attachment to an Existing Path

3.2 Description of Our ESAFE Protocol

In this section, we propose an enhanced SAFE protocol to address the problem mentioned above. A Query message, containing a desired data update rate from a sink node m, is basically flooded toward the source sensor node by using location information as in SAFE protocol using SPEED or LAR scheme. The Query message reaches a source sensor node or a junction node j over an existing dissemination path. Unlike the SAFE protocol, even though the Query message meets a junction node, the Query message should continue to traverse the existing path toward the source sensor node, until it reaches a node (denoted by *pivot_node*) which updates the sensed data at a higher rate than the update rate desired by a sink node, m. The *pivot_node* unicasts an Attach message to the sink node, m. During the unicasting of the Attach message, the increase of the update rate at each intermediate sensor node should be accumulated in the Attach message. When the sink node m gathers multiple Attach messages from the multiple *pivot_nodes* during an interval, the node m selects the best one and sends a Subscribe message to a corresponding *pivot_node*. While the Subscribe message is propagated to the *pivot_node*, the intermediate nodes increase their update rates accordingly. Thus, if we consider a generic route $r_d = pv, n_1, ..., n_j, ..., n_{m-1}, n_m$, where pv is a pivot_node and n_m is a sink node, the total additionally increased update rate is calculated as: $R(r_d) = \sum_{i=1}^{m-1} (r_m - r_i)$, where n_j is a junction node, r_i is a current update rate at node n_i and furthermore, r_is for $i = j + 1$ to $i = m - 1$ are all zero. The optimal route r_O satisfies the following condition:

$$R(r_O) = \min_{r_j \in r_*} R(r_j)$$

where r_* is the set of all possible routes. Note that the *pivot_node* can be a source sensor node when there exists none that updates data at a higher rate than the update rate desired by a sink node.

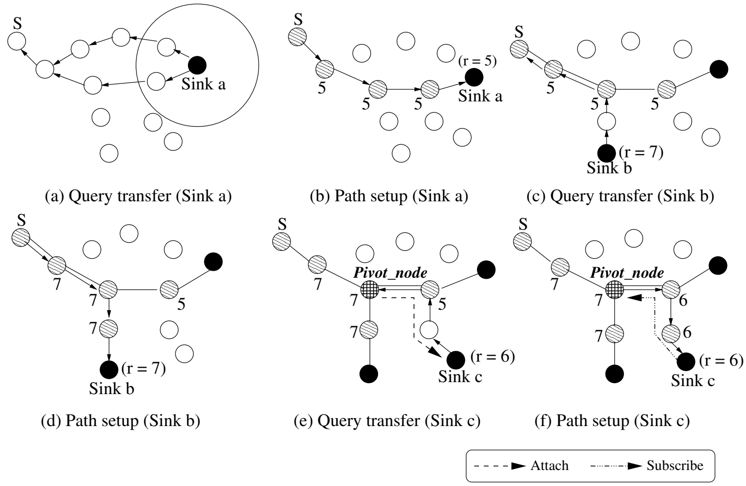


Fig. 3. An Illustrative Example of our ESAFE protocol

3.3 The ESAFE's Illustrative Example

Figure 3 shows an example of our ESAFE protocol's behavior. When sink node *a* needs to update data at an update rate of 5, an initial path is set up as shown in Figure 3 (b). When sink node *b* wants to attach itself to the path at an update rate of 7, the Query message will reach the source sensor because there is no intermediate node with a rate higher than 7 over the dissemination path (Figure 3 (c) and (d)). When the sink node *c* needs to receive the sensed data from the source sensor node at rate 6, the response to its Query message, an Attach message will be sent by a *pivot_node* because the update rate of the *pivot_node* is higher than the requested rate, which is, $7 > 6$ (Figure 3 (e) and (f)). Finally, the sink node will successfully attach itself to the path by sending a Subscribe message to the *pivot_node*.

Note that although both SAFE protocol and ESAFE protocol are developed for a sensor network consisting of stationary sensor nodes or sensor nodes moving at very low speeds, the periodic reconfiguration of the dissemination paths allows them to be used in a dynamic network environment where sensors are moving around.

4 Performance Evaluation

Our ESAFE protocol was implemented using GloMoSim [9]. For simulation, we assumed that all sensor nodes are equipped with IEEE 802.11 network interface cards using IEEE 802.11 CSMA/CA protocol. In addition, we used SPEED [7] as the underlying routing protocol. To create a sensor network, 100 sensor nodes are put over a 3000 m x 3000 m sensor area. To compare our ESAFE to the SAFE protocol, we also used the same simulation parameters that SAFE used (see [2]).

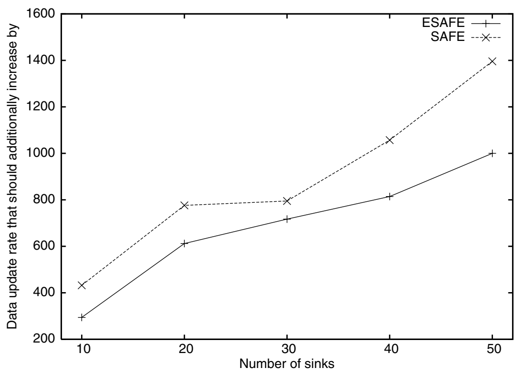


Fig. 4. The sum of update rates that should additionally increase by

The Query inter-arrival times used an exponential distribution ($\mu = 1 - 5$ sec). In particular, a low rate radio bandwidth such as 200 Kbps, was assumed due to the limit of the current sensor's wireless technology.

Both SAFE protocol and ESAFE protocol are suitable for a sensor network with stationary sensor nodes or nodes with very low mobility. In order to cope with a dynamic sensor network, all sink nodes attempt to send Query messages periodically for the purpose of path reconfiguration. All simulation measures were summed every periodic reconfiguration. We used an average of 10 runs for plotting the simulation figures. First, we measured the sum of the increase of the update rates at all intermediate sensor nodes located over an existing dissemination path, when the sink nodes attempt to obtain sensed data. Compared with the SAFE protocol, our ESAFE protocol is superior because it can select a path with a minimum additional increase over an end-to-end path from a source node to a sink node, while satisfying the update rate required by the sink node (Figure 4). Although the SAFE protocol takes into account the hop-distance between the source sensor node and the junction node, it considers the increase of the data update rate at only the junction node. It results in its performance degradation. In addition, both protocols show that a large number of sink nodes make the increased amount of data update rates become higher because the dissemination path is expanded due to many trials for attachments. Second, we investigated the sum of all nodes' update rates over a dissemination path for two protocols. As mentioned before in the previous simulation, our ESAFE protocol tries to minimize the increase of the data update rate at each intermediate node, as well as attempts to satisfy all the sink node demands. Therefore, the ESAFE protocol serves successfully all data update rates, which are desired by all sink nodes, with the minimum update rates more efficiently than the SAFE protocol (Figure 5). This simulation means that we could simply compare two protocols in terms of energy consumption, even though we did not measure the amount of energy in Joules. An unnecessary increase of updated rates causes more energy consumption. Therefore, our ESAFE protocol is superior to the SAFE protocol from an energy's perspective. For both protocols, as the number of sink nodes

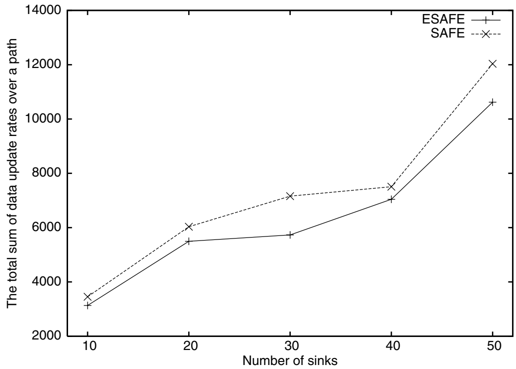


Fig. 5. The sum of all nodes' update rates over a dissemination path

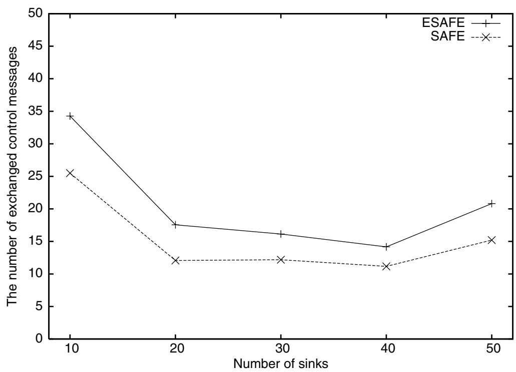


Fig. 6. The average number of exchanged control messages

increases, the amount of data update rates over nodes is also increasing because the scale of the dissemination path is expanded. Finally, we performed a comparison with respect to the average number of exchanged control messages when a sink node attaches itself to the dissemination path. As our ESAFE requires a *pivot_node*, if any, to respond to the Query message rather than a junction node does, our protocol needs a large number of exchanged control messages as shown in Figure 6. These control messages, however, are required only when a sink node attaches itself to a branch of the dissemination path. However, after this expense of more control messages, more energy are saved during the updating of the sensed data thereafter. In addition, the existence of more sink nodes allows new sink nodes to attach themselves to the nodes nearer to themselves, which are located over a dissemination path. This results in a smaller number of exchanged control messages required.

5 Conclusions

In this paper, an efficient protocol to set up a data dissemination path was proposed. The path is shared among multiple sink nodes. The protocol is applied to a special sensor network, where the source sensor nodes update their sensed data according to various desired update rates demanded by the sink nodes. When selecting a branch point among multiple gathered candidate paths for expanding the dissemination path, the basic SAFE (Sinks Accessing data From Environments) just considered a hop-distance from the source sensor node to a junction node and the increased amount of an update rate at only a junction node. The increase of data update rates of all intermediate nodes, which are located over a dissemination path, should also be considered for computing a cost function. Our ESAFE (Enhanced SAFE) protocol attempts to minimize the increase of the data update rate at an intermediate node over a dissemination path. Our ESAFE protocol outperforms the basic SAFE in terms of less energy consumption with low expense for exchanged control messages needed. We are planning to enable our ESAFE protocol to support sensor nodes with high mobility, which is part of our exciting future research.

References

1. I.F.Akyildiz, W. Su, Y. Sankarasubramanian and E. Cayirci, "A survey on sensor networks," IEEE Communications Magazine, August 2002.
2. S.Kim, S.H. Son, J.A. Stankovic, S. Li, and Y. Choi, "SAFE: A Data Dissemination Protocol for Periodic Updates in Sensor Networks," Data Distribution for Real-Time Systems (DDRTS), Providence, RI, U.S.A., May 2003.
3. S.Kim, S.H. Son, J.A. Stankovic and Y. Choi, "Data Dissemination over Wireless Sensor Networks," IEEE Communication Letters, Sep. 2004.
4. C.Intanagonwiwat, R. Govindan and D. Estrin, "Directed Diffusion: A scalable and robust communication paradigm for sensor networks," ACM/IEEE MOBICOM 2000, Boston, 2000.
5. W. Heizelman, J.Hill and H.Balakrishnan, "Adaptive protocols for information dissemination in wireless sensor networks," ACM/IEEE MOBICOM 1999, Seattle, 1999.
6. F.Ye, H.Luo, J.Cheng, S.Lu and L.Zhang, "A Two-tier data dissemination model for large-scale wireless sensor networks," ACM/IEEE MOBICOM 2002, Atlanta, 2002.
7. T.He, J. Stankovic, C. Lu and T.Abdelzaher, "SPEED: A stateless protocol for real-time communication in sensor networks," ICDCS-23, Providence, 2003.
8. Y. B. Ko and N. H. Viadya, "Location-Aided Routing (LAR) in Mobile Ad Hoc Networks", ACM/IEEE MOBICOM 2000, Boston, 2000.
9. http://pcl.cs.ucla.edu/projects/glomosim/

Active Traffic Monitoring for Heterogeneous Environments

Hélder Veiga, Teresa Pinho, José Luis Oliveira, Rui Valadas,
Paulo Salvador, and António Nogueira

University of Aveiro/Institute of Telecommunications - Campus Santiago, Aveiro, Portugal
{jlo, rv}@det.ua.pt, {hveiga, salvador, nogueira}@av.it.pt

Abstract. The traffic management of IP networks faces increasing challenges, due to the occurrence of sudden and deep traffic variations in the network, which can be mainly attributed to the large diversity of supported applications and services, to the drastic differences in user behaviors, and to the complexity of traffic generation and control mechanisms. In this context, active traffic measurements are particularly important since they allow characterizing essential aspects of network operations, namely the quality of service measured in terms of packet delays and losses.

The main goal of the work presented in this paper is the performance characterization of operational networks consisting in heterogeneous environments including both wired and wireless LANs, using active measurements. We propose a measurement methodology and its corresponding measurement platform. The measurement methodology is based on the One-Way Active Measurement Protocol (OWAMP), a recent proposal from the Internet2 and IETF IPPM groups for measuring delays and losses in a single direction. The measurement platform was implemented, tested and conveniently validated in different network scenarios.

Keywords: Network management, traffic monitoring, active measurement, OWAMP.

1 Introduction

The relevance of traffic monitoring in the global management of IP networks has been growing due to the recent acknowledgment that sudden and deep traffic variations demand for frequent traffic measurements. This peculiar behavior of network traffic can be mainly attributed to the combination of different factors, like the great diversity of supported applications and services, different user's behaviors and the coexistence of different mechanisms for traffic generation and control.

Traffic monitoring systems can be classified in active and passive ones [1], [2], [3]. Passive systems simply perform the analysis of the traffic that flows through the network, without changing it. Usually, they are used to identify the type of protocols involved and to measure one or more characteristics of the traffic that flows through the measurement point, like the average rate, the mean packet size or the duration of the TCP connections. Nowadays, there are several passive monitoring systems, like for example NeTraMet [4] and NetFlow [5]. Active systems insert traffic directly into the network. Usually, they are intended to provide network performance statistics between two distinct measurement

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 603–610, 2005.

points, like for example mean packet delay and packet loss ratio. Those statistics can be one-way statistics, when they refer to a single direction of traffic flow, and round-trip statistics, when they refer to traffic that flows in both directions. Active systems require the synchronization of the involved measurement points, using for example GPS (Global Positioning System) or NTP (Network Time Protocol).

The IETF IPPM (IP Performance Metrics) group established in the last few years a set of recommendations in order to assure that measurement results obtained from different implementations are comparable, namely regarding measurements of one-way packet delays and losses [6], [7]. However, these recommendations do not address the interoperability of the measurement elements, that is, the possibility of having traffic senders and receivers that belong to different administrative domains and are developed by different entities. OWAMP is a proposal for a one-way active measurement protocol that intends to solve this problem [8].

In this work, we intend to perform a set of active measurements in a real operational network consisting in a heterogeneous environment that includes both wired and wireless LANs. Thus, instead of using available tools (like PING, for example), some of them with a limited scope of applications, we have decided to implement a complete measurement platform (freely available at `http://www.av.it.pt/JOWAMP/`). In order to guarantee its compliance with other available platforms, its measurement methodology is based on the OWAMP protocol.

The paper is structured in the following way: section 2 describes the architecture and the operational details of the OWAMP protocol, that forms the basis of the implemented solution; section 3 presents the details of the implemented solution; section 4 presents the active measurements experiments, and their corresponding scenarios, that we want to carry out in this work; section 5 presents and discusses the results obtained from its application to the defined measurement scenarios and, finally, section 6 presents the main conclusions.

2 One-Way Active Measurement Protocol

The One-Way Active Measurement Protocol (OWAMP) is a recent proposal from the Internet2 group, developed under the scope of the End-to-End Performance Initiative project [9], [10], for performing active measurements in a single direction. This proposal is also promoted by the IETF IPPM work group [8].

The OWAMP architecture, shown in figure 1, is based on two inter-dependent protocols, the OWAMP-Control and the OWAMP-Test, that can guarantee a complete isolation between client entities and server entities. The OWAMP-Control protocol runs over TCP and is used to begin and control measurement sessions and to receive their results. At the beginning of each session, there is a negotiation about the sender and receiver addresses, the port numbers that both terminals will use to send and receive test packets, the instant of the session beginning, the session duration, the packets size and the mean interval between two consecutive sent packets (it can follow an exponential distribution, for example).

The OWAMP-Test runs over UDP and is used to exchange test packets between sender and receiver. These packets include a Timestamp field that contains the time

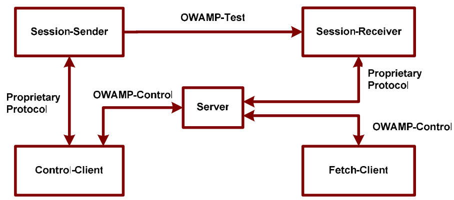


Fig. 1. OWAMP architecture

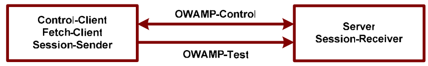


Fig. 2. OWAMP simplified architecture

instant of packet emission. Besides, packets also indicate if the sender is synchronized with some exterior system (using GPS or NTP) and each packet also includes a Sequence Number.

OWAMP supports test packets with service differentiation: DSCP (Differentiated Services Codepoint), PHB ID (Per Hop Behavior Identification Code) or Best-effort. Additionally, OWAMP supports some extra facilities like cypher and authentication for the test and control traffic, intermediary elements called Servers that operate as proxies between measurement points and the exchange of seeds for the generation of random variables that are used in the definition of transmitted test flows. The OWAMP specification also allows the use of proprietary protocols (that can be monolithic or distributed programming interfaces) in all connections that do not compromise interoperability.

The OWAMP architecture includes the following elements:

– Session-Sender: the sender of the test packets;
– Session-Receiver: the receiver of the test packets;
– Server: the entity that is responsible for the global management of the system; it can configure the two terminal elements of the testing network and receive the results of a test session;
– Control-Client: a terminal system that programs demands for test sessions, triggers the beginning of a session set and can also finish one or all ongoing sessions;
– Fetch-Client: a terminal system that triggers the demands for results of test sessions that have already ended or are still running.

A network element can carry out several logical functions at the same time. For example, we can have only two network elements (figure 2): one is carrying out the functions corresponding to a Control-Client, a Fetch-Client and a Session-Sender and the other one is carrying out the functions corresponding to a Server and a Session-Receiver.

3 J-OWAMP: A System Based on OWAMP

In order to create an innovator platform for active measurements, that can also represent a basis for the development and test of new algorithms and models, we built a system designated by J-OWAMP (that stands for Java implementation of OWAMP) that corresponds to the analogous of the OWAMP model. The developed system corresponds

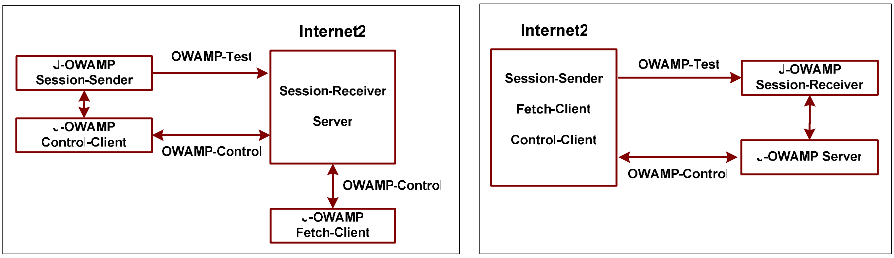


Fig. 3. Configuration of the compliance tests

to the OWAMP most general architecture, depicted in figure 1, allowing the definition of only one client and one server in the network (possibly installed in machines with the highest processing capacity) and the installation of senders and receivers in any machine of the network, which leads to a lower processing impact. In this way, the network manager can perform tests all over the network controlled from a single machine, which is not possible in the simplified scenario of figure 2.

Structure and Implementation - The J-OWAMP system was developed in Java because this language presents a set of favorable characteristics, like semantic simplicity, portability and a set of classes that greatly simplify the construction of distributed applications.

The structure of the system is based on two levels: Messages and Entities. At the Messages level, we developed a set of classes corresponding to each one of the data packets that are exchanged in the OWAMP protocol. A particular class, Packet, is the basis for all messages (derived classes). At the Entities level, a set of classes was developed in order to implement the five elements of the OWAMP architecture: Client, Server, Session-Sender, Session-Receiver and Fetch-Client.

Compliance Tests - In order to guarantee the compliance of the developed system with the OWAMP proposal, we have performed a set of tests involving an implementation (for a UNIX platform) developed by the Internet2 group and publicly available in [9]. The tests were carried out in the private IT-Aveiro network using, in a first experiment, the J-OWAMP modules as the client, monitor and sender modules and using the Internet2 modules as server and receiver modules and, in a second experiment, the J-OWAMP and Internet2 modules in the reverse order (figure 3).

The communication between the J-OWAMP modules (developed in Java language) and the Internet2 modules (developed in C language) was correctly established, in both directions. Using the Ethereal traffic analyzer, we have verified that the control messages and the test packets are correctly exchanged, as specified in the protocol.

4 Measurement Scenarios

Before carrying out active traffic measurements in a real network involving an heterogeneous environment, we have first established a laboratorial measurement setup to test the developed measurement solution in a more controllable environment.

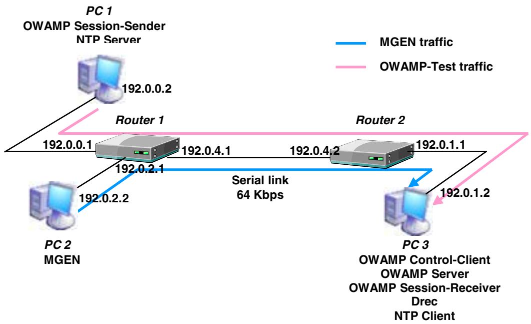


Fig. 4. Network corresponding to the first measurement scenario

Laboratorial Setup - The laboratorial measurement setup is illustrated in figure 4. Routers 1 and 2 are connected through a serial link configured with a transmission capacity of 64 Kb/s and three networks are configured with the following structure: network 192.0.0.0, that contains PC1 running the OWAMP sender; network 192.0.2.0, that contains PC2 running the traffic generator MGEN and network 192.0.1.0 that contains PC3 where we have previously installed the OWAMP client, server and receiver elements as well as a receiver (Drec) of the traffic generated by the MGEN application running on PC2. The service discipline for all queues belonging to the serial interfaces of routers 1 and 2 is FIFO. PCs 1 and 3 are synchronized via NTP.

Using this scenario, we want to measure and study the packet delays that occur in the queuing system of Router 1 as a function of the traffic load in the serial link between Routers 1 and 2. In this way, we have configured the MGEN application to generate traffic according to a Poisson distribution and send it to PC3 (using the serial link). Using the sender installed in PC1 and the receiver installed in PC3 we were able to measure the packet delay values that occurred in the queue of the Router 1 serial interface, for different values of the traffic load. Arrows represented in figure 4 show the directions that are followed by (i) the traffic generated by MGEN and (ii) the test packets generated by the J-OWAMP measurement system.

University of Aveiro (UA) Wireless Network - The network corresponding to this scenario is illustrated in figure 5. In order to evaluate the performance of accessing the UA wireless network from the students' residences, a set of measurements were conducted between a PC located at the laboratory of Institute of Telecommunications (IT), named Lab PC, and another one located at a students' residence of the University campus, named Residence PC. We measured and studied the traffic that flows between the Residence and the Lab PCs, in both directions. The client, server and receiver were installed in the PC that receives the test packets and the sender was installed in the PC responsible for sending the packets. Both PCs are synchronized via NTP. Since Internet access from the student's residences is performed through the UA network, traffic in the downstream direction includes mainly the downloads that are made from the Internet to the residences.

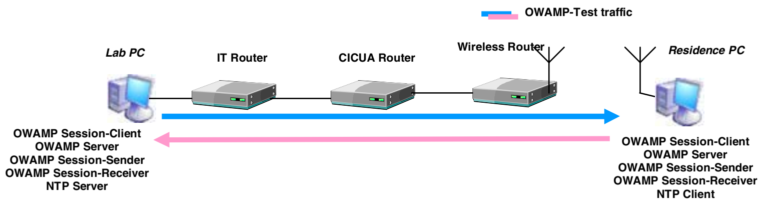


Fig. 5. Network corresponding to the second measurement scenario

All tests were performed in a 24 hours period. In each hour, sets of 10 tests (including both packet delay and loss) were performed, making a total of 240 tests. In each group, the tests beginning instants were separated by 2 minutes. All tests lasted for 1 minute and consisted in sending 60 packets of 14 bytes each, at an average rate of 1 packet/second. In order to conveniently characterize the packet average delay and packet loss ratio, we have calculated 90% confidence intervals based on the 10 average values obtained in each test belonging to a group of 10 tests.

5 Results

First Scenario - Figures 6 and 7 present the results corresponding to the packet delay and packet loss tests, respectively, that were carried out for the first scenario, for different rates of the MGEN generated traffic. From the analysis of the obtained results we can verify that, as expected, there is an increase in packet delays and losses with increasing network load: for network load values that are far from the maximum value supported by the serial link (64 Kb/s) there are no packet losses, however, packet loss values increase very fast as network load approaches the limit load supported by the serial link that connects both routers.

Second Scenario - For this scenario, the results of the average packet delay and packet loss ratio for the upstream direction are presented in figures 8 and 9, respectively,

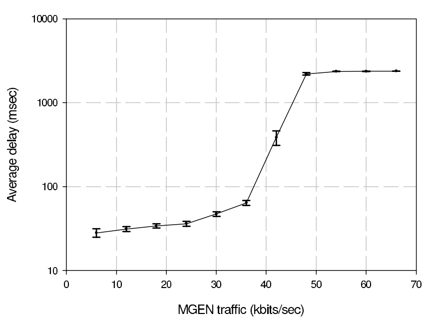


Fig. 6. Results of the first scenario: average packet delay versus MGEN generated traffic

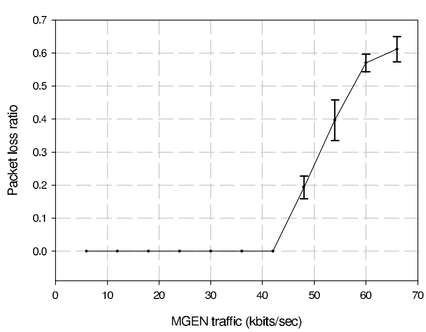


Fig. 7. Results of the first scenario: packet loss ratio versus MGEN generated traffic

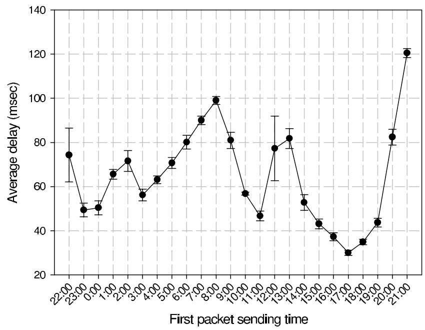


Fig. 8. Results of the second scenario, upstream direction: average packet delay versus first packet sending time

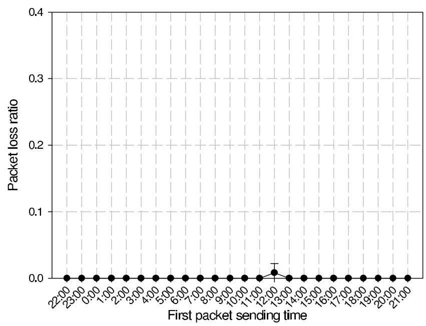


Fig. 9. Results of the second scenario, upstream direction: packet loss ratio versus first packet sending time

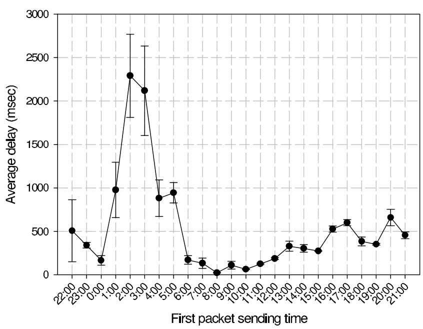


Fig. 10. Results of the second scenario, downstream direction: average packet delay versus first packet sending time

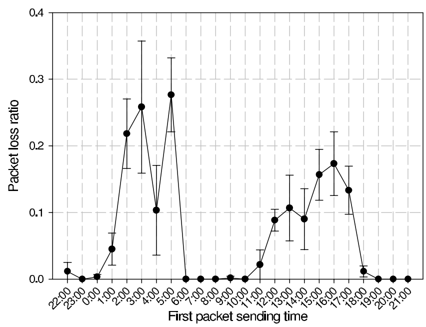


Fig. 11. Results of the second scenario, downstream direction: packet loss ratio versus first packet sending time

and the analogous results corresponding to the downstream direction are presented in figures 10 and 11, respectively. From the analysis of these results we can verify that delays corresponding to the upstream direction vary between approximately 30 and 120 milliseconds, being much smaller that the corresponding values for the downstream direction that vary between 20 and 2300 milliseconds. Packet losses are null in the upstream direction but have non zero values in the downstream direction. As expected, there is a direct relationship between packet delays and losses: higher packet delay values also correspond to higher packet loss values. In the performed tests, downstream traffic was much higher than upstream traffic, which is a typical result for these kind of scenarios. In the downstream direction, the highest delay and loss values were observed in the night and afternoon (between 2PM and 6PM) periods. These values can be attributed to the use of file sharing applications. In the night period, the utilization level of these applications is even higher, mainly from the students' residences. In the afternoon period, the utilization of these applications is mainly performed from the library building, which is also covered by the wireless network.

6 Conclusions

Traffic monitoring through active measurements is having an increasing relevance in the IP networks management context, since it enables to directly monitor quality of service parameters, like for example average packet delay and packet loss ratio. The IETF IPPM group has recently proposed a protocol for conducting active traffic measurements in a single direction, the OWAMP (One-Way Active Measurement Protocol).

This paper presented a solution (based on the OWAMP protocol) for performing active measurements in a heterogeneous network, including its implementation, validation and some examples that allow a further exploration of the OWAMP protocol. The proposed system was developed in Java language, mainly due to its portability. Several compliance tests with the only known implementation (from the Internet2 group) were successfully conducted. The system was evaluated through a set of performed tests, conducted both in a laboratorial environment and in a real operational network. The obtained results show that the implemented system is a very useful active measurement tool that can be used for characterizing quality of service in IP networks.

Acknowledgments. This research was supported by Fundação para a Ciência e a Tecnologia, project POSI/42069/CPS/2001, and European Commission, Network of Excellence EuroNGI (Design and Engineering of the Next Generation Internet).

References

1. A.Pasztor, D.Veitch: High precision active probing for internet measurement. In: Proceedings of INET'2001. (2001)
2. Corral, J., Texier, G., Toutain, L.: End-to-end active measurement architecture in ip networks (saturne). In: Proceedings of Passive and Active Measurement Workshop PAM'03. (2003)
3. Grossglauser, M., Krishnamurthy, B.: Looking for science in the art of network measurement. In: Proceedings of IWDC Workshop. (2001)
4. NeTraMet home page: (http://www.auckland.ac.nz/net/netramet/)
5. White Paper - NetFlow Services and Applications: (http://www.cisco.com/warp/public/cc/pd /iosw/ioft/neflct/tech/napps_wp.htm)
6. Almes, G., Kalidindi, S., Zekauskas, M.: RFC 2679: A one-way delay metric for ippm (1999)
7. Almes, G., Kalidindi, S., Zekauskas, M.: RFC 2680: A one-way packet loss metric for ippm (1999)
8. Shalunov, S., Teitelbaum, B., Karp, A., andMatthew J. Zekauskas, J.W.B.: A one-way active measurement protocol (owamp), internet draft (2004)
9. Internet2 End-to-End Performance Initiative: (http://e2epi.internet2.edu)
10. Boyd, E.L., Boote, J.W., Shalunov, S., Zekauskas, M.J.: The internet2 e2e pipes project: An interoperable federation of measurement domains for performance debugging (2004)

Primary/Secondary Path Generation Problem: Reformulation, Solutions and Comparisons

Quanshi Xia and Helmut Simonis

IC-Parc, Imperial College London, London SW7 2AZ, UK
{q.xia, h.simonis}@imperial.ac.uk

Abstract. This paper considers the primary and secondary path generation problem in traffic engineering. We first present a standard MILP model. Since its size and integrality gap are very large, we then apply a Benders decomposition to isolate the failure case capacity constraints, related linearisation variables and linearisation constraints.

The disaggregated Benders cuts are generated, which is actually the set of *violated* failure case capacity constraints with their linearisation variables and the required linearisation constraints. This corresponds to adding the failure case capacity constraints, their linearisation variables and linearisation constraints only as they are needed.

Some results on generated test cases for different network topologies are given. In comparison with the standard MILP formulation, we reduce execution times on average by a factor of 1000 using the Benders decomposition. We also compare with a scheme of accepting demands one-by-one, which can handle more large-scale problems at the cost of loosing optimality.

1 Introduction

Using the emerging multiprotocol label switching technology, some Internet service providers are switching to a traffic engineered operation from best-effort traffic, where paths between nodes in the network are specified explicitly. This can help to provide some guaranteed quality of service (QoS), such as the bandwidth provision or delay and jitter bounds [1].

For services like voice over IP and video conferencing, the customer needs a reliable service without disruptions even if some elements in the network fail. Fast re-route tunnels can provide an emergency alternative in a very short period. These tunnels can be precomputed and set-up on the network, *e.g.* **Cisco's Tunnel Builder Pro**. However, as a solution for longer lasting outages a secondary path needs to be provided as well [2]. This secondary tunnel is activated when the failure of the primary tunnel is signalled to the head router. As the secondary tunnel should not be affected by the element failure that might affect the primary tunnel, we must require that it is disjoint from the primary path. Most often we only consider link element failures, and the secondary path is not allowed to use the same links as the primary path.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 611–619, 2005.

Only reserving the full bandwidth for the primary path and setting up zero bandwidth tunnels for the secondary paths will minimize the required network resources, but cannot guarantee QoS when an element fails. On the other hand, reserving the full bandwidth for both primary and secondary paths at the same time will results in an inefficient usage of the network. As primary and secondary tunnels for the same demand are never used concurrently, therefore, in the normal network operation, only the primary tunnels are used and require bandwidth. The sum of all bandwidth demands routed over a given link should be within its capacity. In a failure case, most of the primary tunnels will be still active, only those routed over failed elements will be replaced by their secondary tunnels. The total bandwidth required by all primary tunnels still in use and all active secondary tunnels should not exceed the edge capacity. To express all capacity constraints for all failure cases leads to a large number of variables and constraints. We use Benders decomposition to avoid stating all these constraints *a priori*. In our master problem we find primary and secondary paths, and then check in our subproblem if any capacity constraints are violated. We add the violated capacity constraints as Benders cuts to the master problem until a solution is found which does not violate the capacity constraints. By construction, this is the optimal solution to the global problem.

2 Primary/Secondary Path Generation Problem

Recently, Papadakos provided a model [3] to find primary and secondary paths, accounting for the bandwidth use in normal operation and in all failure cases separately. In this section we summarize the results of that paper.

The network is modelled as a set of nodes $\mathbf{N}$ and a set of edges $\mathbf{E}$. Each edge $e(k,l) \in \mathbf{E}$ directly connects node k to node l, associated with a capacity c_e. For each node $n \in \mathbf{N}$ there is a set of edges $\mathbf{I}(n)$ entering n and a set of edges $\mathbf{O}(n)$ leaving n. The set of demands to be routed over the network is called $\mathbf{D}$. Each demand $d \in \mathbf{D}$ has an origin s_d, a destination t_d and a maximum requested bandwidth B_d. For each demand, if accepted, a primary and a secondary path must be found, so that in the normal operation and in all failure cases the bandwidth required by primary and secondary paths active in this failure case do not exceed the available edge capacity.

We introduce three sets of (0/1) variables and four sets of constraints as:

Acceptance variable Z_d *states whether demand d is accepted or not.*
Primary path variable X_{de} *states whether edge e is used for the primary path of demand d.*
Secondary path variable W_{de} *states whether edge e is used for the secondary path of demand d.*
Primary path constraint *states that for an accepted demand there must be a continuous primary path from the source to the destination.*

$$\forall d \in \mathbf{D}, \forall n \in \mathbf{N}: \quad \sum_{e \in \mathbf{O}(n)} X_{de} - \sum_{e \in \mathbf{I}(n)} X_{de} = \begin{cases} -Z_d & n = t_d \\ Z_d & n = s_d \\ 0 & otherwise \end{cases}$$

Secondary path constraint *states for an accepted demand there also must be a continuous secondary path from the source to the destination.*

$$\forall d \in \mathbf{D}, \forall n \in \mathbf{N}: \quad \sum_{e \in \mathbf{O}(n)} W_{de} - \sum_{e \in \mathbf{I}(n)} W_{de} = \begin{cases} -Z_d & n = t_d \\ Z_d & n = s_d \\ 0 & \textit{otherwise} \end{cases}$$

Edge disjoint path constraint *states that primary and secondary paths are not allowed to use the same edges.*

$$\forall d \in \mathbf{D}, \forall e \in \mathbf{E}: \quad X_{de} + W_{de} \leq 1$$

Capacity constraint *states that in the normal operation and in all failure cases the bandwidth required by primary and secondary paths active in this failure case do not exceed the edge capacity.*

$$\begin{cases} \forall e \in \mathbf{E}: \quad \sum_{d \in \mathbf{D}} B_d X_{de} \leq c_e \\ \forall e \in \mathbf{E}, e' \in \mathbf{E}/\mathbf{e}: \quad \sum_{d \in \mathbf{D}} B_d (X_{de} - X_{de'} X_{de} + X_{de'} W_{de}) \leq c_e \end{cases} \tag{1}$$

This capacity constraint limited the bandwidth use for edge e both by the use for all primary paths routed through it and by the largest use in any of the failure cases considered. Assume a failure in edge e'. Then the traffic through e is the sum of all primary paths passing through e which are not passing through e' as well, and the sum of all secondary paths through e for demands where the primary path is routed through e'.

Finally, the MIP formulation is presented as:

$$\max_{\{Z_d, X_{de}, W_{de}\}} \quad \sum_{d \in \mathbf{D}} B_d Z_d$$

$$st. \begin{cases} \forall d \in \mathbf{D}, \forall n \in \mathbf{N}: \sum_{e \in \mathbf{O}(n)} X_{de} - \sum_{e \in \mathbf{I}(n)} X_{de} = \begin{cases} -Z_d & n = t_d \\ Z_d & n = s_d \\ 0 & \text{otherwise} \end{cases} \\ \forall e \in \mathbf{E}: \quad \sum_{d \in \mathbf{D}} B_d X_{de} \leq c_e \\ \forall d \in \mathbf{D}, \forall n \in \mathbf{N}: \sum_{e \in \mathbf{O}(n)} W_{de} - \sum_{e \in \mathbf{I}(n)} W_{de} = \begin{cases} -Z_d & n = t_d \\ Z_d & n = s_d \\ 0 & \text{otherwise} \end{cases} \\ \forall e \in \mathbf{E}, \forall e' \in \mathbf{E}/\mathbf{e}: \sum_{d \in \mathbf{D}} B_d (X_{de} - X_{de'} X_{de} + X_{de'} W_{de}) \leq c_e \\ \forall d \in \mathbf{D}, \forall e \in \mathbf{E}: \quad X_{de} + W_{de} \leq 1 \end{cases} \tag{2}$$

3 Problem Reformulation

The MIP model (2) was using a number of non-linear constraints. Differing from [3], we propose a new linearisation. For this, we introduce new 0/1 **linearisation variables** $Y_{dee'} = (1 - X_{de'})X_{de} + X_{de'} W_{de}$ and a set of **linearisation constraints**:

$$\begin{cases} X_{de} - X_{de'} \leq Y_{dee'} \leq X_{de} + X_{de'} \\ W_{de} + X_{de'} - 1 \leq Y_{dee'} \leq W_{de} - X_{de'} + 1 \end{cases} \tag{3}$$

This results in a new MILP formulation:

$$\max_{\{Z_d, X_{de}, W_{de}, Y_{dee'}\}} \sum_{d \in \mathbf{D}} B_d Z_d$$
$$st. \begin{cases} \forall d \in \mathbf{D}, \forall n \in \mathbf{N}: \sum_{e \in \mathbf{O}(n)} X_{de} - \sum_{e \in \mathbf{I}(n)} X_{de} = \begin{cases} -Z_d & n = t_d \\ Z_d & n = s_d \\ 0 & \text{otherwise} \end{cases} \\ \forall e \in \mathbf{E}: \quad \sum_{d \in \mathbf{D}} B_d X_{de} \leq c_e \\ \forall d \in \mathbf{D}, \forall n \in \mathbf{N}: \sum_{e \in \mathbf{O}(n)} W_{de} - \sum_{e \in \mathbf{I}(n)} W_{de} = \begin{cases} -Z_d & n = t_d \\ Z_d & n = s_d \\ 0 & \text{otherwise} \end{cases} \\ \forall e \in \mathbf{E}, \forall e' \in \mathbf{E}/e: \quad \sum_{d \in \mathbf{D}} B_d Y_{dee'} \leq c_e \\ \forall e \in \mathbf{E}, \forall e' \in \mathbf{E}/e, d \in \mathbf{D}: \begin{cases} X_{de} - X_{de'} \leq Y_{dee'} \leq X_{de} + X_{de'} \\ W_{de} + X_{de'} - 1 \leq Y_{dee'} \leq W_{de} - X_{de'} + 1 \end{cases} \\ \forall d \in \mathbf{D}, \forall e \in \mathbf{E}: \quad X_{de} + W_{de} \leq 1 \end{cases} \tag{4}$$

which can be solved by MIP solvers. However, this MILP formulation has a huge number of (1) failure case capacity constraints; (2) linearisation variables; (3) linearisation constraints. The *integrality gap* is very big, limiting the scalability of the model. Although we can solve smaller problem instances, both memory usage and execution time grow very quickly for larger ones.

4 Solution by Benders Decomposition

Benders decomposition partitions an optimisation problem into two smaller problems, the *master problem* and the *subproblem*. The Benders algorithm iteratively solves the master problem and the subproblem. In every iteration, the subproblem solution provides the *Benders cut*, added to the master problem, narrowing down the search space and leading to optimality [5].

Applying the Benders decomposition to the problem (4), we choose the failure case capacity constraints as the subproblem (SP), and the primary/secondary path generation as the master problem (MP).

SP – Failure Case Capacity Constraint. Suppose the primary/secondary path solution $\{\tilde{x}_{de}^{(k)}, \tilde{w}_{de}^{(k)}\}$. The capacity constraints for the failure cases are checked by the subproblem:

$$\min_{\{S_{ee'} \geq 0\}} \quad \phi_e = \sum_{e' \in \mathbf{E}/e} S_{ee'}$$
$$st. \begin{cases} \forall e' \in \mathbf{E}/e: \quad S_{ee'} \geq \sum_{d \in \mathbf{D}} B_d(\tilde{x}_{de}^{(k)} - \tilde{x}_{de'}^{(k)}\tilde{x}_{de}^{(k)} + \tilde{x}_{de'}^{(k)}\tilde{w}_{de}^{(k)}) - c_e \end{cases} \tag{5}$$

If the objective $\sum_{e\in\mathbf{E}} \phi_e^{(k)} = 0$, then all failure case capacity constraints are satisfied. Therefore the optimal solution is obtained as $\{\tilde{z}_d^{(k)}, \tilde{x}_{de}^{(k)}, \tilde{w}_{de}^{(k)}\}$. However, if $\phi_e^{(k)} > 0$ (not all failure case capacity constraints are satisfied), then new Benders cuts must be generated and added to the master problem to cut off the solution $\{\tilde{x}_{de}^{(k)}, \tilde{w}_{de}^{(k)}\}$.

In order to generate the Benders cut, the dual of subproblem (5)

$$\max_{\{0\le\gamma_{ee'}\le 1\}} \sum_{e'\in\mathbf{E}/\mathbf{e}} [\sum_{d\in\mathbf{D}} B_d(\tilde{x}_{de}^{(k)} - \tilde{x}_{de'}^{(k)}\tilde{x}_{de}^{(k)} + \tilde{x}_{de'}^{(k)}\tilde{w}_{de}^{(k)}) - c_e]\gamma_{ee'} \tag{6}$$

is solved with: $\tilde{\gamma}_{ee'}^{(k)} = \begin{cases} 0 & \sum_{d\in\mathbf{D}} B_d(\tilde{x}_{de}^{(k)} - \tilde{x}_{de'}^{(k)}\tilde{x}_{de}^{(k)} + \tilde{x}_{de'}^{(k)}\tilde{w}_{de}^{(k)}) \le c_e \\ 1 & \sum_{d\in\mathbf{D}} B_d(\tilde{x}_{de}^{(k)} - \tilde{x}_{de'}^{(k)}\tilde{x}_{de}^{(k)} + \tilde{x}_{de'}^{(k)}\tilde{w}_{de}^{(k)}) > c_e \end{cases}$

The *disaggregated* Benders cut, which will make the solution $\{\tilde{x}_{de}^{(k)}, \tilde{w}_{de}^{(k)}\}$ infeasible, is generated as

$$\begin{cases} \forall (e,e') \in \mathbf{C}^{(k)}: & \sum_{d\in\mathbf{D}} B_d(X_{de} - X_{de'}X_{de} + X_{de'}W_{de}) \le c_e \\ \mathbf{C}^{(k)} = \{(e,e') : \sum_{d\in\mathbf{D}} B_d(\tilde{x}_{de}^{(k)} - \tilde{x}_{de'}^{(k)}\tilde{x}_{de}^{(k)} + \tilde{x}_{de'}^{(k)}\tilde{w}_{de}^{(k)}) > c_e\} \end{cases} \tag{7}$$

By use of the related linearisation variables and the linearisation constraints, the Benders cuts (7) can be linearised as

$$\begin{cases} \forall (e,e') \in \mathbf{C}^{(k)}: & \sum_{d\in\mathbf{D}} B_d Y_{dee'} \le c_e \\ \forall (e,e') \in \mathbf{C}^{(k)}, \forall d \in \mathbf{D}: & \begin{cases} X_{de} - X_{de'} \le Y_{dee'} \le X_{de} + X_{de'} \\ W_{de} + X_{de'} - 1 \le Y_{dee'} \le W_{de} - X_{de'} + 1 \end{cases} \end{cases} \tag{8}$$

We then add these Benders cuts to the master problem to narrow down its feasible solution space.

MP – Generating Primary/Secondary Paths. In the next iteration we collect all Benders cuts generated by all subproblems, and construct a new master problem

$$\max_{\{Z_d, X_{de}, W_{de}, Y_{dee'}\}} \sum_{d\in\mathbf{D}} B_d Z_d$$

$$st. \begin{cases} \forall d \in \mathbf{D}, \forall n \in \mathbf{N}: \sum_{e\in\mathbf{O}(n)} X_{de} - \sum_{e\in\mathbf{I}(n)} X_{de} = \begin{cases} -Z_d & n = t_d \\ Z_d & n = s_d \\ 0 & \text{otherwise} \end{cases} \\ \forall e \in \mathbf{E}: \quad \sum_{d\in\mathbf{D}} B_d X_{de} \le c_e \\ \forall d \in \mathbf{D}, \forall n \in \mathbf{N}: \sum_{e\in\mathbf{O}(n)} W_{de} - \sum_{e\in\mathbf{I}(n)} W_{de} = \begin{cases} -Z_d & n = t_d \\ Z_d & n = s_d \\ 0 & \text{otherwise} \end{cases} \\ \forall (e,e') \in \mathbf{E}^{(k)} = \mathbf{E}^{(k-1)} \cup \mathbf{C}^{(k)}: \sum_{d\in\mathbf{D}} B_d Y_{dee'} \le c_e \\ \forall (e,e') \in \mathbf{E}^{(k)}, \forall d \in \mathbf{D}: \begin{cases} X_{de} - X_{de'} \le Y_{dee'} \le X_{de} + X_{de'} \\ W_{de} + X_{de'} - 1 \le Y_{dee'} \le W_{de} - X_{de'} + 1 \end{cases} \\ \forall d \in \mathbf{D}, \forall e \in \mathbf{E}: \quad X_{de} + W_{de} \le 1 \end{cases} \tag{9}$$

We resolve the master problem to find the new primary/secondary path solution $\{\tilde{x}_{de}^{(k+1)}, \tilde{w}_{de}^{(k+1)}\}$, and then check the satisfaction of the failure case capacity constraints in our subproblem. Initially, the master problem is solved without any failure case capacity constraints, *i.e.* $\mathbf{E}^{(0)} = \emptyset$.

It is worth drawing attention to the difference between the MILP formulation (4) and the master problem (9): only a few failure case capacity constraints and their related linearisation variables and linearisation constraints involved in master problem (9). This makes the master problem (9) easier to solve, at least as long as we do not have to add too many cuts in the process.

Adding Constraints and Variables as Needed. Instead of stating all failure case capacity constraints *at the very beginning*, we start to solve the primary/secondary path generation problem without the failure case capacity constraints, find a primary/secondary path solution and check which failure case capacity constraints are violated. If no violations are detected, then the solution is optimal. If violations are found, then we add *violated* failure case capacity constraints and related linearisation variables and linearisation constraints, and resolve the master problem. They act as a cutting plane, *i.e.* the previous solution becomes infeasible and a new solution must be found, which does not have the defect of the previous solution. In the worst case, we need to add all failure cases capacity constraints before finding a solution.

This strategy is similar to one which used when solving the Travelling Salesman Problem (TSP) with MILP to efficiently handle the exponential number of sub-tour elimination constraints. However, the sub-tour elimination constraints are linear [6], while the failure case capacity constraints are nonlinear, and need a large number of linearisation variables and linearisation constraints to make them linear. This means that in our case, the strategy of *adding constraints and variables as needed* is even more powerful.

5 Implementation, Results and Comparison

Our current implementation consists of a single MILP master problem (9) and, in principle, LP subproblems for each failure case capacity constraint. The efficient handling of the MILP master problem and addition of new rows to the master problem is supported by the hybrid MILP/CP software platform ECL^{i}PS^e [7], which provides interfaces to CPLEX [4] for solving MILP problems.

Three different networks have been used to evaluate our algorithm, which have between 10 and 38 nodes and 30 to 116 directed edges. We choose demand sets from 10 to 40 demands, and divided the bandwidth required into 2 classes (small and large). The small bandwidth is randomly chosen from 100 to 500 in increments of 50, the large bandwidth from 100 to 2000 in increments of 200.

For the small bandwidth group, Benders decomposition can solve all instances very easily requiring only 1-3 iterations. Table 1 shows the objective as the bandwidth size of the accepted demands, MILP solution time (CPU) and size in number of variables (Vars) and constraints (Cstrs), and Benders decomposition

Table 1. Small Bandwidth Demand

Network Nodes	Network Edges	Demands Number	Obj	MILP CPU	MILP Vars	MILP Cstrs	BD CPU	BD IT
10	30	10	2650	2.79	9311	36401	0.58	2
		20	4700	144.96	18621	71901	1.38	2
		30	6950	OM	27931	107401	340.51	2
20	66	10	2650	536.26	44231	177417	1.28	1
		20	4550	5564.54	88461	350477	5.40	2
		30	6800	12989.33	132691	523537	10.88	2
		40	8500	OM	176921	696597	25.08	3
38	116	10	2150	33822.51	132497	536785	4.17	1
		20	3050	OM	266379	1065635	8.88	1
		30	5200	OM	398875	1588963	27.34	2

Table 2. Large Bandwidth Demand

Network Nodes	Network Edges	Demands Number	Obj	MILP CPU	MILP Vars	MILP Cstrs	BD CPU	BD IT
10	30	10	6000	4594.88	8093	31676	5.01	2
		20	8400	TO	16881	65151	419.92	4
		30		OM	25147	96601	TO	
20	66	10	8600	4164.53	39751	159742	10.17	6
		20	13000	OM	82061	325227	3381.68	6
		30		OM	122451	483137	TO	
		40		OM	160921	633472	TO	
38	116	10	7000	11516.08	110041	447409	12.13	3
		20	8400	OM	232583	931235	56.89	6
		30	12600	OM	346095	1379435	6650.73	11

Table 3. Large Bandwidth Demand

Network Nodes	Network Edges	Demands Number	BD Nr	BD Obj	BD CPU	fXfW Nr	fXfW Obj	fXfW CPU	fXvW Nr	fXvW Obj	fXvW CPU
10	30	10	8	6000	5.01	8	5600	2.82	8	5600	3.52
		20	14	8400	419.92	15	8100	7.72	15	8100	9.78
		30	TO			20	10600	10.7	21	11900	17.81
20	66	10	8	8600	10.17	10	8600	85.84	10	8600	112.00
		20	18	13000	3381.68	17	11300	329.55	17	11300	252.51
		30	TO			24	16000	427.23	23	15300	357.21
		40	TO			26	17400	464.63	28	19000	414.72
38	116	10	8	7000	12.13	8	7000	775.22	8	7000	588.78
		20	12	8400	56.89	12	8400	1284.23	12	8400	1262.42
		30	18	12600	6650.73	17	10700	1412.44	18	12600	1602.65

solution time (CPU) and iterations (IT). As the demand bandwidth increases, Benders decomposition requires more CPU time and iterations to find the optimal solution, results are shown in Table 2. CPU are given for a Linux based workstation with a 2GHz Pentium4, OM indicates out of memory on a 1GB machine, and TO indicates a time out after 72000 seconds.

In our 20 test instances, the Benders decomposition solves all but 3 problem instances within the time limit, requiring between 1 and 11 iterations. Of all 20 test instances, the MILP cannot solve 11 problem instances, and for the 9 solved instances, MILP is not efficient and takes on average 1000 times longer than the Benders decomposition.

Place the Demands One by One. As an alternative scheme, we can try to place demands one by one. Usually, we will not obtain the optimal solution to the global problem, but we can handle much larger problem sizes. We consider two variants of this scheme.

Scheme **fXfW** - Suppose that at step k, $\mathbf{D}^k$ is the set of demands which previously have been tried to be placed in the network ($\tilde{z}_d$ accepted or rejected), therefore we know their primary/secondary path solution($\tilde{x}_{de}, \tilde{w}_{de}$). And at step k, we try to place a set of new demands, $\mathbf{D}^{(k)}$, keeping already placed ones fixed.

Scheme **fXvW** - We can also allow to reroute all secondary tunnels, keeping only the primary paths of previously accepted demands fixed. Therefore, when placing new demands at step k, we can treat the secondary paths of previous accepted demands as additional variables.

When we compare this approach with the Benders decomposition results in terms of the number of accepted demands, the bandwidth size and the solution time, which are given in Table 3, we can see that **fXfW** and **fXvW** cannot guarantee the optimality although they can solve all problem instances.

6 Conclusion

When reformulated as a MILP, the primary/secondary path generation problem needs a large number of linearisation variables and linearisation constraints. Because of the big integrality gap, it is quite difficult to solve with a commercial MILP package. In this paper, we have applied a Benders decomposition to solve the primary/secondary path generation problem. Isolating the failure case capacity constraints and their corresponding linearisation variables and linearisation constraints from the other constraints, and adding them only as needed, allows us to solve more realistic problem instances.

References

1. D. Awduche et al, *"Requirements for Traffic Engineering over MPLS"*, Internet Draft - draft-ietf-mpls-traffic-eng.00.txt, (1998)
2. Cisco Inc., *"MPLS Bandwidth Protection"*, White Paper(2001)

3. N.P. Papadakos, *"Optimising Network Utilisation with Backup Routes"*, M.Sc. Thesis, Imperial College London (2002)
4. ILOG Inc., *"ILOG CPLEX 6.5 User's Manual"*, (1999)
5. J.F. Benders, *"Partitioning Procedures for Solving Mixed Variables Programming Problems"*, Numerische Mathematik, Vol.4, 238-252, (1962)
6. G. Pataki, *"Teaching Integer Programming Formulation Using the Travelling Salesman Problem"*, SIAM Review 45(1) (2003)
7. Imperial College London, *"ECL^{i}PS^e 5.6 User's Manual"*, (2003)

A Discrete-Time HOL Priority Queue with Multiple Traffic Classes

Joris Walraevens, Bart Steyaert, Marc Moeneclaey, and Herwig Bruneel

Ghent University, Department TELIN (TW07)
Sint-Pietersnieuwstraat 41, B-9000 Gent, Belgium
jw@telin.UGent.be

Abstract. Priority scheduling for packets is a hot topic, as interactive (voice,video) services are being integrated in existing data networks. In this paper, we consider a discrete-time queueing system with non-preemptive (or Head-Of-the-Line) priority scheduling and a general number of priority classes. Packets of variable length arrive in the queueing system. We derive expressions for the probability generating functions of the packet delays. From these functions, some performance measures (such as moments and approximate tail probabilities) are calculated. We apply the theoretical results to a queue that handles arriving multimedia traffic.

1 Introduction

In recent years, there has been much interest in incorporating multimedia applications in packet-based networks (e.g. IP networks). Different types of traffic need different QoS standards. For real-time interactive applications, it is important that mean delay and delay-jitter are bound, while for non real-time applications, the loss ratio is the restrictive quantity. In order to guarantee acceptable delay boundaries to delay-sensitive traffic (such as voice/video), several scheduling schemes – for switches, routers, ... – have been proposed and analyzed, each with their own specific algorithmic and computational complexity. The (strict) priority scheduling is the most drastic one. With this scheduling, as long as delay-sensitive (or high-priority) packets are present in the queueing system, they will be served first. Delay-insensitive packets can thus only be transmitted when no delay-sensitive traffic is present in the system. Clearly, this is the most rigorous way to meet the QoS constraints of delay-sensitive traffic (and thus the scheduling with the most disadvantageous consequences to the delay-insensitive traffic), but also the easiest to implement. In the related literature, there have been a number of contributions with respect to HOL priority scheduling (see e.g. [1, 2, 3, 4, 5, 6, 7]).

In this paper, we focus on the effect of a non-preemptive or HOL (Head-Of-the-Line) priority scheduling discipline on the performance of a queue with multiple traffic classes. More delay-sensitive traffic is assumed to have non-preemptive priority over traffic with more flexible delay constraints, i.e., when the server

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 620–627, 2005.

becomes idle, a packet of the most delay-sensitive traffic that is available is scheduled for service next. Newly arriving packets cannot interrupt the transmission of a packet that has already commenced, whatever their priority level. The transmission times of the packets are assumed to be generally distributed and class-dependent (which reflects the case where different classes represent different applications). We will demonstrate that an analysis based on probability generating functions (pgf's) is extremely suitable for modeling this type of buffers with a priority scheduling discipline. From these pgf's, we calculate closed-form expressions for some interesting performance measures.

2 Mathematical Model

We consider a discrete-time single-server queueing system with infinite buffer space. Time is assumed to be slotted. There are M types of traffic arriving in the system. We denote the number of packet arrivals of class-j during slot k by $a_{j,k}$ $(j = 1, \ldots, M)$. All types of packet arrivals are assumed to be i.i.d. from slot-to-slot and the number of per-slot arrivals are characterized by the joint pgf

$$A(\mathbf{z}) \triangleq \mathrm{E}\left[\prod_{j=1}^{M} z_j^{a_{j,k}}\right],$$

with $\mathbf{z}$ defined as a vector with elements z_j $(j = 1, \ldots, M)$. We define the marginal pgf's of the arrivals from class-j during a slot by

$$A_j(z) \triangleq \mathrm{E}[z^{a_{j,k}}] = A(\mathbf{z})\big|_{z_j=z, z_i=1, i\neq j}.$$

We furthermore denote the arrival rate of class-j $(j = 1, \ldots, M)$ by $\lambda_j = A_j'(1)$ and the total arrival rate by $\lambda_T = \sum_{j=1}^{M} \lambda_j$. The service times of the class-j packets are assumed to be i.i.d. and are characterized by the probability mass function $s_j(m)$, $m \geq 1$, and pgf $S_j(z) = \sum_{m=1}^{\infty} s_j(m) z^m$, with $j = 1, \ldots, M$. We furthermore denote the mean service time of a class-j packet by $\mu_j = S_j'(1)$ and define the load offered by class-j packets as $\rho_j \triangleq \lambda_j \mu_j$. The total load is given by $\rho_T \triangleq \sum_{j=1}^{M} \rho_j$, and the equilibrium condition requires that $\rho_T < 1$.

The system has one server that provides for the transmission of the packets. Class-i packets are assumed to have non-preemptive priority over class-j packets when $i < j$, and within one class the service discipline is FCFS.

3 System Contents at Service Initiation Epochs

In order to be able to analyze the packet delay, we first analyze the system contents at the beginning of so-called start-slots, i.e., slots at the beginning of which a packet (if available) can enter the server. Note that every slot during which the system is empty is also a start-slot. We denote the system contents $n_{j,l}$

as the number of class-j ($j = 1, \ldots, M$) packets in the buffer at the beginning of the l-th start-slot, including the packet being served (if any). Their joint pgf is denoted by

$$N_l(\mathbf{z}) \triangleq \mathrm{E}\left[\prod_{j=1}^{M} z_j^{n_{j,l}}\right]. \tag{1}$$

The set $\{(n_{1,l}, \ldots, n_{M,l}), l \geq 1\}$ forms a Markov chain, since the arrival process is i.i.d. and the buffer solely contains entire packets at the beginning of start-slots. s_l^* denotes the service time of the packet that enters service at the beginning of start-slot l (which corresponds - by definition - to regular slot k). We then establish the following system equations:

– If $n_{1,l} = \ldots = n_{M,l} = 0$:

$$n_{i,l+1} = a_{i,k} \quad \text{for } i = 1, \ldots, M.$$

– If $n_{1,l} = \ldots = n_{j-1,l} = 0, n_{j,l} > 0$:

$$n_{i,l+1} = n_{i,l} - 1_{i=j} + \sum_{m=0}^{s_l^*-1} a_{i,k+m} \quad \text{for } i = 1, \ldots, M,$$

for $j = 1, \ldots, M$. 1_X is the indicator function of X.

Using the system equations, we can derive a relation between $N_l(\mathbf{z})$ and $N_{l+1}(\mathbf{z})$. We assume that the system is stable (implying that the equilibrium condition $\rho_T < 1$ is satisfied) and as a result $N_l(\mathbf{z})$ and $N_{l+1}(\mathbf{z})$ converge both to a common steady-state value $N(\mathbf{z})$ for $l \to \infty$. By taking the $l \to \infty$ limit of the relation between $N_l(\mathbf{z})$ and $N_{l+1}(\mathbf{z})$, we obtain:

$$\begin{aligned} N(\mathbf{z}) = &\frac{z_1}{z_1 - S_1(A(\mathbf{z}))}\left\{\frac{z_M A(\mathbf{z}) - S_M(A(\mathbf{z}))}{z_M} N(\mathbf{0})\right. \\ &\left. + \sum_{j=2}^{M}\left[\frac{S_j(A(\mathbf{z}))}{z_j} - \frac{S_{j-1}(A(\mathbf{z}))}{z_{j-1}}\right] N(\mathbf{z_j})\right\}. \end{aligned} \tag{2}$$

There are M quantities yet to be determined in the right hand side of equation (2), namely the functions $N(\mathbf{z_j})$ ($j = 2, .., M$) and the constant $N(\mathbf{0})$. First, we will recursively express $N(\mathbf{z_m})$ ($m = 1, .., M$) in terms of the $N(\mathbf{z_j})$ ($j = m + 1, .., M$) and $N(\mathbf{0})$. We define $X_0(\mathbf{z}) \triangleq A(\mathbf{z})$. In the m-th step of our (recursive) calculation, we assume that $X_{m-1}(\mathbf{z_m})$ has already been defined and that the following equation holds:

$$\begin{aligned} N(\mathbf{z_m}) = &\frac{z_m}{z_m - S_m(X_{m-1}(\mathbf{z_m}))}\left\{\frac{z_M X_{m-1}(\mathbf{z_m}) - S_M(X_{m-1}(\mathbf{z_m}))}{z_M} N(\mathbf{0})\right. \\ &\left. + \sum_{j=m+1}^{M}\left[\frac{S_j(X_{m-1}(\mathbf{z_m}))}{z_j} - \frac{S_{j-1}(X_{m-1}(\mathbf{z_m}))}{z_{j-1}}\right] N(\mathbf{z_j})\right\}. \end{aligned} \tag{3}$$

Substituting $m = 1$ in this equation yields equation (2), which is the starting point of our recursive procedure. Applying Rouché's theorem, it can then be proved that for given values of z_j with $|z_j| < 1$ $(j = m+1, .., M)$, the equation $z_m = S_m(X_{m-1}(\mathbf{z_m}))$ has a unique solution in the complex unit circle for z_m, which will be denoted by $Y_m(\mathbf{z_{m+1}})$ in the remainder, and which is implicitly defined by $Y_m(\mathbf{z_{m+1}}) \triangleq S_m(X_{m-1}(\mathbf{z_m}))|_{z_m=Y_m(\mathbf{z_{m+1}})}$. Since any pgf is finite inside the unit circle and since $Y_m(\mathbf{z_{m+1}})$ is a zero of the denominator of the right hand side of (3), $Y_m(\mathbf{z_{m+1}})$ must also be a zero of the numerator. Defining $X_m(\mathbf{z_{m+1}}) \triangleq X_{m-1}(\mathbf{z_m})|_{z_m=Y_m(\mathbf{z_{m+1}})}$ (and $X_0(\mathbf{z_1}) = A(\mathbf{z})$), this leads to expression (3) with m substituted by $m+1$, which means that the $m+1$-th step of the algorithm can be applied next. After $M-1$ iterations we finally find:

$$N(\mathbf{z_M}) = N(\mathbf{0}) \frac{z_M X_{M-1}(\mathbf{z_M}) - S_M(X_{M-1}(\mathbf{z_M}))}{z_M - S_M(X_{M-1}(\mathbf{z_M}))}. \tag{4}$$

Next, we can calculate the functions $N(\mathbf{z_m})$ $(m = 1, \ldots, M-1)$ as a function of $N(\mathbf{0})$. We therefore iteratively substitute the (in that step already) found expressions of $N(\mathbf{z_j})$ $(j = m+1, \ldots, N)$ in equation (3). Equaling m to 1 finally gives $N(\mathbf{z})$ as a function of $N(\mathbf{0})$. The expression for general M is too elaborate though, but we have outlined the principle by which this M-th dimensional function can be calculated. The last remaining unknown is $N(\mathbf{0})$. This constant can be calculated by applying the normalization condition $N(\mathbf{1}) = 1$, with $\mathbf{1}$ a vector of size M with all elements equal to 1. This concludes the procedure to calculate $N(\mathbf{z})$, which is used in the analysis of the packet delays in the next section.

4 Packet Delays

The delay of a packet is defined as the number of slots between the end of the packet's arrival slot and the end of its departure slot. We denote the delay of a tagged class-j packet by d_j and its pgf by $D_j(z)$ $(j = 1, \ldots, M)$. We furthermore denote the arrival slot of the tagged packet by slot k. If slot k is a start-slot, it is assumed to be start-slot l. If slot k is not a start-slot on the other hand, the last start-slot preceding slot k is assumed to be start-slot l. In this section, we show how an expression for $D_j(z)$ - for general j - is derived.

Let us first refer to the packets in the system at the end of slot k, but that have to be served before the tagged packet as the "primary packets". So, basically, the tagged class-j packet enters the server, when all primary packets and all packets with higher priority that arrived after slot k (i.e., while the tagged packet is waiting in the queue) are transmitted. In order to analyze the delay of the tagged class-j packet, the number of packets that are served between the arrival slot of the tagged class-j packet and its departure slot is important (and more specifically the time necessary to transmit them), not the precise order in which they are served. Therefore, in order to facilitate the analysis, we will consider an equivalent virtual system with an altered service discipline. From slot k on, we aggregate the $j-1$ highest priority classes in one class and serve the packets in this aggregated class in a LCFS way (those in the queue at the end of slot k and newly arriving ones). So, a primary packet can enter the server, when the system becomes free (for the first time) of packets of this aggregated class that arrived during and after the service time of the primary packet that preceded it according to the new service discipline. Let $v_{i,m}^{(n)}$ $(i=1,\ldots,j)$ denote the length of the time period during which the server is occupied by the m-th class-i packet that arrives during slot n and its "successors" of the aggregated class, i.e., the time period starting at the beginning of the service of that packet and terminating when the system becomes free (for the first time) of packets of the $j-1$ highest priority classes which arrived during and after its service time. The $v_{i,m}^{(n)}$'s $(i=1,\ldots,j)$ are called sub-busy periods, initiated by the m-th class-i packet that arrived during slot n. Notice that the $v_{i,m}^{(n)}$ depend on the class we are analyzing (i.e. class-j), but to alleviate the notation this dependency is taken into account implicitly. It is clear that the lengths of consecutive sub-busy periods initiated by class-i packets are i.i.d. and thus have the same pgf $V_i(z)$ (which implicitly depends on j). This pgf is given by

$$V_i(z) = S_i(zA(V_1(z),\ldots,V_{j-1}(z),1,\ldots,1)), \tag{5}$$

with $i=1,\ldots,j$; $j=1,\ldots,M$. This can be understood as follows: when the m-th class-i packet that arrived during slot n enters service, its sub-busy period, $v_{i,m}^{(n)}$, consists of two parts: the service time of that packet itself, and the service times of the packets of higher priority than the tagged class-j packet (the aggregated class) that arrive during its service time and of their successors of the aggregated class. This leads to equation (5).

Finally, d_j can be expressed in terms of the $n_{i,l}$, $i=1,\ldots,M$ defined in the previous section. $D_j(z)$ is then calculated as a function of $N(\mathbf{z})$ by z-transforming this expression for d_j. The function $N(\mathbf{z})$ was already calculated in section 3 and as a result $D_j(z)$ can be found. We refer to [7] and [8] for more details on similar queueing analyses. $D_j(z)$ is found to be given by (after some extensive mathematical manipulations)

$$\begin{aligned} D_j(z) =& \frac{1-\sum_{i=1}^{j}\rho_i}{\lambda_j}\,\frac{S_j(z)(zB_{j-1}(z)-1)}{zB_{j-1}(z)-B_j(z)}\,\frac{B_j(z)-B_{j-1}(z)}{V_j(z)-1} \\ &\times\left(1-\frac{\sum_{i=j+1}^{M}\rho_i}{1-\sum_{i=1}^{j}\rho_i}+\frac{1}{1-\sum_{i=1}^{j}\rho_i}\sum_{i=j+1}^{M}\rho_i\frac{V_i(z)-1}{\mu_i(zB_{j-1}(z)-1)}\right), \end{aligned} \tag{6}$$

with expression (5) of the $V_i(z)$ expanded to $i=j+1,\ldots,M$ and with $B_i(z)\triangleq A(V_1(z),\ldots,V_i(z),1,\ldots,1)$ $(i=1,\ldots,j)$. Note that this expression is also correct for $D_1(z)$, the pgf of the highest priority class.

5 Performance Measures

The functions $V_i(z)$, defined in the previous section, can be explicitly found only in case of some specific arrival and service processes. Their derivatives for $z = 1$, necessary to calculate the moments of the packet delay, on the contrary, can be calculated in closed-form. So means, variances and higher moments of the packet delays of all classes can be calculated straightforwardly by taking the appropriate derivatives of expression (6) and substituting z by 1.

Furthermore, the tail probabilities of the packet delays can also be approximately calculated from the pgf's calculated in the previous section. These tail distributions are often used to impose statistical bounds on the guaranteed QoS for both classes. In order to determine the asymptotic behavior of the tail distribution, the dominant singularity of the respective pgf is important. The tail behavior of the delay of class-j packets is a bit more involved than in usual queueing analyses, since it is not a priori clear what the dominant singularity of $D_j(z)$ is. This is due to the occurrence of the functions $V_i(z)$, $i = 1, \ldots, j-1$ in (6), which are only implicitly defined. In [6] it is proved that these implicitly defined functions have a branch-point singularity z_B where their first derivatives become infinite but the functions themselves remain finite. z_B is then also a branch point of $D_j(z)$. A second potential singularity of $D_j(z)$ is given by the (dominant) zero z_P of $zB_{j-1}(z) - B_j(z)$ on the real axis (see expression (6)).

The tail behavior of the packet delay of class-j packets is thus characterized by the singularities z_P or z_B, depending on which one is dominant (i.e., which one has the smallest modulus). Three types of tail behavior may occur, namely when z_P is dominant, $z_P = z_B$ and z_B is dominant. In those three cases, the tail probabilities of the class-j packet delay are given by (see [6] for more details)

$$\mathrm{Prob}[d_j = n] \approx \begin{cases} K_1 z_P^{-n+1} & \text{if } z_P \text{ dominant} \\ \dfrac{K_2 n^{-1/2} z_B^{-n}}{\sqrt{z_B \pi}} & \text{if } z_P = z_B \text{ dominant} \\ \dfrac{K_3 n^{-3/2} z_B^{-n}}{2\sqrt{\pi / z_B}} & \text{if } z_B \text{ dominant.} \end{cases} \tag{7}$$

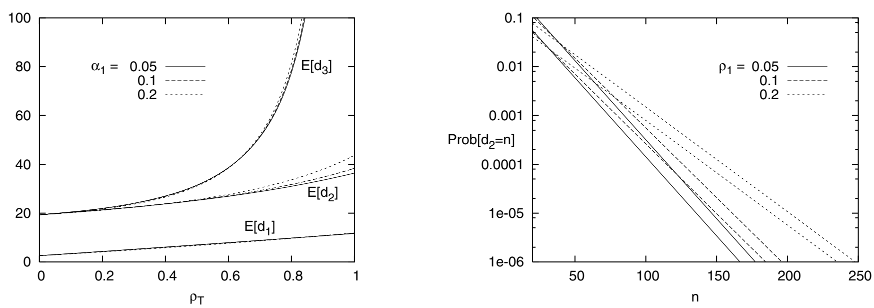
The constants K_i $(i = 1, 2, 3)$ can be found by investigating the pgf $D_j(z)$ in the neighborhood of its singularity. The first expression of (7) shows a typical geometric tail behavior, the third expression shows a typical non-geometric tail behavior and the second expression gives a transition between both.

6 Numerical Examples

In this section, we present some numerical examples. We assume traffic of three traffic classes being handled by a queue, e.g. a class consisting of voice traffic, one of Video-on-Demand (VoD) traffic and a third one of data traffic. We call these class-1, class-2 and class-3 respectively in the remainder. Evidently, an interactive voice application will have the most stringent delay requirements while data (file transfer) will have the loosest delay bounds, with VoD somewhere in between. This is reflected by the priority level that has been assigned to each of the three traffic classes. The numbers of per-slot arrivals of class-j are distributed according to a Poisson process with arrival rate λ_j. We furthermore assume deterministic service times for class-j equal to μ_j number of slots. The arriving packets are transmitted via an output link. We assume that this link has a transmission rate of 620 Mb/s. The video and data packets all carry 1500 bytes corresponding to the length of an Ethernet packet. Due to the relatively low bitrate of a voice codec (8-64 kb/s), the filling time of a voice packet can become significant if the packet length is too high; as a result voice packets are usually kept small and are chosen to be equal to 200 bytes in this section. The slot-length Δ then equals the amount of time to transmit 100 bytes, or $\Delta = 1.29\mu s$. Define α_j $(j = 1, 2, 3)$ as the fraction of load of class-j in the total traffic mix, i.e., $\alpha_j = \rho_j/\rho_T$.

In Figure 1a., the means of the packet delay of the class-1, class-2 and class-3 packets are shown as functions of the total load ρ_T for $\alpha_1 = 0.05, 0.1$ and 0.2 respectively and for $\alpha_2 = 0.4$. One can observe the influence of assigning priorities: mean delay of voice packets is kept as low as possible. Since the voice packets constitute a limited fraction of the traffic stream, the mean delay of the video packets is also kept relatively low. However, one can see that, especially for $\alpha_1 = 0.2$ and for high loads, the influence of the voice packets on the mean delay of the video packets is not negligible. The price to pay for limiting the delay of voice and video packets is a larger mean delay of the data packets (as expected). This figure also shows that the mean delays of *all* classes suffers as the share of the high-priority traffic in the overall load increases.

Figure 1b. shows the tail probabilities of the packet delay of the class-2 (video) packets with $\rho_1 = 0.05, 0.1$ and 0.2 respectively and $\rho_2 = 0.4$. The value of d_2 of this figure is expressed in μs. For each value of ρ_1 we have plotted two curves. The lower curves depict the tail probabilities when no data traffic arrives (i.e, $\rho_3 = 0$). The upper curves show the tail probabilities when the bandwidth that is not used by the voice or video traffic is consumed entirely by data packets $(\rho_3 = 1 - \rho_1 - \rho_2)$. We see that data packets have a non-negligible influence



a. *Mean packet delay (in μs)* b. *Tail behavior of the class-2 delay (in μs)*

Fig. 1. Numerical examples

on the delay characteristics of video traffic (since the priority scheduling is non-preemptive and thus video packets that arrive during the transmission of a data packet have to wait until that packet is fully transmitted), although this influence remains limited. The impact of the voice packets on the delay characteristics of the video packets is much larger. This was to be expected because of the priority given to voice packets over video packets.

7 Conclusions

In this paper, we analyzed the packet delays of all classes in a queueing system with a non-preemptive (HOL) priority scheduling discipline and with a general number of priority classes. A generating-functions-approach was adopted, which led to closed-form expressions of the moments and accurate approximations of the tail probabilities of the packet delays of all the classes, that are easy to evaluate. The service times are class-based and generally distributed. Therefore, the results could be used to evaluate the system performance in packet-based networks, that support multiple applications to which different priorities are assigned. An example is touched upon wherein voice, video and data streams are multiplexed.

References

1. Rubin, I., Tsai, Z.: Message delay analysis of multiclass priority TDMA, FDMA, and discrete-time queueing systems. IEEE Transactions on Information Theory 35 (1989) 637–647
2. Takahashi, Y., Hashida, O.: Delay analysis of discrete-time priority queue with structured inputs. Queueing Systems 8 (1991) 149–164
3. Takine, T., Matsumoto, Y., Suda, T., Hasegawa, T.: Mean waiting times in nonpreemptive priority queues with Markovian arrival and i.i.d. service processes. Performance Evaluation 20 (1994) 131–149
4. Sugahara, A., Takine, T., Takahashi, Y., Hasegawa, T.: Analysis of a nonpreemptive priority queue with SPP arrivals of high class. Performance Evaluation 21 (1995) 215–238
5. Takine, T.: A nonpreemptive priority MAP/G/1 queue with two classes of customers. Journal of Operations Research Society of Japan 39 (1996) 266–290
6. Laevens, K., Bruneel, H.: Discrete-time multiserver queues with priorities. Performance Evaluation 33 (1998) 249–275
7. Walraevens, J., Steyaert, B., Bruneel, H.: Delay characteristics in discrete-time GIG-1 queues with non-preemptive priority queueing discipline. Performance Evaluation 50 (2002) 53–75
8. Bruneel, H., Kim, B.: Discrete-time models for communication systems including ATM. Kluwer Academic Publisher, Boston (1993)

SCTP over High Speed Wide Area Networks

Dhinaharan Nagamalai, Seoung-Hyeon Lee, Won-Goo Lee, and Jae-Kwang Lee

Department of Computer Engineering, Hannam University, 306-791, Daejeon, South Korea
dhinaharann@yahoo.com
{shlee, wglee, jklee}@netwk.hannam.ac.kr

Abstract. The Stream Control Transmission Protocol (SCTP) is a reliable transport protocol to tackle the limitations of TCP and UDP. SCTP was originally designed to transport PSTN signaling messages over IP networks, but is also capable of serving as a general-purpose transport protocol. SCTP provides attractive features such as multi-streaming and multi-homing that may be helpful in high-mobility environment. SCTP congestion control mechanisms are based upon TCP congestion principals with the exception of the fast recovery algorithm. Original SCTP congestion control can perform badly in high-speed wide area networks because of its slow response with large congestion window. We proposed a new congestion control scheme based on the simple congestion window modification for SCTP to improve its performance in high-speed wide area networks. The results of several experiments we performed proved that our new suggested congestion control scheme for SCTP in high-speed network improved the throughput of the original SCTP congestion control scheme significantly.

1 Introduction

The SCTP is a reliable transport protocol operating on top of a potentially unreliable connectionless packet service such as IP [1]. It was originally designed to be a general-purpose transport protocol for message oriented applications, as is needed for the transportation of signaling data. It provides acknowledged, error-free, non-duplicated transfer of messages through the use of checksums, sequence numbers and selective retransmission mechanism.

In SCTP, a connection is referred as an association. An association is established through a four-way handshake as opposed to the three-way handshake in TCP. The passive side of the association does not allocate resources for the association until the third of these messages has arrived and been validated. This helps to avoid the issues of Denial of Service attacks to an extent. The most important features of SCTP are multi-streaming and multi-homing. The other enhancement features are message bundling, unordered delivery and path MTU discovery.

Multi-streaming is one of the most important features of SCTP, allowing data from the upper layer application to be multiplexed onto one channel. Sequencing

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 628–634, 2005.

of data is done within a stream. If a segment that belongs to a certain stream is lost, the segments from that stream following the lost one will be stored in the receiver's stream buffer until the lost segment is retransmitted from the source. However, since data from the other streams can still be passed to the upper layer application, the head of line blocking found in TCP can be avoided.

Multi-homing allows association between two endpoints to cross multiple IP addresses or network interface cards. If the nodes and the interconnection network are configured in such a way that the data from one node to another travels on physically different paths if different destination IP addresses are used, the association can become tolerant against physical network failures. The information about multiple addresses is exchanged at the time of association setup. One of the addresses is selected as the primary path over which datagram is transmitted by default. However, retransmissions can be done on one of the available paths.

SCTP uses an end-to-end window based flow and congestion control mechanism similar to the one that is used in TCP [1]. The receiver specifies a receive window size and returns its current size with all the SACK chunks. The sender maintains a congestion window to control the amount of unacknowledged data in flight. The acknowledgements contain a Cumulative TSN Ack that indicates all the data that has been successfully reassembled at the receiver's side. The Gap Blocks indicate the segments of data chunks that have arrived with some data chunks missing in between. If four SACK chunks have reported gaps with the same data chunk missing, the retransmission is done via the Fast Retransmit mechanism.

In high-speed wide area network (WAN), it is known that both performances of typical SCTP and TCP congestion control are deteriorated because of slow response time for bulk data transfer. In this paper, we suggest the new congestion control scheme based on the simple congestion window modification for SCTP to improve the throughput performance. The experiments show that our new scheme produces a significant performance.

We begin by presenting typical SCTP congestion control scheme and new SCTP congestion control scheme in the next section. Section 4 describes our experimental setup and results, followed by concluding remarks in section 5.

2 SCTP Congestion Control

Congestion control is one of the basic functions in SCTP. For some applications, it may be likely that adequate resources will be allocated to SCTP traffic to assure prompt delivery of time-critical data - thus it would appear to be unlikely, during normal operations, that transmissions encounter severe congestion conditions. However SCTP must operate under adverse operational conditions, which can develop upon partial network failures or unexpected traffic surges. In such situations, SCTP must follow correct congestion control steps to recover from congestion quickly in order to get data delivered as soon as possible. In the absence of network congestion, these preventive congestion control algorithms should show no impact on the protocol performance.

The advanced congestion control mechanism of SCTP consists of three basic algorithms. a) Slow-start b) Congestion Avoidance c) Fast Retransmit. The endpoints maintain three variables receiver advertised (rwnd), congestion window (cwnd) and slow start threshold (ssthresh) to regulate data transmission rate. SCTP requires an additional control variable partial bytes acked (pba) that is used during congestion avoidance.

Gap Ack Blocks in the SCTP SACK carry the same semantic meaning as the TCP SACK. TCP considers the information carried in the SACK as advisory information only. SCTP considers the information carried in the Gap Ack Blocks in the SACK chunk as advisory. In SCTP, any DATA chunk that has been acknowledged by SACK, including DATA that arrived at the receiving end out of order, are not considered fully delivered until the Cumulative TSN Ack Point passes the TSN of the DATA chunk (i.e., the DATA chunk has been acknowledged by the Cumulative TSN Ack field in the SACK).Consequently, the value of cwnd controls the amount of outstanding data, rather than (as in the case of non-SACK TCP) the upper bound between the highest acknowledged sequence number and the latest DATA chunk that can be sent within the congestion window.

3 New SCTP Congestion Control Scheme

Slow response and bulk transfers in high-speed WAN deteriorate the TCP performance. High-speed WANs have speeds greater than 100Mbps and round trip times above 50ms. Traditional TCP connections are unable to achieve high throughput in high speed wide area networks due to the long packet loss recovery times and the need for low supporting loss rates.

High Speed TCP [4] was recently proposed as a modification of TCP congestion control mechanism in high-speed wide area links. Scalable TCP is an evolution of the existing congestion control algorithm that improves performance when there is a high available bandwidth on long haul routes. It is designed to be easily implemented in current TCP stacks and incrementally deployable without needing modifications to network devices. Scalable TCP builds on the High Speed TCP and previous work on engineering stable congestion controls [8].

TCP congestion control algorithms are referred to as AMID (additive increase and multiplicative decrease) and are the basis of its steady state Congestion. TCP increases the congestion window by one packet per window data acknowledged, and halves the window for every window of data containing the packet loss. Packet loss is used as a signal of congestion; it is assumed due to buffer overflow due to offered traffic exceeding available capacity on the end-to-end path of a connection.

TCP senders update the congestion window in response to acknowledgments of received packets and the detection of congestion [3]. For each acknowledgment received in a round trip time in which congestion has not been detected, Congestion Avoidance

$$cwnd = cwnd + (1/cwnd)\ . \tag{1}$$

$$cwnd = cwnd/2\ . \tag{2}$$

This process of increasing and decreasing cwnd allows TCP to aggressively utilize the available bandwidth on a given end-to-end path. To alleviate this problem a High Speed TCP algorithm was proposed [3] and a similar approach is adapted in [4]. The High Speed TCP proposes two modifications.

The High Speed TCP response is represented by new additive increase and multiplicative decrease parameters. These parameters modify both the increase and decrease parameters according to cwnd. During congestion avoidance algorithm for each acknowledgment received in a round trip, the congestion window is increased by

$$Ack : cwnd = cwnd + [0.01 * cwnd]\ . \tag{3}$$

And on the first detection of the congestion, the congestion window is reduced by the equation

$$Drop : cwnd = cwnd - [0.125 * cwnd]\ . \tag{4}$$

The selection of the congestion window increase and decrease in equation (3) and (4) is based on design analysis of the networks possessing in large congestion networks in [4].

The congestion control of SCTP follows the same congestion window reduction mechanism as that of TCP. SCTP might behave in a similar way in the event of multiple packet losses. Depending on the size of the congestion window, new packets are injected into the network. If the congestion window is larger, then the data input to the network will automatically increase. Therefore we have modified the congestion control of SCTP as shown in equation (4). The results are presented in section 4.

4 Performance Evaluation

The general purpose of this work was to study the effectiveness of SCTP in high-speed wide area links as a mechanism for bulk transfer. This investigation was developed using a simple topology scenario to minimize complexity and reduce the number of variables. The network topology for test is shown in Fig. 1.

The topology consists of a sending and receiving host labeled (N1 and N2 respectively) connected via a drop-trail router labeled R. The parameters of the links are indicated in the figure 1.The experiments were conducted using NS-2

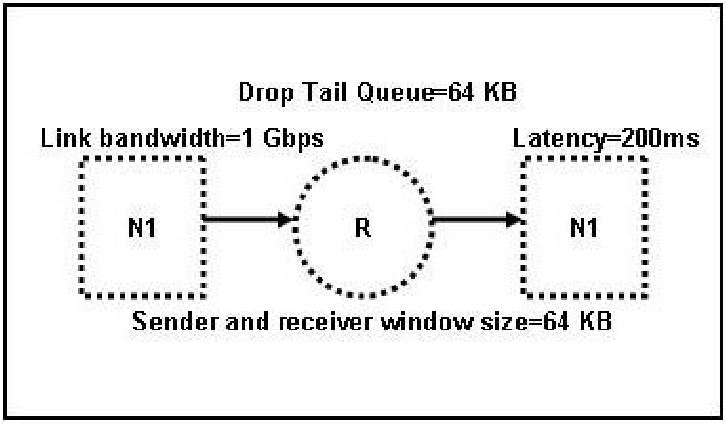


Fig. 1. Network topology

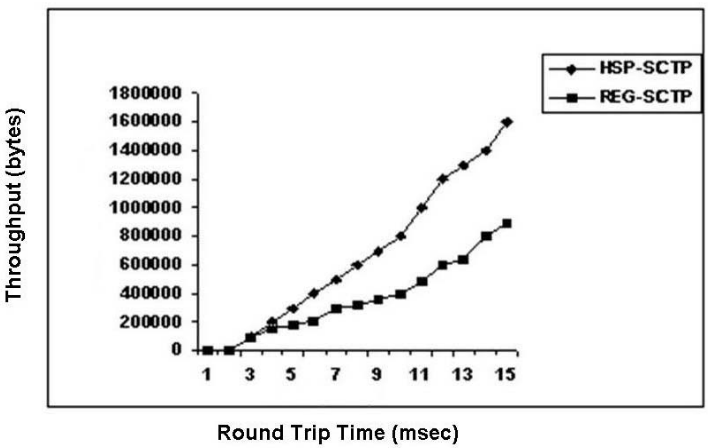


Fig. 2. Throughput Comparisions

simulator [5] and the SCTP module has been ported from Protocol Engineering lab at Delaware University [6].

This section presents throughput comparisons of the regular SCTP and the suggested high speed SCTP (HS-SCTP) in this paper from various experiments.

The results show that throughput of the HS-SCTP is 1.7Mbps whereas that of RG-SCTP is 0.9 Mbps as shown in Fig. 2. Performance results show an improvement of up to 53% over RG-SCTP was seen when the congestion control was modified.

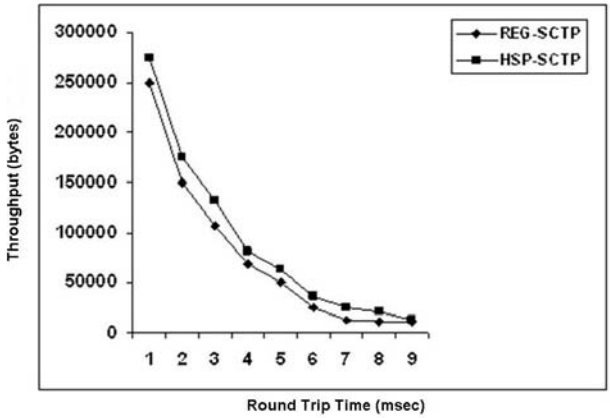


Fig. 3. Throughput Comparions

Further experiments show the effectiveness of HS-SCTP over LAN and WAN. The HS- SCTP does not improve the throughput over LAN, but for networks having latency in the range of 30 ms to 100 ms (RTT), there was a little improvement in the performance as shown in Fig. 3.

5 Conclusion

We suggest high-speed SCTP (HS-SCTP) with a simple modification to the typical congestion window update algorithm, which improves throughput in high-speed wide area networks. The performance improvement can be dramatic for senders using the high speed SCTP in bulk transfer networks; the improvement attributable to the algorithm was an average of 53 %. By adjusting the congestion window size of congestion control mechanism of SCTP, the performance of HS-SCTP is significantly better when compared to original SCTP. The HS-SCTP shows a throughput of 1.7Mbps when compared to the regular SCTP, which has a throughput of 0.9Mbps. HS-STCP can be used to bulk data transfer applications, because it is able to maintain high throughput in different network conditions, and it is easy to deploy when compared with other solutions already in use. Future extension of this work includes the performance evaluation for mobile networks and the development of more efficient scheme.

Acknowledgement. This work was supported by the Industrial technology development program,Ministry of Commerce, Industry and Energy, Republic of Korea.

References

1. Steward R., Xie Q.: Stream Control Transfer Protocol (SCTP), A Reference Guide, Addison-Wesley (2001)
2. Savage S.,Cadwell N., Wetherall D., Anderson T.: TCP Congestion Control with a Misbehaving Receiver, ACM Computer Communication Review, vol. 29, no. 5,(1999)
3. Kelly T.: Scalable TCP- Improving Performance in Wide Area Network (2002), available at, http://www.lce.eng.cam.ac.uk/ ctk21/scalable/
4. Flyod S.: High speed TCP for Large Congestion Windows, IETF (2003)
5. Network Simulator ns-2 available at http://www.isi.edu/nsnam/ns/
6. SCTP module for NS-2 available at http://pel.cis.udel.edu/
7. Alamgir R., Atiquzzaman, Ivancic W.: Effect of Congestion Control on the performance of TCP and SCTP over satellite Networks. Proceedings of NASA Earth Science Technology Conference, Pasadena (2002)
8. Kelly T.: On Engineering a Stable and Scalable TCP Variant. Technical Report CUED/FINFENG TR-435, Laboratory for Communication Engineering, Cambridge Univer sity (2002)
9. Allman M., Paxson V., Stevens W.: TCP Congestion Control. RFC-2581 (1999)

Improving a Local Search Technique for Network Optimization Using Inexact Forecasts

Gilberto Flores Lucio, Martin J. Reed, and Ian D. Henning

University of Essex, Wivenhoe Park Colchester, UK
{gflore, mjreed, idhenn}@essex.ac.uk
http://privatewww.essex.ac.uk/~gflore/

Abstract. This paper presents an evolutionary computation approach to optimise the design of communication networks where traffic forecasts are uncertain. The work utilises Fast Local Search (FLS), which is an improved hill climbing method and uses Guided Local Search (GLS) to escape from local optima and to distribute the effort throughout the solution space. The only parameter that needs to be tuned in GLS is called the regularization parameter lambda (λ). This parameter represents the degree up to which constraints on the features in the optimization problem are going to affect the outcome of the local search. To fine-tune this parameter, a series of evaluations were performed in several network scenarios to investigate the application towards network planning. Two types of performance criteria were evaluated: computation time and overall cost. Previous work by the authors has introduced the technique without fully investigating the sensitivity of λ on the performance. The significant result from this work is to show that the computational performance is relatively insensitive to the value of λ and a good value for the problem type specified is given.

1 Introduction

It is frequently stated that network traffic in practical networks has a tendency to grow over time, but that this growth is uncertain [1]. Within a business, forecasting is used to predict demands and this may include migration to different services with different needs for QoS in the network [2]. This paper starts with the premise that a network operator defines a set of future network demands as a set of traffic matrices each specifying predicted demand at a specific point (or epoch) in the future. In reality, predicting such demands is not an exact science and there is likely to be some uncertainty as to whether the demand will be actually required. Determining the demand and likelihood (stated here as a probability of demand) is a business level activity; this paper assumes the operator specifies them. At each point (or epoch) in the future the routing of each traffic demand (a commodity) has to be determined such that the operator achieves some business goal and this is the focus of this work. Specifically, this paper considers the goal of maximizing the use of the network at lowest cost whilst balancing load across the network; however, the approach is highly applicable to other goals. The planning of routes in a QoS enabled network with the associated constraints of integral fixed bandwidth path allocation and capacity constraints is

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 635–645, 2005.

classed as an *NP*-complete problem [3] (specifically an *integer multicommodity flow optimization* [4]). Consequently, finding an optimum solution is not practicable and even finding "good" solutions in an efficient manner is a challenge.

Evolutionary Computation (EC) techniques have proved to be valuable in this type of constrained optimization problems [5]. In particular, these schemes have been used successfully for the design and planning of communication networks [6]-[9].

The use of search optimization techniques in large networks (or general combinatorial problems) is more effective if the method focuses in the areas that are more likely to give improved solutions; this reduces considerably the solution space [10]. The approach used in this paper combines two EC techniques that use an efficient methodology of hill-climbing that focus in areas that look more promising in terms of solution quality. Specifically, this work uses Fast Local Search (FLS)[11] and Guided Local Search (GLS)[12].

The paper is organized as follow, Section 2 provides the problem description and application of GLS+FLS; Section 3 provides a general explanation of the network scenarios, experimental procedures and measures to test the efficiency of the technique. Section 4 presents and discusses the computational results of (GLS+FLS) and investigated the results to determine the sensitivity to the parameter λ.

2 Problem Formulation and Our Evolutionary Computation Approach to Network Optimisation

2.1 Multicommodity Flow Model with Uncertainties

Consider a network with a set of nodes $n \in N$ and a set of links $l \in L$, where link l has capacity μ_{lt} at time t. The traffic flow for a commodity $m \in M$ at time t, is defined as $\mathbf{X}_m$ from vectors $\mathbf{x}_{mt}$ each representing the flow of m on links $(1 \ldots L)$, x_{mtl} defines the flow of m in link l at time t. $\mathbf{X}_m$ is subject to the constraint:

$$\gamma_l = \sum_{m \in M} x_{mtl} \le \mu_{lt} \quad \forall l, \forall t \tag{1}$$

Where (1) defines the constraint that for all routes ($m \in M$) passing through a link l the assigned traffic (vector $\mathbf{x}$) is less than the total capacity μ_l on that link l. Note that all of these variables are dependent on time t.

The objective is to maximize the network usage (by deploying the maximum number of commodities M) at the lowest cost. This can be stated:

$$\text{Maximize } \beta = \sum_{m \in M} d_{mt} p_{mt} \tag{2}$$

Where: p_{mt} is the certainty of commodity m in time t of being deployed, and:

$$d_{mt} = \begin{cases} 1 \text{ if the commodity } m \text{ is accepted} \\ 0 \text{ if the commodity } m \text{ is not accepted} \end{cases}$$

And, for the maximum value of β minimize the cost:

$$C=\xi\sum_{m\in M}K(s_m,v_m)+\psi\sqrt{\sum_{l\in L}\left(\bar{l}-(\mu_l-\gamma_l)\right)^2}\;/L \tag{3}$$

Where ξ and ψ represent weight factors to suit specific application requirements, $K(s_m,v_m)$ represent the number of hops for each commodity accepted, $\overline{l}$ is the mean of the remaining resources in the whole network, $(\mu_l-\gamma_l)$ is the remaining resource of edge *l*. (3) aims to reduce the hop count and provide homogeneous flow throughout the network.

2.2 Local Search Techniques

Techniques that restrain the combinatorial nature of network design optimization do so by sacrificing the completeness of the solution [11]. Some of the better-known methods to tackle this are called local search methods, of which the most commonly used techniques utilize *hill climbing* [13]. The problem with this method is that it will probably settle in local optima. Methods like Tabu search (TS) [14] and Simulated Annealing (SA)[15] have been used to overcome this phenomenon. In this paper a method called *Guided Local Search* (GLS) is used to lead a local search away from local optima. A factor that affects considerably the performance of hill climbing is the size of the solution space. If there are many subspaces to consider the amount of computation could be very costly. In this paper, we use a method called *Fast Local Search* (FLS) to reduce the size of the neighborhood to evaluate in each iteration of the GLS. The combination of these two methods (GLS+FLS) has been shown to outperform both TS and SA in a number of combinatorial optimization problems [10], [16], [17]. Furthermore, GLS has the advantage that it only requires one internal variable (λ) to be tuned unlike many other techniques such as Tabu search [18] or Genetic Algorithms.

2.3 Guided Local Search

GLS repeatedly applies a local search heuristic and exploits the information obtained by the local search to guide it to more promising areas where better solutions can be found. To do this, the original objective function (3) is augmented so it can include a set of penalty stipulations. When the local search finds no improvement, the penalties are modified and local search is called again. The intention is to penalize bad features when the local search settles in a local optimum. As a result prior information is used to guide the search to reduce the number of solutions to be considered for evaluation. However, there is a risk that a bad feature in one iteration will be repeatedly penalized in each iteration and this may cause a feature to be “over” penalized. Consequently, a “utility” for each feature is introduced whereby the probability that the feature is penalized in subsequent iterations is reduced. By taking cost and the current penalty into consideration in selecting the feature to penalize, the search effort is distributed throughout the search space.

The generic GLS algorithm can be stated as [11]:

algorithm GLS+FLS(C, λ, G, F)

1 Create initial feasible solution $\mathbf{X}_0$

2 Set $\theta_j = 0$, $\forall\, j = 1 \cdots J$ (penalties set to zero)

3 **for** i:=1 **to** `maxiterations`

3.1 **begin**

3.2 $\mathbf{X}_i = FLS(\mathbf{X}_{i-1}, \Theta, C, G, \lambda)$

3.3 **for each** j: = 1 **to** J evaluate $u_j := \dfrac{d_j(\mathbf{X_i}) f_j}{1+\theta_j}$;

3.4 determine j that maximizes u_j and for this j update the penalty $\theta_j := \theta_j + 1$

3.5 **end**

4 **return** $\mathbf{X}i$ **where** $C(\mathbf{X}i)$ is minimum of all solutions;

Where C is the problem cost function, λ is a regularization parameter (used in the *FLS*), G represents the problem specification (e.g. network topology and demand matrix in our case). F represents a vector of problem specific multipliers $f_1 \cdots f_J$, one for each feature selected from the original objective function C. It could be said that each f represents a feature's "badness" in the solution. d is a decision function defined as

$$d_k(\mathbf{X}) = \begin{cases} 1, \text{if feature } k \text{ is in } \mathbf{X} \\ 0, \text{if feature } k \text{ is not in } \mathbf{X} \end{cases}$$

and $\Theta = \begin{bmatrix} \theta_1 & \cdots & \theta_J \end{bmatrix}$.

FLS can be generally described as an algorithm that breaks down the neighborhood of solutions into a number of smaller sub-neighborhoods. The order in which the sub-neighborhoods are selected is randomly chosen each time the FLS is performed. An activation-bit is associated with each of these sub-neighborhoods (a neighborhood is said to be active when this bit is set). The FLS iteratively applies perturbations to each sub-neighborhood and deactivates that sub-neighborhood when minimized or no better solution found. If an iteration produces an improved move in the current active sub-neighborhood the process activates adjacent sub-neighborhoods [19], deactivating the already minimized. The iterations continue until there are no remaining active sub-neighborhoods and the resulting solution is returned to the GLS.

In this implementation of FLS we define the sub-neighborhoods as sub-paths of each commodity that are shared by another commodity and apply a modification of the *approximate* 2-Opt method as the perturbation technique. The *approximate* 2-Opt method was proposed by J. J. Bentley [20] to solve the traveling salesman problem (TSP). The approximate 2-Opt swaps routes in the TSP and applies a repair function to maintain a feasible solution. Here the *approximate* 2-Opt is modified, as the repair function for TSP is not suitable for this problem class; we term this as modified *approximate* 2-Opt (ma2-Opt). Our implementation of ma2-Opt has been reported by others [16] so here we give a brief description. The ma2-Opt is applied to a sub-path that is shared by two commodities and produces a new solution. At each end of the common sub-path the two routes for each commodity are swapped for the preceding and following links; this produces a changed but unfeasible solution as only two new incomplete paths are formed. The repair function performs a shortest path route from the ends of the swapped links to try to produce a feasible solution. ma2-Opt moves that result in either a higher cost or cannot be repaired into a feasible solution are ignored.

The Fast Local Search algorithm is formally stated as:

algorithm FLS ($\mathbf{X}$, Θ , *C, G,* λ)

1 divide $\mathbf{X}$ into B sub-neighborhoods $\mathbf{x}_1 \cdots \mathbf{x}_B$ where each neighborhood represents a sub-path common to two or more commodities

2 associate activation bits $a_1 \cdots a_B$ with each sub-neighborhood and set each to 1

3 **while** any activation bit $a_1 \cdots a_B$ is 1 **do**

3.1 **begin**

3.2 **for** k:=1 **to** B

3.2.1 **if** $a_k == 1$ **then**

3.2.1 **begin**

3.2.2 Apply *ma*2-*opt* algorithm to two commodity paths sharing $\mathbf{x}_k$ using modified objective function $C'(C, \mathbf{X}, \Theta)$ giving new solution $\mathbf{X}'$

3.2.3 $a_k := 0$

3.2.4 **if** $\mathbf{X} \neq \mathbf{X}'$ **then** for all sub-neighborhoods adjacent to k set a:=1;

3.2.5 **end;**

3.2.6 $\mathbf{X} = \mathbf{X}'$

3.3 **end**

4 **return** $\mathbf{X}'$;

The modified objective function C' is defined as

$$C'(\mathbf{X}) = C(\mathbf{X}) + \lambda \sum_{k=1}^{S} \theta_k d_k(\mathbf{X}) \tag{4}$$

To apply a GLS to a specific problem it is only necessary to identify features in the objective function that should be examined and assign relative weights to them ($f_1 \cdots f_J$) according to application specification. Note that these are application relevant variables and not some obscure hidden parameters as found in many other EC techniques. As these variables are regularized by λ it is only the relative weights that are important. As an example of determining the features for GLS consider our own problem that has broadly three features in the objective function(s): minimize hop count, minimize overloading by spreading load, maximize number of commodities routed.

Thus the decision functions for this specific problem as may be stated as:

1. $d_1(X) = 1$ if any commodity exceeds a specific target hop-count along its routed path.

2. $d_2(X) = 1$ if there are edges that are overloaded beyond a maximum load value *i.e.* if there is a link l for which $\sum_{m \in M} x_{mtl} > \varepsilon_l \mu_{lt}$ where $\in_l$ represents the maximum loaded fraction of the capacity on that link.
3. $d_3(X) = 1$ if the solution has fewer routed commodities than the previous best solution.

GLS is related to the TS methodology although there are some important differences [18]. The most significant difference for this work is that, unlike TS, GLS requires only one parameter to be "tuned". The GLS tuning parameter is a regularization factor termed λ and it is required to determine an optimal value of this from experimentation with the problem search space. λ is normalized using:

$$\lambda = a \cdot \frac{g(\text{average min solution})}{N} \tag{5}$$

where g (average min solution) is the average minimal value of the initial FLS run 10 times in the network scenario with different sub-neighborhoods chosen at random; N is the number of nodes in the network; and, *a* is the new value that needs to be tuned. This approach permits the tuning of the GLS to be more problem independent. For the specific problem in this paper, the initial solution $\mathbf{X}_0$ is generated deterministically by the modified custom Dijkstra Algorithm [21]. This generates a solution following shortest paths in terms of hop count but which obeys capacity constraints.

3 Experimental Procedures

To evaluate the effectiveness of the algorithm, 3 different networks with different levels and inexact traffic forecasts were used. The first network scenario is conformed by 10 nodes and 15 edges, the second 19 and 30 edges and the last scenario with 35 nodes and 50 edges. The value of λ was varied over first a wide range and then an increasingly small range so that an optimal value could be determined. The experiments were conducted in an Intel® Pentium® 4 PC 1.7Ghz with 512 Mbytes RAM in a Linux O.S using C++ where speed of execution in this environment was used for the speed comparisons.

Two performance measures are used to compare the behavior of the (GLS+FLS) approach with gradual changes in the parameter (λ) for the 3 scenarios: the overall cost and the computation time. Each of them took to obtain an equivalent solution in terms of number of commodities "routed", in this case 20 commodities. The stop criterion was no improvement in 100 iterations. According to the behavior of the results both in terms of computation time and overall cost, the range of *a* defined in (5) was reduced as well as the number of steps. Finally, a set of experiments for the smallest and biggest network scenarios used (10 and 35 nodes) was conducted to route 20, 30 and 40 commodities and test the sensitivity of the performance to changes in λ. The certainty of these planned commodities is shown in Table 1, expressed as a probability *p*. For example a commodity *m* with p_m=1 represents a planned commodity that is certain to be required whereas a value of p_m=0 represents a commodity that will never be required. Commodities with a higher value of certainty will have more priority to be deployed in the network.

Table 1. Commodities used for the 12 network scenarios where P is the certainty that the commodity will be required (P =1 certain to be required, P=0 not required)

No. Of Commodities	Certainty Value
20	80% placed with p=1.0 20% placed with p=0.75
30	70% placed with p=1.0 20% placed with p=0.75 10% placed with p=0.5
40	60% placed with p=1.0 25% placed with p=0.75 10% placed with p=0.5 5% placed with p=0.25

The overall cost of each solution is calculated by combining two values: the number of requirements supported by the established network and the cost of that network in "economic units". In order to calculate the amount of economic units that a solution can have, two factors are considered. The first factor is the total number of hops that each path of each requirement has (every hop is considered an economic unit). The second factor is obtained by calculating the standard deviation of the resources that the solution leaves after deploying the requirements (the total value is passed as economic units), this factor is to define how well the load in the network is distributed (wider distribution is given lower cost). At the end, the solution with the highest number of commodities is selected. If two or more solutions are equal in this respect, the one with the lowest number of economic units will be considered the best.

4 Computational Results

4.1 Fine Tuning the Regularization Parameter Lambda (λ)

As mentioned in section 2.2, the only parameter required to be tuned for the GLS is the regularization parameter (λ) that represents the degree up to which constraints on the features are going to affect the outcome of the local search [19]. In order to evaluate the impact that this parameter has over the solution, the 3 network scenarios were tested initially with a wide range of values for a (from 0 to 1000) to obtain values of λ. After a number of experiments, the range of a was then reduced from 0 to 20 due to the extended amount of time it took to converge when the value of λ was considerably high (λ typically more than 10000) for the 3 scenarios. Fig. 1 shows the results obtained for the value of a in terms of computation time for the 3 network scenarios.

scenarios. Fig. 2 shows the same results but in terms of overall cost. Again, the stop criterion was no improvement in 100 iterations.

The results obtained provide us with guidelines to test a range of a at smaller steps. The focus was in a range from 0.01 to 0.30 obtaining the best values of a between 0.2 and 0.22 for all scenarios, where the value of a is 0.21 for the 10 network scenario, 0.198 for the 19 and 0.2 for the 35 node network. The difference in terms of computation time between the 3 scenarios is considerable when the value is fine-tuned. While in the largest network (35 nodes with 50 edges), the worst-case time it took to obtain the "optimal" solution was 54.56 hours (a=19.97); the best overall fine-tuned solution for the same case took only 0.328 hours (a=0.194). The comparison in terms of overall solution quality (number of commodities placed and cost) did not have a remarkable impact in the any of the cases. The reason for this is because in each iteration, the best value is always saved until a better solution is found. Nevertheless, the fine-tuned results had a small improvement in terms of cost; this is probably because with lower values of a the GLS is allowed to improve on some highly probable local minima without the excessive changes in regions searched by each iterative FLS. Higher values of a give poor convergence times as each iterative FLS can make excessive changes without concentrating on finding local minima.

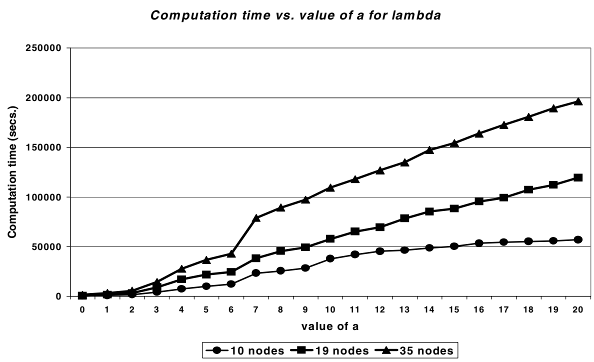


Fig. 1. Comparison of the solution quality in terms of computation time for different values of a in the 3 network scenarios

4.2 Comparing the Effectiveness of the Fine-Tuning with Several Sets of Inexact Forecasts

For the 2 network scenarios with 10 and 35 nodes, the best values for overall cost, computation time and number of accepted requirements were obtained. These scenarios were used with 20, 30 and 40 different commodities and the certainty of these planned commodities expressed as a probability p. Table 2 shows the 3 different sets of commodities used along with the value of certainty to occurred, amount accepted and the best values of λ found for both scenarios. This set of commodities was used for the two network scenarios. Commodities with a higher value of certainty will have more priority to be deployed in the network.

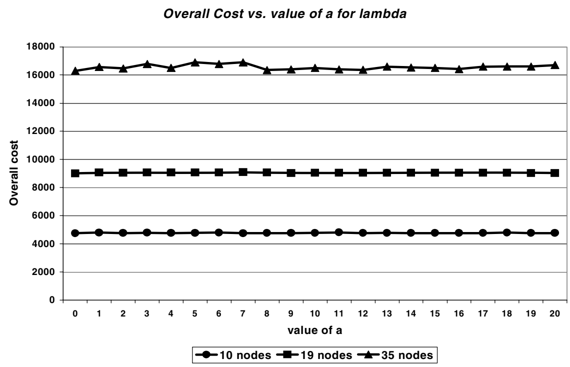


Fig. 2. Comparison of the solution quality in terms of Overall cost for different values of *a* in the 3 network scenarios

The first conclusion from the results is that regardless of the number of commodities deployed and network size (within the range that was investigated), a reasonable value for λ is for a=0.2 ± 0.02. It is hypothesized that this value is relatively insensitive for a wider range of network sizes and commodity scenarios. One reason for this relative insensitivity in the value of a (and thus λ) is that the GLS has an element of self tuning through the initial sample of the search space by a run of 10 randomly placed FLS iterations.

The second conclusion from these results is that this algorithm can be used to combine a number of priorities in commodity placement. In this work we are most interested in ranking this priority through the likelihood of the commodities being

Table 2. Commodities used for the 2 network scenarios where P is the certainty that the commodity will be required (P=1 certain to be required, P=0 not required)

Comm.	Certainty Value	Commodities Accepted for 10 nodes	Value of (λ) for 10 nodes	Commodities Accepted for 35 nodes	Value of (λ) for 35 nodes
20	16 placed with p=1.0 4 placed with p=0.75	16 accepted 4 accepted	95.06	16 accepted 4 accepted	93.2
30	21 placed with p=1.0 6 placed with p=0.75 3 placed with p=0.5	20 accepted 5 accepted 2 accepted	85.5	21 accepted 6 accepted 2 accepted	97.86
40	24 placed with p=1.0 10 placed with p=0.75 4 placed with p=0.5 2 placed with p=0.25	22 accepted 7 accepted 2 accepted 1 accepted	90.25	24 accepted 9 accepted 3 accepted 1 accepted	93.2

needed, however, it is applicable to other applications for example prioritized QoS routes. The priority given by the algorithm to fulfill the commodities with a higher probability to happen is shown independently of the size of the network.

5 Conclusions

An evolutionary computation approach to optimize the design of communication networks were traffic forecasts are uncertain was presented. A meta-heuristic method called Guided Local Search (GLS) in combination with an improved method for hill climbing called Fast Local Search (FLS) were used. The regularization parameter lambda (λ) is the only parameter in GLS to be tuned and this was further normalized by the problem size (network dimension in this case). To tune this parameter, a series of evaluations were performed in several network scenarios to make the search more efficient for a specific problem class. The optimisation was required to route different sets of commodities with diverse levels of certainties and with minimum network cost. The results showed the effectiveness of the proposed methodology in this Multicommodity Flow problem (MCF) in 3 different network scenarios when the parameter is tuned. A significant finding is that the technique is relatively insensitive to the normalsized value of the regularization parameter λ for the problem class considered.

References

1. Marbukh V., " Network Provisioning as a Game Against Nature" in the IEEE International Conference on Communications ICC 2003, Anchorage Alaska USA 11-15 May 2003.
2. Tohru Ueda; " Demand Forecasting and network Planning Methods under Competitive Environment"; in the IEICE Transactions in Communications, Vol. E80-B, No. 2, February 1997, pp. 214-218.
3. Dengiz Berma, Altiparmak Fulya, Smith E. Alice; "Local Search Genetic Algorithm for Optimal Design of Reliable Networks"; in the IEEE Transactions on Evolutionary Computation, Vol. 1, No. 3; September 1997, pp. 179-188.
4. Awerbuch Baruch, Leighton Tom; " A Simple Local- Control Approximation Algorithm For Multicommodity Flow"; in the Proceedings of the IEEE 34th Conference on Fundamentals of Computer Science, Oct. 1993.
5. Jong-Hwan Kim, Hyun Myung; " Evolutionary Programming Techniques for Constrained Optimisation Problems"; in the IEEE Transactions on Evolutionary Computation Vol. 1, No. 2, July 1997, pp. 129-140.
6. Dengiz Berna, Altiparmak Fulya; " A Genetic Algorithm Approach to optimal Topological Design of All Terminal Networks" in the Intelligent Engineering Systems Through Artificial Neural Network, Vol. 5, 1995, pp. 405-410.
7. Jaroslaw Arabas, Stanislaw Kozdrowski; " Applying an Evolutionary Algorithm to Telecommunication Network Design"; in the IEEE Transactions on Evolutionary Computation; Vol. 5, No.4, August 2001, pp. 309-322.
8. Awerbuch Baruch, Leighton Tom; " A Simple Local-Control Approximation Algorithm For Multicommodity Flow"; in the Proceedings of the IEEE 34th Conference on Fundamentals of Computer Science, Oct. 1993.

9. Leighton Tom, Makedon Fillia, Plotkin Serge, Stein Clifford, Tardos Eva, Tragoudas Spyros; " Fast Approximation Algorithms For Multicommodity Flow Problems"; in the Proceedings of the 23rd Annual Symposium on Theory of Computing, 1991, pp. 101-111.
10. Tsang P. K. Edward, Voudouris Christos; " Fast Local Search and Guided Local search and their application to British Telecom's workforce scheduling problem"; the Operations Research Letters 20, Elsevier Science Publishers 1997, pp.119-127.
11. Tsang P. K. Edward, Wang J Chang, Davenport Andrew, Voudouris Christos, Leng Lau Tung; " A family of Stochastic Methods For Constraint Satisfaction and Optimisation", in the First International Conference on The Practical Application of Constraint Technologies and Logic Programming, London, April 1999, 359-383.
12. Voudouris Christos, Tsang P. K. Edward; " Guided Local Search Joints the elite in Discrete Optimisation"; in DIMACS Series in Discrete Mathematics and Theoreti- cal Computer Science, Volume 57, 2001, pp. 29-39.
13. Michalewicz Zbigniew; " Genetic Algorithms + Data Structures = Evolution Programs", Springer-Verlag Editorial, 1992.Part 1, pp. 16-18.
14. Glover Fred, Laguna M., " Tabu Search", in C. Reeves Modern Heuristic Techniques for Combinatorial Problems, Blackwell Scientific Publishing, Oxford, 1993, pp. 71-141.
15. Feng-Tse Lin, Cheng-Yan Kao, Ching-Chi Hsu; "Applying the Genetic Approach to Simulated Annealing in Solving Some NP-Hard Problems" in the IEEE Transactions on Systems, Man and Cybernetics, Vol. 23, No. 6, November/December 1993, pp. 1752-1767.
16. Voudouris Christos, Tsang P. K. Edward; " Guided Local Search and its Application to the Traveling Salesman problem"; in the European Journal of Operational Research, Vol.113, No.2, November 1998, pp. 80-110.
17. Lau T. L., Tsang P. K. Edward; " Guided Genetic Algorithm and its Application to radio Link Frequency Assignment Problems"; International Journal of Constraints, Kluwer Academic Publishers Vol. 6, No. 4, October 2001, pp. 373-398.
18. Hertz Alan; " Finding a Feasible Course Schedule Using Tabu Search", in the Discrete Applied Mathematics and Combinatorial Operations Research and Computer Science, Vol. 35, 1992.
19. Voudouris Christos, Tsang P. K. Edward; " Guided Local Search" Technical reports CSM-247, Department of Computer Science, University of Essex, UK, August 1995, pp. 1-18.
20. Bentley J.J.; "Fast Algorithms for Geometric Traveling Salesman Problems", in the ORSA Journal on Computing Vol. 4, 1992, pp. 387-411.
21. Kershenbaum Aaron; "Telecommunications Network Design Algorithms"; Computer Science Series, McGraw-Hill Ed., International Editions 1993, pp. 157-159.

Distributed Addressing and Routing Architecture for Internet Overlays

Damien Magoni[1] and Pascal Lorenz[2]

[1] Université Louis Pasteur – LSIIT,
Boulevard Sébastien Brant, 67400 Illkirch, France
magoni@dpt-info.u-strasbg.fr
[2] Université de Haute Alsace – GRTC,
34 rue du Grillenbreit, 68008 Colmar, France
lorenz@ieee.org

Abstract. A growing number of network applications create virtual networks called overlays on top of the Internet. Because advanced communication modes such as multicast have not been able to be successfully deployed at the network layer, they are now being implemented at the application layer thus creating such virtual networks. However these overlays require some form of addressing and routing inside themselves. Usually their topology is kept as simple as possible (e.g. tree, ring, etc.) but as their size increases, the need to be able to cope with a non trivial topology will increase. Our aim is to design a simple but robust addressing and routing scheme for topologically complex overlays. The only assumption is that the overlays are built upon the Internet and thus their topologies are constrained by the Internet topology. The benefit of our architecture is that they will not have to set up and maintain specific trivial topologies. In this paper we present the mechanisms of our distributed addressing and routing scheme. We have carried out simulations and we present some performance results of our routing algorithm such as path inflation and resistance to network dynamics.

1 Introduction

Designing a light-weight application level addressing and routing architecture is not easy when no constraint is put on the topology of the members. For instance, setting up a tree topology is easy but provides very little robustness. Complex mechanisms must be used to recreate the tree in case of branch failures. The advantage of permitting a free topology only restrained by the underlying network (i.e. the Internet for our purpose) is that it is very easy to add nodes and redundant links provide increased robustness. We have designed an architecture that puts no constraint on the topology of overlays. Thus we have to define a routing mechanism to route data inside the overlay. Our routing mechanism is addressing-driven (i.e. path information is stored in the addresses). The core principle of our architecture is that the nodes do not store routing tables in

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 646–653, 2005.

the usual way (i.e. tables containing the addresses of all the possible destinations). Each node only needs to store the addresses of his neighbors in order to properly route packets towards their destination. Thus its routing table is only composed of its neighbors' addresses (which are usually not numerous). However this also means that the path towards the destination is partly contained in the address itself and thus it is usually not optimal. Our architecture implies a trade-off between routing table size and path length. The smaller the first, the bigger the second and vice-versa. Our paper contains three sections. In section 2 we briefly present prior and related work on compact routing protocols. In section 3 we describe concisely how our addressing and routing solution works. Finally in section 4 we present some path length performance results obtained by simulation for assessing the efficiency of our solution.

2 Related Work

The trade-off between routing table sizes and path lengthes has been actively studied by the distributed algorithm community. Theoretical work by Eilam *et al.* has proved in [1] that it is possible to bound the average stretch (i.e. path length inflation) by 3 with routing tables of size $O(n^{3/2}\log^{3/2}n)$. Similarly Cowen has proved in [2] that it is possible to bound the maximum stretch by 3 with routing tables of size $O(n^{2/3}\log^{4/3}n)$. However in all these cases the table sizes are still a function of n which may not scale in real implementations even if this function is sub-linear. Furthermore they use an unique adequate labelling for every vertex to achieve their goal and they do not describe how to do it in a widely distributed environment. In this paper we present an architecture where table sizes are not a function of the network size. Although our architecture does not provide an upper bound on the average stretch, it is usually kept below 3 which seems bearable as shown in section 4.

3 Architecture Description

In this section we describe how our architecture works. In order to make the addressing and the routing scalable, we define a hierarchical addressing. In order to make the routing reliable to network dynamics, we also define a dynamic addressing.

3.1 Address Structure

Each overlay node has an address. An address is composed of one or several fields containing numbers and separated by dots, one field for each level of the hierarchy. Each field of an address is also called a label. The level of the address is equal to the number of fields in the address. The prefix of an address is equal to the address without the latest field. The last field is called the local field or local label. The number of levels in the hierarchy is theoretically unlimited and thus technically extensible. Each node in the overlay network has at least one address and typically more in order to cope with the network dynamics as explained later.

3.2 Hierarchical Addressing

The addressing plan contains zones that correspond to the address fields. The label size thus defines the maximum zone size. All zones have the same fixed size n (called the zone size). There is one level 1 (i.e. top level) zone containing n nodes (defined in the first address field). Then there are at most n level 2 zones each containing at most n nodes (defined by the first two address fields). Then there are at most n^2 level 3 zones each containing at most n nodes and so on. This means that all the address space can be theoretically distributed and if we have k levels and l bits per level, we can address $2^{k \times l}$ nodes.

The addressing of the overlay nodes is fully distributed: it relies only on local knowledge in a node. The only local knowledge we rely on for the moment is the degree of the node and the addresses and degrees of its neighbors. Any node is supposed to know this information. Let us assume that the zone size is n. Each node that has address $w.x.y$ is responsible for attributing the following addresses to its neighbors:

- the address $w.x.(y+1)$ (called a "next" address) where $(y+1) \leq n$,
- the address $w.x.y.1$ (called a "down" address),
- the addresses $w.x.y.z$ (called a "leaf" address) where $z > n$.

The first node of the overlay takes the address 1. Nodes join the overlay successively by connecting themselves to already connected ones. When a new node want to become part of the overlay, it asks for address proposals to all its neighbors. Each neighbor proposes an address to the newcomer (from the above list possibilities) that it has not already given to its other neighbor nodes. The newcomer then chooses the most suitable address: usually the shortest next or down address that belongs to a node with a high degree (i.e. number of connections).

A said above, a leaf address is an address whose local label is above the zone size value (e.g. if the zone size is 32, the first leaf label is 33). Nodes that have a leaf address can only route data to their father even if they are connected to other nodes, they are considered as leaf nodes for the routing system. That is why they are chosen by newcomer with the lowest priority. Figure 1 illustrates a small network of nodes addressed using our architecture.

3.3 Hierarchical Routing

The aim of the hierarchical addressing is primarily to enforce the construction of local zones of limited size in order to fragment routing.

Hierarchical routing works in the following way. When a packet is routed in a node, if the destination address in down this node hierarchy, the packet is driven to the node of the current zone that leads further towards the destination zone (we call it the ingress node). If the destination address is not down the current node hierarchy, the packet is driven to the first node of the zone (i.e. the one with a local label of 1) in order to be sent to the upper level zone (we call it the egress node). When a packet is routed inside a zone because the destination is in the zone or to go to the ingress or egress node, it is routed by a technique that we call the closest label. This technique only needs to know the addresses of the neighbors, that is why it performs a stateless routing.

The closest label routing technique works as follows. If the destination local label is lower than the current node local label, then the packet is forwarded to the neighbor node which has the lowest local label higher or equal to the destination local label. If the destination local label is higher than the current node local label, then the packet is forwarded to the neighbor node which has the highest local label lower or equal to the destination local label. As the neighbors have a continuous label assignment, this technique ensures that the packet will reach its destination although not necessarily by a shortest path.

Figure 1 illustrates the effects of hierarchical routing and flat routing between the nodes 4.1 and 2.2 on path length. The hierarchical routing forces the message to be routed via 3 and 2 thus giving a path length of 5 hops while a flat shortest path routing requires only 4 hops via 5 to reach the destination. We define the path length ratio as the value of the hierarchical routing path length in hops divided by the flat routing path length in hops. We also call it the routing overhead (with respect to the number of hops).

To conclude with hierarchical addressing and routing we can say that there is a clear trade-off between the amount of network topology knowledge in the nodes and the routing efficiency with respect to route lengthes.

Hierarchical Trade-off: the more we create hierarchy in the network, thus reducing routing information to local topology, the more we increase the route lengthes compared to their corresponding shortest paths.

3.4 Dynamic Addressing

As we can see in our architecture if a node moves or fails (thus making its address invalid), all packets routed to a destination that contains the address of the moving or failed node will not be able to be routed anymore. To solve this issue, each node takes (and thus is identified by) several addresses (i.e. more than one). The additional addresses can be chosen at the time of insertion in the overlay (i.e. newcomer node) or later on when the overlay connectivity changes

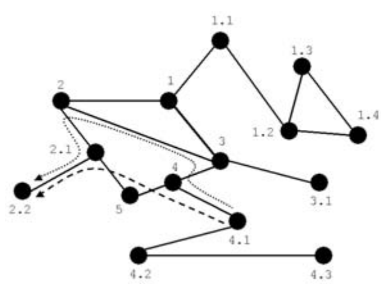


Fig. 1. Hierarchical addressing and routing example

and more addresses become available to the node as a result. All the addresses owned by a given node must come from different neighbors. All the addresses owned by a given node must be different, that is they must not have a common prefix. Otherwise if the disappearing node address is included in the common prefix, all addresses will not work. This multiple address solution increases the amount of routing information to be stored by a factor equal to the number of addresses per node but the advantage is that the network dynamics are handled transparently by the addressing protocol. If a packet is blocked because a node has disappeared, it can use one of the alternate addresses to get through to the destination. Two solutions are available here: either the packet carries all the destination addresses and thus it can be rerouted on the fly by using its alternate addresses (but this uses some more bandwidth) or a warning message is sent back to the source which then will change the address by an alternate one in all the future packets. Invalid addresses will be given a time-out value and will be attributable again at the end of the time-out if the owning node does not reappear again (e.g. in the case of a temporary failure).

To conclude with dynamic addressing and routing we can say that there is a trade-off between the loss of addressing space (as alternate routing information increases) and the ability of our architecture to handle mobility (node movement) and reliability (node failure).

Dynamic Trade-off: the more we distribute alternate addresses in the network, thus increasing routing reliability to network dynamics, the more we increase the address space consumption and routing information storage.

4 Experiments

In this section we present the results obtained by simulation for evaluating the efficiency of our architecture. We show that our hierarchical addressing and routing architecture has acceptable routing overhead performances and good resistance to network dynamics.

4.1 Settings

Ensuring the accuracy of the topologies used for our simulations on addressing and routing is crucial as the results heavily depend on them. Thus, to evaluate our addressing and routing protocols, we have used 3 Internet maps gathered by our software *nec* in 2003. We assume on first approximation that the overlays can be accurately modelled by these maps (as these maps are subgraphs of the Internet topology) or by subgraphs of these maps when we study network dynamics.

For path inflation, we have analyzed various addressing plans by using zone sizes ranging from 32 to 32768. For network dynamics, we have analyzed periodical percentage of random node removal ranging from 0 to 75% of the overlay size and attributing 1 to 8 (at most) addresses to each node.

As the process of generating addressing plans and selecting source-destination nodes for routing involves random selection (and thus random rolls), we have used a sequential scenario of simulation [3] to produce the results shown in the next section. As the random rolls are the only source of randomness in our simulation, we can reasonably assume that the simulation output data obey the central limit theorem. We have performed a terminating simulation where each run consists in picking two nodes and determining the flat and hierarchical distances (path length) between them (i.e. one run is the time horizon) as well as the success of hierarchical routing (when the overlay is not connected).

Network dynamics are a macroscopic way to simulate the addition, removal, movement and failure of the overlay nodes. At the beginning of the simulation all nodes belong to the overlay. Before the simulation starts, a given x % of nodes are randomly selected and removed from the overlay. After every 10 runs, all the removed nodes are re-inserted in the overlay and again the same % of nodes are randomly selected and removed from the overlay. Although x remains the same, the actual nodes that are removed at every 10 runs will be different most of the time especially when x is low. This simulates the addition, removal, movement and failure of the overlay nodes while keeping the size of the overlay equal to $100 - x$ %. When we have simulated network dynamics we have determined if the hierarchical routing is successful or not.

All the simulation results have been obtained assuming a confidence level of 0.95 with a relative statistical error threshold of 5% for all measured metrics.

4.2 Results

Figure 2 shows the values of the ratio between the path length provided by the hierarchical routing and the shortest path length (flat routing) as a function of the overlay size. We call this ratio the *path length ratio*, the *path inflation* or the *routing overhead*. A ratio of 2 for instance means that on average a hierarchical path is twice as long as its corresponding shortest path (i.e. between the same nodes). The plot labelled random origin corresponds to the case where the first overlay node is randomly chosen. The plot labelled specific origin corresponds to the case where the first overlay node is the highest degree node of the map. With a random first node, the path length ratio is linear with respect to the overlay size with values roughly ranging from 2.3 to 2.9 for a 75k-node overlay. This indicates a weak scalability but the slope coefficient is very and low thus the inflation remains acceptable for overlays under 100k-node. However when the first node is the biggest the plot falls back: this is an interesting property. This gives us an lower bound on the path inflation if we can manage to re-attribute the address of the first node to a bigger one. Figure 3 shows the values of the path length ratio as a function of the zone size. The path inflation does not depend on the zone size whatever the map and the origin. This means that we can label large overlays with large zones without having to create too many levels.

Figure 4 shows the percentage of successful routing attempts as a function of the network dynamics percentage. As explained above, a given percentage of nodes are absent, thus the overlay may not be connected but composed of

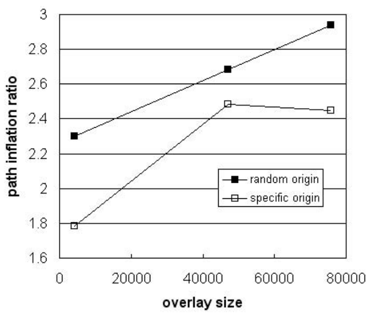


Fig. 2. Path length ratio *vs* overlay size

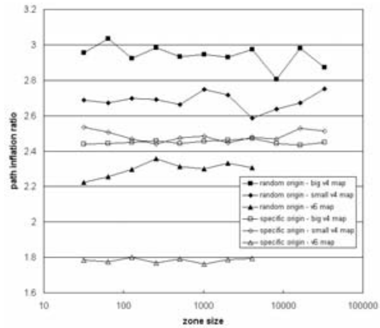


Fig. 3. Path length ratio *vs* zone size

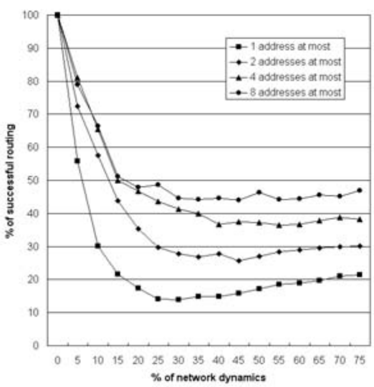


Fig. 4. Percentage of success *vs* network dynamics for the v6 map

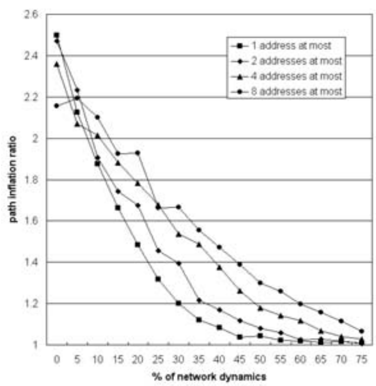


Fig. 5. Path inflation *vs* network dynamics for the v6 map

multiple connected components. The percentage is calculated as the number of successful hierarchical routing attempts divided by the number of successful flat routing attempts. As the hierarchical path is longer than the flat (i.e. shortest) path, it may go out of the source-destination component and thus it will make the routing fail. We can see that with only one address (i.e. no route alternative), 15% of dynamics makes the success fall at 20%. However the addition of addresses to the nodes heavily increases the routing success. With up to 4 addresses per node and 15% of dynamics, the success reaches 50%. Increasing the maximum number of addresses per node does not linearly improve the success because the maximum number of addresses per node is still bounded by its neighborhood size (and this is small for most of the nodes because of the underlying Internet topology). Figure 5 shows the path inflation as a function of the network dynamics percentage. The path inflation decreases with the dynamics because both the hierarchical and flat path lengthes decrease. As the network becomes more fragmented (high dynamics), the connected components become smaller and so do their inner paths. This explains why the success plots with nodes having 1 or 2 addresses at most, reach a minimum and increase again with high dynamics. When the components are small, the hierarchical and flat paths are closer and thus the hierarchical path will less likely be broken (i.e. it will remain inside the component). It's worth reminding that any routing protocol that do not provide shortest paths (e.g. intra-routing protocol plus BGP) is subject to routing failure if the calculated path exits a connected component.

To conclude this section we have seen that our architecture provides a stateless routing system (i.e. only neighbor addresses are stored) with a reasonable path inflation of 2 to 3 for overlays of sizes up to 75k (with a lower bound of 2.5 for 10k magnitude overlays). It also provides adaptation to network dynamics with reasonable performances: 80% of success when 5% of the overlay is changing to 45% of success when 50% of the overlay is changing. We are currently working on improving the addresses attribution: this will lower the hierarchical path lengthes thus reducing the path inflation and increasing the robustness to network dynamics.

5 Conclusion

In this paper, we have proposed a distributed addressing and routing architecture designed for random topology overlay networks set up on the Internet. Our simulation results obtained upon three realistic Internet maps of 4k to 75k nodes have shown that our solution yields a routing overhead ranging between 78% to 193% depending on the overlay size and the first node location. We have described how to cope with network dynamics and our simulation results have shown that simply attributing several addresses to each node without any other recovery mechanism multiply by 2 the routing success percentage when the network dynamics are above 10%. We are currently implementing our addressing and routing mechanisms in a host-level network middleware. This middleware will enable us to evaluate the behavior of our architecture in real life and confirm or invalidate our simulation results.

References

1. Eilam, T., Gavoille, C., Peleg, D.: Compact routing schemes with low stretch factor. In: Proceedings of the 17th ACM Symposium on Principles of Distributed Computing. (1998) 11–20
2. Cowen, L.: Compact routing with minimum stretch. In: Proceedings of the 10th ACM-SIAM Symposium on Discrete Algorithms. (1999)
3. Law, A., Kelton, W.: Simulation Modelling and Analysis. 3rd edn. McGraw-Hill (2000)

On Achieving Efficiency and Fairness in Video Transportation

Yan Bai[1], Yul Chu[2], and Mabo Robert Ito[1]

[1] Department of Electrical and Computer Engineering, University of British Columbia, Vancouver, BC V6T1Z4, Canada
{yanb, mito}@ece.ubc.ca
[2] Department of Electrical and Computer Engineering, Mississippi State University, P. O. Box 9571, Mississippi State, MS 39762-9571, USA
chu@ece.msstate.edu

Abstract. This paper proposes an intelligent scheme to enhance Quality of Service for video streaming over IP networks. The idea is to discard packets intelligently at a router in Active Networks (AN) before the buffer is full. This paper also presents an AN-based network node architecture to support the proposed scheme. Our simulation results show that it improves not only the visual quality per video stream but also network efficiency significantly.

1 Introduction

This work is motivated by both technology "pull" and "push." The "pull" is the emergence of active networking technologies. Active Networks (AN) are a novel approach in the field of networking research [1]. In Active Networks, routers may perform complex computation when forwarding packets, rather than just simple storing and forwarding packets, as in traditional networks. In other words, active routers examine and possibly modify the packet content, whereas conventional routers are limited to processing only the headers. Moreover, users can program the network to choose the specific processing that their packets will experience while the packets traverse through the routers. Recent research shows that many AN-based schemes provide benefits for reliable multicast, network management, traffic control, and multimedia applications, mobile IP services and grid computing [2, 7]. Our innovation is to leverage and extend Active Networking technology for use in other areas not really covered today in ways that will greatly benefit video over IP-based networks, in particular, buffer management techniques.

The "push" comes from the limitations in buffer management techniques. Buffer management plays an important role in enhancing Quality of Service (QoS) for video streaming over IP networks. It adjusts buffer occupancy to prevent congestion at routers, thus decreasing packet loss and improving video quality. The most common buffer management technique is Drop Tail (DT) in which packets are dropped when the buffer is full. Since it can deteriorate the video quality, recent researches have proposed proactive discard approaches to ensure that the buffers (queues) will not actually reach their full discard thresholds. Representative techniques are Random Early Detection (RED) [4] and its variants including RED with Input and Output

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 654 – 661, 2005.

(RIO) [3], RED with Priority Dropping [8] and LRU-RED [6]. In these schemes, an arriving packet is randomly discarded with a probability proportional to the average queue length of a router when a preset threshold is exceeded. They are not stand-alone mechanisms and rely on joint use of rate control techniques in order to reduce the packet loss for networked video. Furthermore, most existing buffer management schemes are content-blind. They transfer data between nodes without knowledge or modification of data content. They manage issues such as packet loss rate and congestion. However, loss distribution, which has a significant impact on video quality, cannot be effectively controlled. Since low packet loss ratios do not necessarily translate to high video quality, these methods do not improve the video quality as much as expected.

Given these limitations, we propose: a) an intelligent packet discard scheme based on the active networking paradigm, and b) an AN-based node architecture for supporting the proposed scheme. The rest of this paper is organized as follows: Section 2 describes the proposed schemes. Section 3 presents the simulation results. Section 4 discusses the proposed node architecture. Finally, Section 5 gives conclusions.

2 Intelligent Packet Discard

This section describes the proposed intelligent packet discard scheme, called IPD, for MPEG video transport over IP networks. The design objectives are to achieve high perceived video quality, high effective throughput and a high level of fairness of service. IPD consists of two parts: per-node-based packet drop (PPD) and inter-node loss allocation (InterLA). PPD is designed from a perceived quality control viewpoint, rather than from the network measurement viewpoint (e.g. packet loss) used current approaches. In particular, PPD uses knowledge of the characteristics of compressed video, i.e., the relative importance of different video frames. For example, three types of frame, I-, P- and B-frame exist in MPEG video. The I-frame is coded independently. The P-frame and B-frame are coded by using the closest past I- or P-frame, and the closest past and future I- or P-frames, respectively. Hence, I-frame is more important than P-frame, which in turn, is more important than B-frame. It also considers the correlation between video characteristics, network resource requirements and the resulting visual quality.

Furthermore, PPD classifies a video stream into different classes based on loss tolerance. It then applies queue-limits to these classes by calculating the "weight" of each class. The queue-limits determine the number of packets of each video that are allowed in the router buffer when the total buffer occupancy exceeds the predefined buffer length threshold. If the number of packets is larger than the queue limit, then the packet is dropped. The weight of each class matches the class criteria. This means that low loss tolerance traffic has priority over high loss tolerance traffic. Specifically, if two classes of videos are considered, where class r has lower packet loss tolerance than class s, the weight parameters to each are ω_r and ω_s, and the numbers of video within each are n_r and n_s, therefore a relationship will are defined as: $\omega_r n_r + \omega_s n_s = 1$ subject to $\omega_r > \omega_s$. Figure 1 shows the algorithm of PPD.

```
/*
E-frame:  a partially discarded frame.
Len: the buffer length.
Size: the buffer size.
LOW:  the buffer length threshold at which B-packets start being
dropped.
HIGH:  the buffer length threshold at which P-packets start being
dropped.
Wi: weight parameter for video stream i.
*/

if (packet == E-frame)
   drop();
if (Len == Size)
   drop();
else if (Len > HIGH ){
          if((packet == first P-packet ) || (packet == B-packet)){
               drop();
          }
          else
               accept();
}else if (Len > LOW){
          if ((packet == B-packet) || (Len > Wi*Size ))
               drop();
          else
          accept();
}else   accept();
```

Fig. 1. Algorithm of PPD

Inter-node loss allocation (InterLA), on the other hand, focuses on translating the end-to-end loss requirement of a video to a set of local nodal loss constraints, such that a video that meets every local loss constraint through the use of PPD also meets corresponding end-to-end loss requirements. InterLA first allocates equal shares of the end-to-end loss requirements of a video to all the nodes along the source-destination path. It then modifies downstream loss constraints based on a video's upstream loss performance for each video. In order to illustrate the InterLA mechanism, we use the following notations:

$PLR_j(i)$: acceptable packet loss rate of video j at node i
PLR_j : acceptable end-to-end packet loss rate of video j
Δ_j: difference between the actual $PLR_j(i)$ and the initial $PLR_j(i)$
$\Delta_j(i\text{-}k)$: accumulated Δ_j from node i to k.

One can easily see that the $\Delta_j(i\text{-}k)$ can take both positive and negative values. Positive values correspond to *worse loss performance* where video j has experienced excess loss when passing through node i to k, whereas negative values indicate *better loss performance* with packet loss of video j less than expected. InterLA evenly distributes the positive Δj(i-k) to the remaining lightly loaded nodes, while negative Δj(i-k) to the remaining heavily loaded nodes.

The IPD scheme is "active" for the two reasons: firstly, data content is taken into consideration. Specifically, the payload of a packet influences the network management computations that are to be performed on it. Also, a router can examine the packet payload and decide which action should be performed. Secondly, the implementation of the scheme is based on AN node architectures, which are discussed in detail in Section 4.

3 Simulation Results

The objective of the simulations is to study the structure of the computations and the performance of IPD. As a result of the router-based computations, the AN router selectively forwards video packets to the users, providing good video quality and efficient network utilization. In the simulations, MPEG-4 video traces are used [http://peach.eas.asu.edu/index.html]. To simulate an active node-based network, an active module is added to a router. An active buffer management algorithm is installed at the module prior to the start of video transfer. During the transmission video packets are passed to the active algorithm. Once processed, the packets are forwarded. This complements the non-active schemes as the router no longer just passively transport packets. The parameter settings are in the following: packet size<=1500 bytes, Output link capacity=100Mbps, B=150KB, HIGH=0.90, LOW=0.8, ω_r = 0.20 and ω_s =0.13. Here, a threshold of 0.90 means that buffer length threshold is 90% of the buffer size (in packets). The three performance metrics are Frame Error Rate (FER), Effective Throughput (ET) and Fairness Index (FI). The definitions are given in Table 1.

Table 1. Definition of Performance Metrics

FER	The fraction of frames in error for each video stream. If one packet in a frame is lost then this whole frame and its propagated frames (P and or B) are considered to be frames in error.
ET	The fraction of usable data over all the video streams. "Usable data" is the video data that belongs to a successfully delivered frame, i.e., a video frame in which all of the packets and reference frames (I and or P) are completely transmitted.
FI	The ratio of actual packet loss rate (PLR) over the acceptable PLR for each video stream. A value of one or less than one for a satisfactory loss performance and a value of greater than one for unsatisfactory loss performance.

The IPD scheme is compared with the Uncoordinated Buffer Management (UBS) scheme. In IPD, each individual node employs PPD scheme with incorporation between nodes through the use of InterLA. While In UBS, each individual node employs the Drop Tail (DT) scheme independently, without inter-node cooperation. In the DT, if a buffer is full, incoming packets are discarded. The simulation results for the transport of six videos over a six-node network, with a load pattern between nodes, following high, high, low, high, low, low are presented in Figures 2 to 4.

Videos 1-3 are sports sources called “Soccer” with a packet loss constraint of 6%, and videos 4-6 are news sources called “ARD Talk” with a packet loss constraint of 12%. The results presented show the final values of the average of different five runs, where the starting sequence of a video stream was randomly selected. In each run, the test videos were started randomly over a 60-second interval.

.

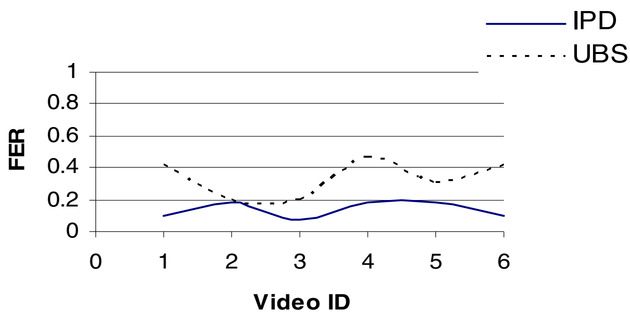


Fig. 2. Difference in FER

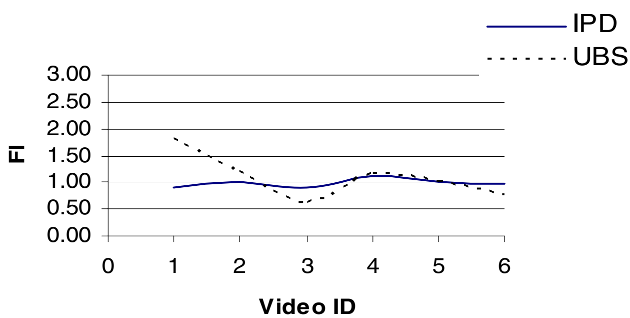


Fig. 3. Difference in ET

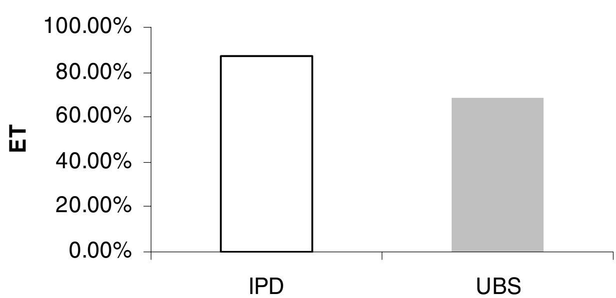


Fig. 4. Difference in FI

The load levels (L) of High and Low refers to 90% and 70%, respectively and is given by L= (N+m)×ρ. Here, N is the number of background flows, m is the number of test video sources, and ρ is the load contributed by each source. The ρ is calculated by $\rho=t_{on}/t_{off}$. t_{on} and t t_{off} are the mean duration of on and off periods, respectively. In the on period, the source generates packets at a variable rate specified by the packet inter-arrival time given in a video frame interval.

As seen in Figures 2 to 4, compared to the UBS, the advantages of the IPD are:

1) Lower FER: IPD normally admits packets belonging to completely correct video frames and discard packets from partially corrupted frames. Also, it performs preventative priority packet dropping. Thus, error propagation due to frame dependent nature is reduced, decreasing FER. On the other hand, the UBS treats all the packets in the same way, regardless of the packet type, and performs a random drop, most likely distributing packet loss, which translates into a large FER and low perceived quality of a video. This is because each lost packet may belong to a different frame. In particular, a lost packet could belong to an I or P-frame. Consequently, small packet loss will affect a large number of consecutive frames due to error propagation.
2) Higher ET: The IPDS reduces I and P-packet losses and improves the output of perceptually important information, thus increasing real network efficiency.
3) FI closer to one: This is a consequence of the mutual support of InterLA and PPD schemes. The InterLA scheme effectively controls the intra-node loss of each video and attempts to achieve their individual end-to-end loss requirements. Specifically, the loss constraints of a video can be adjusted based on the performance in preceding nodes. As a result, downstream nodes can help the video “catch up” if there are excessive data losses in upstream nodes, whereas videos receiving extra servicing can have their loss constraints reduced to allow more urgent videos to pass through. Meanwhile, the IPD scheme sets the weight in the buffer sharing mechanism according to the acceptable PLR. Videos with a higher acceptable PLR receive a lower share of buffer and vice versa. Therefore, when the IPD scheme is applied, less buffer space is allocated to streams 4-6 than streams 1-3. In turn, every stream achieves loss performance at a level commensurate with expectations. This results in an equitable loss distribution among the video streams with different loss tolerances. Conversely, the UBS scheme does not have a mechanism to adjust buffer allocation in equilibrium with different loss tolerances. Thus, losses are arbitrarily distributed among video streams. For example, the streams with same packet loss tolerance, i.e., streams #1 and #3, exhibit significantly different FI, meaning that unfair service is provided to both streams.

4 Active Network-Based Node Architecture

The active schemes have been shown to be effective in improving QoS for video traffic. To support these schemes, new node architecture is proposed (Figure 4). The new architecture exhibits three key components: 1) three types of cache memories, e.g., conventional, AN-stream, and Bulk-data caches; 2) layered data processing elements in CPU; and 3) the filter in network interface.

First of all, there are three types of cache memories in the node architecture: a conventional cache (to hold programs, state information and other information that is reused or updated), a stream cache (AN-stream cache, to process AN streams) and a data cache (Bulk-data cache, to store bulk data). The design decisions were made based on the following observations: 1) AN streams require some processing (or computing) rather than just simple forwarding as in bulk data transmission; 2) streaming data usually is touched only a relatively few times, but require timely and speedy access. Hence, the processing of AN streams can be sped up by putting AN streams directly into the AN-stream cache instead of the Bulk-data cache. That means it should be possible to store and process AN stream and bulk data separately. The CPU can select a cache memory according to the types of traffic data. This selective storage system allows quicker access of the necessary data and simpler cache system design. Thus, more efficient data movement through the node can be achieved.

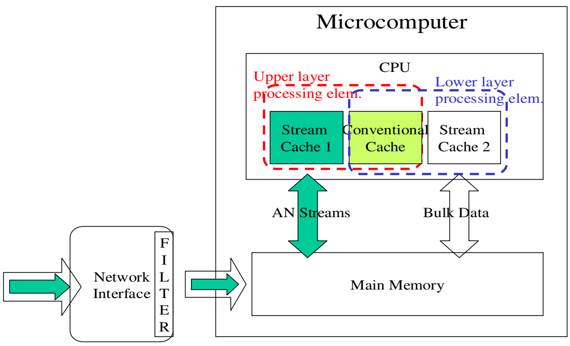


Fig. 5. An Active Network-based node architecture

Next, layered data processing elements in the CPU are used to isolate and speed up various operations from one another. Simple packet forwarding is only carried out in the upper layer while complex packet processing is implemented in the lower layer. Bulk data (for the forwarding operation) are processed in the upper layer without interfering with the lower layer, leading to a fast delivery of bulk data. In the delivery of AN streams, examination of the header of each packet and possibly a relative complex computation on its payload have to be performed. For example, IPD involves: a) computing the packet loss ratio and the weight; and b) selectively discarding the packets based on the packet content. Considering that the computation for the proposed active scheme is not very complicated, no special ALU operations are needed. Specifically, the IPD is associated with addition, multiplication, and division operations only. In sum, the layered processing structure allows handling different types of data in parallel. A high throughput for bulk data and QoS-aware transfer for stream data is provided and the overall delay at a network node is reduced.

Finally, the network filter is placed inside the network interface, where all the data coming from the network is divided into bulk data and stream data for delivery to the appropriate cache. Overall, the three components in the proposed node architecture make the network node work more efficiently and effectively when dealing with the combination of AN stream and bulk data.

5 Conclusions

In this paper, an active packet discard scheme is proposed. Simulation experiments using actual MPEG video traces have been carried out to test the performance of the proposed scheme. The experiments show that it not only significantly increases the viewing quality of per video streams, but also improves network efficiency. The scheme also provides a superior level of fairness of service among competing video streams. A possible node architecture has been proposed but an in depth study of that architecture has not yet been completed. Presently, we are simulating the prototype nodes and emulate the proposed scheme into the node in order to validate the proposed node architecture.

References

1. D.L. Tennenhouse, J.M. Smith, W.D. Sincoskie and D. J. Wetherall, A Survey of Active Network Research, *IEEE Communication Magazine*, Vol. 35, No. 1, pp.80-86, January 1997.
2. K. Psounis, Active networks: Applications, Security, Safety, and Architectures, *IEEE Communications Surveys*, 2(1), Q1 1999.
3. D. Clark and W. Fang, "Explicit Allocation of Best-Effort Packet Delivery Service", *IEEE/ACM Transactions on Networking*, pp.362-373, Vol.6, No.4, August 1998.
4. S. Floyd and V. Jacobson, Random Early Detection Gateways for Congestion Avoidance, *IEEE/ACM Transactions on Networking*, pp.397-413, August 1993.
5. Y. G. Kim, J. Kim, and C. Kuo, "TCP-Friendly Internet Video with Smooth and Fast Rate Adaptation and Network-Aware Error Control", *IEEE Transactions on Circuit and Systems for Video Technology*, Vol.14, Issue 2, pp. 256 – 268, February 2004.
6. Smitha and A. L. N. Reddy, "LRU-RED: An Active Queue Management Scheme to Contain High Bandwidth Flows at a Congested Router", *IEEE Globecom'01*, San Antonio, USA, November 2001.
7. J.M. Smith and S.M. Nettles, "Active Networking: One View of the Past, Present, and Future", *IEEE Transactions on Systems, Man and Cybernetics, Part C*, pp. 4-18, February 2004.
8. R. Mahajan, S. Floyd and D. Wetherall, "Controlling High-Bandwidth Flows at Congested Router", *the 9th IEEE International Conference on Network Protocols*, Riverside, USA, November 2001.

Quality Adapted Backlight Scaling (QABS) for Video Streaming to Mobile Handheld Devices

Liang Cheng[1,*], Stefano Bossi[2], Shivajit Mohapatra[1], Magda El Zarki[1], Nalini Venkatasubramanian[1], and Nikil Dutt[1]

[1] Donald Bren School of Information and Computer Science, University of California, Irvine, CA 92697, USA
{lcheng61, mopy, magda, nalini, dutt}@ics.uci.edu
[2] stboss@tin.it

Abstract. For a typical portable handheld device, the backlight accounts for a significant percentage of the total energy consumption (e.g., around 30% for a Compaq iPAQ 3650). Substantial energy savings can be achieved by dynamically adapting backlight intensity levels on such low-power portable devices. In this paper, we analyze the characteristics of video streaming services and propose an adaptive scheme called Quality Adapted Backlight Scaling (QABS), to achieve backlight energy savings for video playback applications on handheld devices. Specifically, we present a fast algorithm to optimize backlight dimming while keeping the degradation in image quality to a minimum so that the overall service quality is close to a specified threshold. Additionally, we propose two effective techniques to prevent frequent backlight switching, which negatively affects user perception of video. Our initial experimental results indicate that the energy used for backlight is significantly reduced, while the desired quality is satisfied. The proposed algorithms can be realized in real time.

1 Introduction

With the widespread availability of 3G cellular networks, mobile hand-held devices are increasingly being designed to support streaming video content. These devices have stringent power constraints because they use batteries with finite lifetime. On the other hand, multimedia services are known to be very resource intensive and tend to exhaust battery resources quickly. Therefore, conserving power to prolong battery life is an important research problem that needs to be addressed, specifically for video streaming applications on mobile handheld devices.

Most hand-held devices are equipped with a TFT (Thin-Film Transistor) LCD (Liquid Crystal Display). For these devices, the display unit is driven by

* This research was in part funded by a gift from Conexant, Newport Beach, CA through the auspices of the Center for Pervasive Communications and Computing (CPCC) at UC, Irvine.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 662–671, 2005.

the illumination of backlight. The backlight consumes a considerable percentage of the total energy usage of the handheld device; it consumes 20%-40% of the total system power (for Compaq iPAQ) [1].

Dynamically dimming the backlight is considered an effective method to save energy [1, 2, 3] with scaling up of the pixel luminance to compensate for the reduced fidelity. The luminance scaling, however, tends to saturate the bright part of the picture, thereby affecting the fidelity of the video quality.

In [2], a dynamic backlight luminance scaling (DLS) scheme is proposed. Based on different scenarios, three compensation strategies are discussed, i.e., brightness compensation, image enhancement, and context processing. However, their calculation of the distortion does not consider the fact that the clipped pixel values do not contribute equally to the quality distortion. In [3], a similar method, named concurrent brightness and contrast scaling (CBCS), is proposed. CBCS aims at conserving power by reducing the backlight illumination while retaining the image fidelity through preservation of the image contrast. Their distortion definition and proposed compensation technique may be good for static image based applications, such as the graphic user interface (GUI) and maps, but might not be suitable for streaming video scenarios, because their contrast compensation further compromises the fidelity of the images. In addition, Neither [2] nor [3] solves the problem associated with frequent backlight switching which can be quite distracting to the end user.

In this paper, we explicitly incorporate video quality into the backlight switching strategy and propose a quality adaptive backlight scaling (QABS) scheme. The backlight dimming affects the brightness of the video. Therefore, we only consider the luminance compensation such that the lost brightness can be restored. The luminance compensation, however, inevitably results in quality distortion. For the video streaming application, the quality is normally defined as the resemblance between the original and processed video. Hence, for the sake of simplicity and without loss of generality, we define the quality distortion function as the mean square error (MSE)(see Equation (1)) and the quality function as the peak signal to noise ratio (PSNR)(see Equation (2)), both of which are well accepted objective video quality measurements.

$$MSE = \frac{1}{M} \times \sum_{i=1}^{M} (x_i - y_i)^2 \tag{1}$$

$$PSNR(dB) = 10log_{10} \sum_{i=1}^{M} \frac{255^2}{(x_i - y_i)^2} \tag{2}$$

where x_i and y_i are the original pixel value and the reconstructed pixel value, respectively. M is the number of pixels per frame.

It is to be noted that any improved quality metrics may be adopted to replace the MSE/PSNR metrics used here without affecting the validity of our proposed scheme.

As is mentioned in [3], for video applications, the continuous change in the backlight factor will introduce inter-frame brightness distortion to the observer.

In our experiments, we find that the "unnecessary" backlight changes fall into two categories: (1) small continuous changes over adjacent frames; (2) abrupt huge changes over a short period. Therefore, we propose to quantize the calculated backlight to eliminate the small continuous change and use a low-pass digital filter to smooth the abrupt changes.

The rest of the paper is organized as follows. In Section 2, we introduce the principle of the LCD display - experimental results show that backlight dimming saves energy while the pixel luminance compensation results in minimal overhead. In Section 3, we present our QABS scheme, which includes determining the backlight dimming factor and two supplementary methods to avoid excessive backlight switching. Section 4 shows our prototype implementation, experimental methodology and simulation results. We conclude our work in Section 5.

2 Characteristics of LCD

In this section, we outline the characteristics of the LCD unit from two perspectives, the LCD display mechanism and the LCD power consumption, both of which form the basis for our system design.

2.1 LCD Display

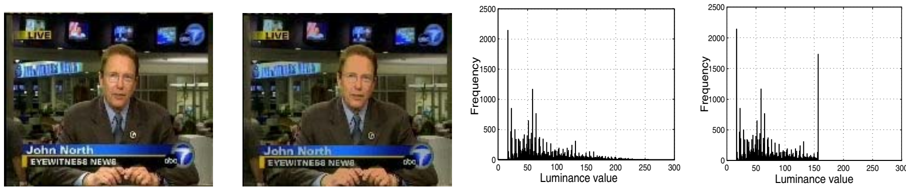
The LCD panel does not illuminate itself, but displays by filtering the light source from the back of the LCD panel [2][3]. There are three kinds of TFT LCD panels: transmissive LCD, reflective LCD, and transflective LCD. We focus in this paper on the reflective, since it is the most commonly used LCD for handheld devices. Henceforth, when we mention LCD, we refer to reflective LCD and we refer to both backlight and forelight as backlight. As will be shown, our idea is generic to any backlight based LCD.

The perceptual luminance intensity of the LCD display is determined by two components: backlight brightness and the pixel luminance. The pixel luminance can be adjusted by controlling the light passing through the TFT array substrate. Users may detect a change in the display luminance intensity if either of these two components is adjusted. That is, the backlight brightness and the pixel luminance can compensate each other. In Section 2.2, we will show that the pixel luminance does not have a noticeable impact on the energy consumption, whereas the backlight illumination results in high energy consumption. Hence, in general, dimming backlight level while compensating the pixel luminance is an effective way to conserve battery power in hand-held devices.

Let the backlight brightness level and the pixel luminance value be L and Y, respectively, and the perceived display luminance intensity I. We may denote I using Equation (3).

$$I = \rho \times L \times Y \tag{3}$$

where ρ is a constant ratio, denoting the transmittance attribute of the LCD panel, and as such $\rho \times Y$ is the transmittance of the pixel luminance.



(a) Original image (b) Compensated image (c) Histogram before clipping (d) Histogram after clipping

Fig. 1. Image and its luminance histogram before and after clipping

We may reduce the backlight level to L' by multiplying L with a dimming factor α, i.e., $L' = L \times \alpha$, $0 < \alpha < 1$. To maintain the overall display luminance I invariable, we need to boost the luminance of the pixel to Y'. Since the pixel luminance value is normally restricted by the number of bits that represent it (denoted as n), Y' may be clipped if the original value of Y is too high or the α is too low. The compensation of the backlight is described in Equation (4).

$$Y' = \begin{cases} Y/\alpha, & if\ Y < \alpha \times 2^n \\ 2^n, & if\ Y \geq (\alpha \times 2^n) \end{cases} \tag{4}$$

Combining Equation (4) and Equation (3), we have

$$I' = \begin{cases} I, & if\ Y < \alpha \times 2^n \\ \rho \times L \times \alpha \times 2^n & if\ Y \geq (\alpha \times 2^n) \end{cases} \tag{5}$$

Equation (5) clearly shows that the perceived display intensity may not be fully recovered, instead, it is clipped to $\rho \times L \times \alpha \times 2^n$ if $Y \geq (\alpha \times 2^n)$. In Figure 2, we illustrate the clipping effect of the display luminance.

In Figure 1-a and Figure 1-c, we show an image and its luminance histogram. This image is the first frame of a typical news video clip ("ABC eye witness news") captured from broadcasting TV signal. Figure 1-b and Figure 1-d illustrate the image and its luminance histogram after backlight dimming and pixel luminance compensation. Figure 1-d shows that the pixels with luminance higher than 156 are all clipped to 156. This clipping effect eliminates the variety in the bright areas, which is subjectively perceived as the luminance saturation and is objectively assessed as 30dB with reference to the original image shown in Figure 1-a.

2.2 LCD Power Model

In our experiments, we observe that the backlight dimming can save energy whereas the compensation process, i.e., scaling up the luminance of the pixel,

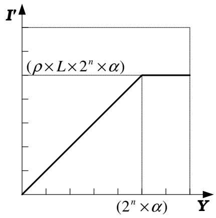


Fig. 2. Clipping

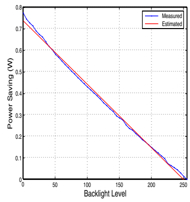


Fig. 3. Power saving vs. backlight level

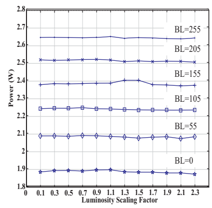


Fig. 4. Energy overhead

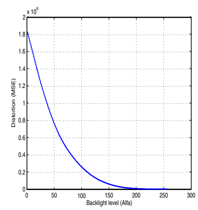


Fig. 5. MSE with different $Alfa$

has a negligible energy overhead. We measure the energy saving as a difference of the total system power consumption with backlight set to different levels from that with the backlight turned to the maximum (brightest). Figure 3 shows the plot between the various backlight levels and their corresponding energy consumption for a Compaq iPAQ 3650 running Linux. A more detailed setup of our experiments is described in Section 4. It is noticed that the backlight energy saving is almost linear to the backlight level and can be estimated using Equation (6).

$$y = a1 \times x + a2 \tag{6}$$

where y is the energy savings in Watt; x denotes the backlight level; a1 and a2 are coefficients. We apply the curve fitting function of MATLAB and obtain $a1 = -0.0029567$ and $a2 = 0.73757$ with the largest residual fitting error as 0.085731.

Contrary to the backlight switching, the pixel luminance scaling is uncorrelated to the energy consumption. In Figure 4, we show that for one specified backlight level (BL) the system energy consumption basically remains stable and is independent of the luminance scaling.

Figure 3 and Figure 4 justify the validity of the generic backlight power conservation approach, i.e., dimming the backlight while enhancing the pixel luminance value. Note that in Figure 4, "BL" refers to the backlight level and "Luminosity Scaling Factor" refers to α. In the next section, we apply this method to the video streaming scenario, discussing a practical scheme to optimize the backlight dimming while taking into consideration the effect on video distortion.

3 Adaptive Backlight Scaling

As explained in Equation (5), the backlight scaling with the luminance compensation may result in quality distortion. The amount of backlight dimming, therefore, has to be restricted such that the video fidelity will not be seriously affected.

3.1 Optimized Backlight Dimming

We define the optimized backlight dimming factor as the one whose induced distortion is closest to a specified threshold. Henceforth, we replace the factor α with the real backlight level $Alfa$, $Alfa = N \times \alpha$ (N is the number of backlight levels (256 for Linux on iPAQ)), and the optimized backlight dimming is represented as $Alfa^*$.

In Figure 5, we illustrate the image quality distortion in terms of MSE over different backlight levels. (Note that we use the image shown in Figure 1-a.) We see that as $Alfa$ increases, the induced video quality distortion due to the brightness saturation monotonously decreases. Hence, for a given distortion threshold, we can find a unique $Alfa(= Alfa^*)$ for each image. In video applications, for a given distortion, different frames may have distinct $Alfa^*$, depending on the luminance histogram of that frame. However, it is hard to have an accurate analytical representation of the quality distortion using $Alfa$ as a parameter. We therefore adopt an optimized search based approach, where we calculate the MSE distortion with different $Alfa$ until the specified distortion threshold is met. The results of our scheme are accurate and can be used as the benchmark for the design of other analytical methods.

Figure 6 shows the exhaustive searching algorithm for finding $Alfa^*$ for one image. $FindAlfa(th)$ takes the distortion threshold (th) as input, and returns the $Alfa^*$ as output. Note that $MSE(Alfa)$ calculates the MSE with the specified $Alfa$ for one frame.

However, the complexity of an exhaustive search shown in Figure 6 is too high. As shown in Equation (2), the per-frame MSE calculation consists of M multiplications and $2M$ additions. M is the number of pixels in one frame, e.g., $M = 25344$ for QCIF format video. We regard the per-frame MSE as the basic complexity measurement unit. We assume that the optimized backlight level is uniformly distributed in [0, N], and thus the complexity of algorithm in Figure 6 is $O(N)$. In our test, $N = 256$. Obviously, the optimized backlight dimming factor can hardly be calculated in real-time.

Therefore, we apply a faster bisection method [4] to improve the algorithm for finding $Alfa^*$. Since we can easily find an upper bound (denoted as u) and a lower bound (denoted as d) on the backlight levels, we get as good an approximation as we want by using bisection. We assume that $u > d$ and let ϵ be the desired precision and present the algorithm in Figure 7.

By using the bisection method, we may achieve the complexity of $O(log_2 N)$ in the worst case. For instance, for $N = 256$ and $\epsilon = 1$, we only need to calculate per-frame MSE at most eight times, which is fast enough for real-time processing.

3.2 Smoothing the Backlight Switching

It has been discussed in [3] that the backlight dimming factor may change significantly across consecutive frames for most video applications. The frequent switching of the backlight may introduce an inter-frame brightness distortion to the observer. Hence, it is necessary to reduce frequent backlight switching.

```
Proc: FindAlfa(th)
  Alfa := 0;
  while Alfa ≤ N do
    if MSE(Alfa) > th then
      Alfa := Alfa + 1;
    else
      return(Alfa);
    end if
  end while
```

Fig. 6. Exhaustive algorithm for finding Alfa*

```
Proc: FastFindAlfa(th, ϵ)
  u := upper bound;
  d := lower bound;
  while (u - d) > ϵ) do
    Alfa = round((d + u)/2);
    if (MSE(Alfa) > th) then
      u = Alfa;
    else
      d = Alfa;
    end if
  end while
  return(Alfa);
```

Fig. 7. Fast algorithm for finding Alfa*

In our study, we observe that the calculated $Alfa^*$, although based on an individual image, does not experience huge fluctuations during a video scene, i.e., a group of frames that are characterized with similar content. Actually, the redundancy among adjacent frames constitutes the major difference between the video and the static image application and has long been utilized to achieve higher compression efficiency. Hence, the backlight switching should be smoothed out within the scene and most favorably only happen at the boundary of video scenes.

We propose two supplementary methods to smooth the acquired $Alfa^*$ in the same video scene. First, we apply a low-pass digital filter to eliminate any abrupt backlight switching that is caused by the unexpected sharp luminance change. The passband frequency is determined by the subjective perception of the "flicker moment" and the frame display rate. Second, we propose to quantize the number of backlight levels, i.e., any backlight level between two quantization values can be quantized to the closest level, by which we prevent the needless backlight switching for small luminance fluctuations during one scene. In our experiments, we quantize all 256 levels to "N"levels (N=5 in our study). We switch the backlight level only if the calculated $Alfa^*$ changes drastically enough, so that it falls into another quantized level.

4 Performance Evaluation

In this section, we introduce our prototype implementation, the methodology of our measurement and the performance of the proposed algorithm.

4.1 Prototype Implementation

Figure 8 shows a high level representation of our prototype system. Our implementation of the video streaming system consists of a video server, a proxy server and a mobile client. We assume that all communication between the server and the mobile client is routed through a proxy server typically located in proximity to the client.

The video server is responsible for streaming compressed video to the client; The proxy server transcodes the received stream, adds the appropriate control information, and relays the newly formed stream to the mobile client (Compaq iPAQ 3650 in our case). For the sake of simplicity and without loss of generality, in our initial prototype implementation, we use the proxy server to also double up as our video server.

The proxy server includes four primary components - the video transcoder, the proposed QABS module, the signal multiplexer, and the communication manager. The transcoder uncompresses the original video stream and provides the pixel luminance information to the QABS module. The QABS module calculates the optimized backlight dimming factor based on the user quality preference feedback received from the client (user). The multiplexer is used to multiplex the optimized backlight dimming information with the video stream. The communication manager is used to send this aggregated stream to the client.

On the mobile client, the demultiplexer is used to recover the original video stream and the encoded backlight information from the received stream. The LCD control module renders the decoded image onto the LCD display. The backlight information is fed to the "Backlight Adjustment Module", which concurrently sets the backlight value for the LCD. In particular, users may send the quality request to the proxy when requesting for the video, based on his/her quality preference as well as concern for battery consumption.

4.2 Measurement Methodology

For video quality and power measurements, we use the setup shown in Figure 9. The proxy in our experiments is a Linux desktop with a 1GHz processor and 512MB of RAM. All our measurements are made on a Compaq iPAQ 3650. We use a National Instruments PCI DAQ board to sample voltage drops across a resistor and the iPAQ, and sample the voltage at 200K samples/sec. We calculate the instantaneous and average power consumption of the iPAQ using the formula $P_{iPAQ} = \frac{V_R}{R} \times V_{iPAQ}$.

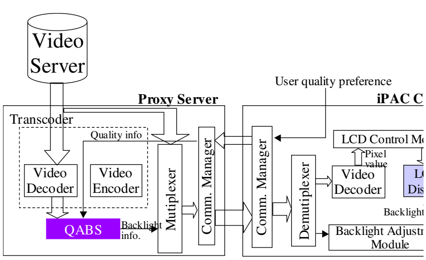


Fig. 8. Prototype implementation

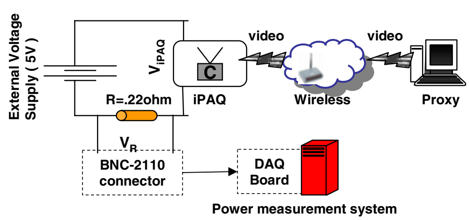


Fig. 9. Setup for our measurements

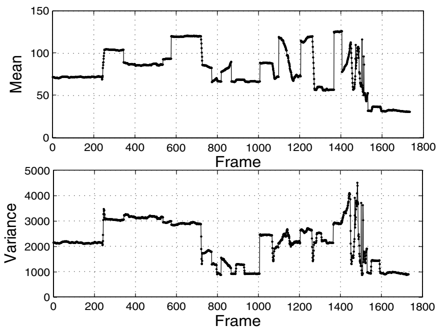


Fig. 10. Basic statistics of *abc_news*

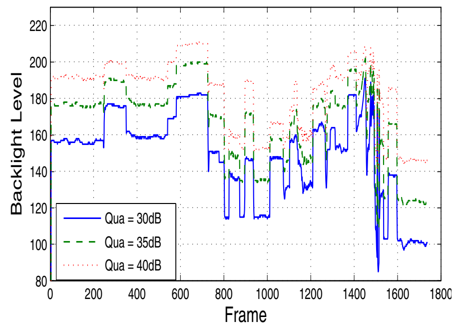


Fig. 11. $Alfa^*$ adapted to three given quality thresholds

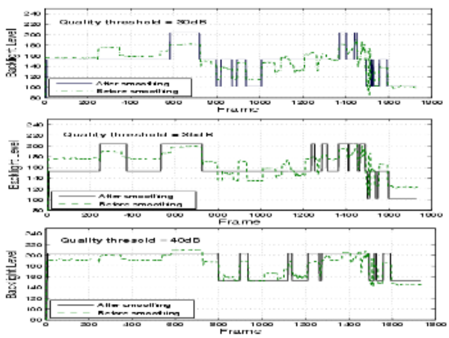


Fig. 12. $Alfa^*$ before and after filtering and quantization

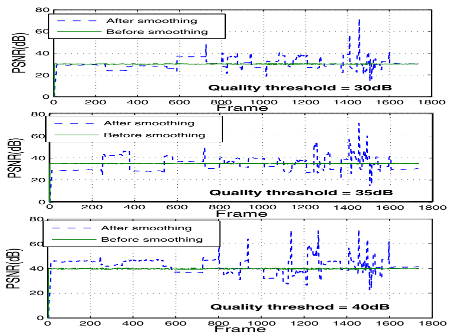


Fig. 13. Quality before and after Alfa smoothing

4.3 Experimental Results

In our simulation, we use a video sequence captured from a broadcasted *ABC_news* program, whose first frame is shown in Figure 1-a. We choose this video as representative of a typical usage of a PDA. In Figure 10, we show the basic statistics (i.e., the mean and the variance of luminance per frame) of this video.

We assume that the users are given three quality options, fair, good, and excellent, which respectively correspond to the PSNR value of 30dB, 35dB, and 40dB. After applying the algorithm "Proc: FastFindAlfa", we obtain the adapted $Alfa^*$ for these three quality preferences, as is shown in Figure 11. It can be seen that higher video quality needs higher backlight level on average.

In Figure 12, we show $Alfa^*$ before and after the backlight smoothing process. It is seen that the small variation and the abrupt change of the backlight switching are significantly eliminated after the filtering and quantization. In addition, as we expected, the backlight switching mostly happens at the boundary of major scenes.

In Table 1, we summarize the results of our QABS. The mean $Alfa^*$ of different quality preferences produces a quality on average very close to the

Table 1. Results of QABS

Alfa Mean			Quality(dB)			Power Saving(%)		
Fair	Good	Excellent	Fair	Good	Excellent	Fair	Good	Excellent
149	162	186	30.17	34.28	42.31	41.8%	36.7%	27.3%

pre-determined quality threshold. It is noted that different quality requirements result in various power saving gains. Higher quality preference must be traded using more backlight energy. Nevertheless, we can still save 29% energy that is supposed to be consumed by the backlight unit if we set the quality preference to be "Excellent".

In Figure 13, we show that the filtering and quantization process may lead to instantaneous quality fluctuation, which is contrasted to the consistent quality before backlight smoothing. Nevertheless, we observe that the quality fluctuation is around the designated quality threshold and mostly happens at scene changes.

5 Conclusion

In this paper, we apply a backlight scaling technique to video streaming applications, and explicitly associate backlight switching to the perceptual video quality in terms of PSNR. The proposed adaptive algorithm is fast and effective for reducing the energy consumption while maintaining the designated video quality. To reduce the frequency of backlight switching, we propose two supplementary schemes that smooth the backlight switch process such that the user perception of the video stream can be substantially improved.

Acknowledgement

We would like to thank Michael Philpott, who helped us with the experiment setup and the power measurements.

References

1. S. Pasricha, M. Luthra, S. Mohapatra, N. Dutt, N. Venkatasubramanian, "Dynamic Backlight Adaptation for Low Power Handheld Devices," *To appear in IEEE Design and Test (IEEE D&T), Special Issue on Embedded Systems for Real Time Embedded Systems,* Sep. 2004.
2. N. Chang, I. Choi, and H. Shim, "DLS: Dynamic Backlight Luminance Scaling of Liquid Crystal Display," *IEEE Transaction on VLSI System,* vol. 1, Aug. 2004.
3. W.-C. Cheng, Y. Hou, and M. Pedram, "Power Minimization in a Backlit TFT-LCD Display by Concurrent Brightness and Contrast Scaling," *Proceedings of the Design,* Automation and Test in Europe, Feb. 2004.
4. J. L. Zachary, *Introduction to Scientific Programming: Computational Problem Solving Using Maple and C.* Telos Publishers, 1996.

Video Flow Adaptation for Light Clients on an Active Network

David Fuin, Eric Garcia, and Hervé Guyennet

LIFC, Laboratoire d'Informatique de l'université de Franche-Comté,
16, route de GRAY, 25030 BESANÇON CEDEX, France,
tel: (33) 3 81 66 20 92, fax: (33) 3 81 66 64 50
{fuin, garcia, guyennet}@lifc.univ-fcomte.fr

Abstract. Hierarchical video allows to send different qualities of flow to clients from a single video file located on a server. Light clients (such as PDA) can't display this kind of video because of the lake of computing resources to decode (aggregate layers) the video in real-time. That's why we propose to distribute this aggregation on active nodes located along the video flow path between the video server and the light client (each active router decides independently to process packets). To achieve this the best way, we make our protocol respectful of others traffics in guaranteeing them a minimum part of active node resources. In order to guarantee some QoS level, active nodes check their resources and their interface congestion. Thus, in case of overloaded active nodes, they contact the video server to decrease the quality of the video to ease the network. This allows a hierarchical video to be displayed on a PDA with the best quality without disturbing others traffics.

1 Introduction

The use of hierarchical codec [1] for Video on Demand (VoD) allows to provide several different qualities from a unique video file to client. This avoids the storage of multiple files for the same video at different qualities[2]. Thus, the use of such codec allows to have more choice of qualities with a storage of the same size as a video using a non-hierarchical codec. Displaying this kind of video requires aggregation of its layers, i.e. the fusion of the various information contained in the different layers in order to restore the original picture.

However, it is difficult to display a hierarchical video flow made of several layers on a light client (such as PDA or mobile phone). Indeed, most of the time, PDAs do not have neither enough computing resources to decode (layers aggregation) the video in real time nor have the adequate codec.

Thus, a solution is to adapt on the fly the data coming from the video server in a format that the light client can use. Three kinds of solutions are possible: to adapt video on the server, to use a transcoding proxy or to use network resources to carry out adaptation.

The first two solutions use a centralized approach, then, this causes troubles when the number of clients increases. Thus, it is preferable to use a distributed approach in order to better support scalability. Thanks to Active Network[3], network (more particularly Active Nodes) can carry out computation on flows which cross them.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 672–679, 2005.

This idea is to distribute aggregation and transcoding of flows on network nodes in order to reduce computation resources needed on light clients. Active node carry out video layers aggregation.

Each active node monitor its computing resources as well as congestion of their network interfaces to be able to warn the video server if necessary. Then, this one decides to reduce the number of layers that it emits. This decreases the bandwidth and the computing resources of active nodes used to aggregate the video layers. This behavior allows us to keep semantic information while decreasing network load.

In section 2, we present in details how our protocol works as well as how we include QoS management. The next section shows the results we obtain from a simulator we have developed to validate our work. We conclude in the last section and present our future work.

2 Working of Our Protocol

A user with a light client (such as PDA) wants to watch a particular video sequence available on a distant video server (see figure 1). This server proposes this video in several qualities thanks to hierarchical encoding. The client and the video server are separated by `n` active nodes.

This client emits a request towards the video server. This one sends back the video in a hierarchical format corresponding to the request. Layers aggregation is carried out along the network, and in case of congestion, the filter (ordered by a congested router) decreases emission of the video. We will see in detail these operations.

2.1 The Request for a Particular Video Sequence

The client emits a request towards the video server. This request specifies, in addition to the wished video, the desired quality and parameters such as the CPU speed, supported codec... While crossing network, this request marks the first router met (`Rn` on the figure 1) as being the last router who will be crossed by video flow.

Indeed, if packets arrive on the last router and were not already aggregated, that means that all active nodes are overloaded. Therefore, we are able to take specific decisions in order to solve this problem.

The request also examines the static bandwidth of links between each router and deduces the maximum usable bandwidth. This information is communicated to the video server so that this one knows the maximum number of layers that it can send to this client (depending of available bandwidth).

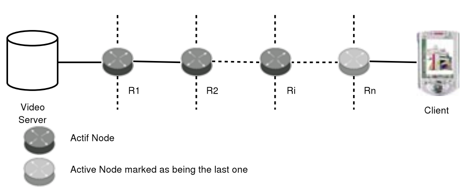


Fig. 1. Network architecture

2.2 Sending of Data and Layers Aggregation Distribution

Aggregation Distribution. With the reception of the video request from the client, the server can begin to send the video.

The video server orders the filter so that this one selects a given number of layers to be sent according to the bandwidth measured previously and to client request.

Until now, working is close to a classical video on demand application on a non-active network. Indeed, on a classical network, it is the filter that selects the layers to be sent. Then, the client must aggregate flows in order to be able to display the video. However, we know that it is not always possible with a light client (problems of CPU resources). Thus, it is from this moment that the working of active version of the application differs from the non-active one. We propose that active routers carry out aggregation and transcoding of flows so that the client receives the video in a format that it can display.

However, we cannot aggregate all flows on the first active router. This one would be then overloaded as soon as a too many light clients would be connected to the video server. Then, it would be unable to manage all flows, but moreover, its overload would also cause the collapse of the routing speed of other packets.

Moreover, it is strongly probable that the first router after the video server (`R1`) is common to all flows even if the recipients are different. It is for these reasons that we planned to distribute layers aggregation on all active routers located between the client and the server. Thus, we propose that the first active node met (`R1`) undertakes this task if it has enough processing resources. If it is not the case, the second active router (`R2`) try to undertake it, if it can not, the packet will be aggregate on one of the following (`Ri`). Until the arrival on the last router (`Rn`) which represents the worst case. Indeed, that means that all the routers along video flow are overloaded. We will see further which behavior we adopt in order to solve this problem.

Thus, active packets transporting flows are marked "non-aggregated" at their emission, then marked "aggregated" after layers aggregation on a router (so that others routers do not try to carry out them once again).

A router do not carry out the whole video flow what would bring us back to the problems of a centralized treatment seen previously. Routers share work, that is possible because the treatment of each picture (layers aggregation of a picture) is independent of others pictures thanks to the use of an appropriate codec.

Each active router decides independently (of others packets and of others active node) to process packets. Thus, the first packet could be aggregated on `R4`, following packet on `R2`, following one on `Rn`... According to routers available resources.

Then, packets marked as "aggregated" are simply sent from router to router towards their destination without change.

Problems of the Number of Flows of a Hierarchical Video. Current solutions of video on demand (i.e. those that do not use active networks) using hierarchically encoded video such as [4], send video through several flows: each layer is sent in a different flow. This causes a problem. Let's take as example a video using three layers. It is possible that the all packets do not use the same path (see figure 2). Packets cannot be aggregated anywhere anymore (in particular on `Ri1` and `Ri2` in this example).

On the one hand, pictures can be processed independently from each other; on the other hand, all components of a particular picture must be available at the same time on

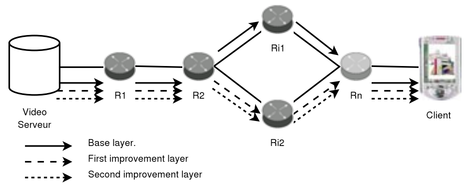


Fig. 2. Hierarchical video on non-active network

the same router in order to be able to aggregate them. But, if packets of a picture are distributed among three flows, how to make available packets corresponding to the same picture at the same time on a router. It would be necessary to be able to synchronize the packets of the same picture with a system of queues for example. This would add an extra delay to the video delivery.

There are several reasons to use one flow per layer. For example, that makes possible the use of a kind of "advanced" multicast where the various clients can receive a different number of layers. Thus, they can receive the same video but with qualities adapted to their connections and their needs. This is made possible thanks to routers having the possibility of filtering flows (and thus specific layers) what has for effect to diffuse the same video but with different qualities.

Another reason is the possibility of using mechanisms of quality of service such as DiffServ with different priorities on flows. Thus, packets from basic layer can be set to a high priority (preventing them to be dropped). A lower priority is given to the next layer. The probability of destroying packets of layer is all the more stronger as this layer contains only details. The transmission of the video with a minimal quality (basic layer) is ensured and, if the network can handle it, a better quality (with the various additional layers) is sent.

These two examples show the main interests to have a flow per layer in traditional networks. Within the framework of active networks, it is possible at an active node to change the content (payload) of a packet. Consequently, if the data corresponding to the three layers of a picture of our preceding example are embedded in one active packet, we keep the same flexibility as if using a flow by layer in the traditional networks.

Indeed, concerning multicast diffusion, active nodes are able to remove some layers from packets towards a client and to keep it intact towards another one. This mechanism is also possible in the case of quality of service where active nodes reduce the number of layers from packets rather than eliminate completely some packets in case of congestion (preserving most semantics information). Thus, we can affirm that, within the framework of active networks, we can use one flow for all layers while keeping the same advantages as "a flow per layers" version in non-active networks.

Thus, we propose that the payload of an active packet contains all information relating to one picture. This makes packets aggregation completely independent of others. Then, each packet contains all the layers of the corresponding picture: when any router receives any active packet, this router is able to aggregate all picture layers.

Case of Packets Using Different Path.

In active networks, it is not a problem. Indeed, on the one hand we saw that packets aggregation is independent, on the other hand, thanks to active network, our protocol is deployed where it is necessary.

2.3 Quality of Service Management

Our protocol takes care of the importance of resources management in order to avoid disturbing the behavior of the network.

Processing Resources. If the last router (`Rn` on the figure 1) receives packets marked as "non-aggregated", then, we can conclude that the part of the network from the video server to `Rn` does not succeed to aggregate all packets. This part of the network is thus overloaded.

With the reception of a "non-aggregated" packet, if `Rn` has enough computing resources, it aggregates the packet. If not, it only restores the basic layer (in order to limit computations on this packet to the bare minimum). The video then consist of pictures in desired quality (treated on the previous routers) and of pictures in low quality (which arrived on `Rn` "non-aggregated"). But in all the cases, the hierarchical codec is not useful on the light client. If this critical situation persists, then `Rn` contacts the filter telling him that it must reduce the number of layers sent (it is useless to send a layer that will be dropped before the restitution of the video). That causes to reduce the quality of the video, to reduce the bandwidth consumed by the flow and to decrease computing resources useful to aggregate the video.

Thus, we ensure that our protocol will not decrease network performances if the required computing resources are too significant for the various routers of the network. If we do not control computing resources, our protocol can block the traffic of other information going through this router. Consequently, when a packet arrives on a router, we analyze available resources to make sure that if the packet is processed locally then, it remains a minimum of resources available for packets corresponding to others traffics.

Bandwidth. If a network queue of a router saturates, this router contacts the filter in order to remove a layer from the video flow causing the same behavior as previously. As the request of "decreasing the number of layer" from the router go through the network to the server, this request configure crossed routers to also remove at their level the now-useless layer of packets already sent.

Rather than using RTCP in order to measure packets loss, we prefer to supervise the filling of routers queues. Where RTCP only makes it possible to detect the effective loss of packets and announces this loss to the server, we prefer to measure on each router the filling of the queues and to trigger, when reaching a given limit, a request to the server (more particularly to the filter) to decrease emission. Thus, this process enables us to detect a bandwidth near to saturation (and not to wait until the files are completely full) to react a little before in order to avoid packets loss. We have already shown in [5] that monitoring router internal state thanks to SNMP can help us to predict congestion.

Thus, each router supervises its own queues and, if necessary, triggers a decrease of emission at video server filter. If used resources are released, then, one can contact the filter in order to increase the number of layers to be sent to get a better video quality.

3 Simulation and Results

3.1 Implementation

We have modified ANTS [6] because it strongly limits the classes usable by protocols for safety reasons. These restrictions do not make it possible to develop protocols using all the possibilities of the active networks. Thus, we extended ANTS, for example to be able to request system information such as computing resources usages as well as router networks interfaces queues filling.

In particular, we modified the class `PrimordialNode` in order to change the variable `exportedAntsClasses` and added the classes necessary to the use of SNMP in order to be able to control the filling level of the queues of networks interfaces of routers.

In ANTS, the method `evaluate` of the class `ants.core.protocol` specifies the code applied to a packet incoming on an active router. Packets call this method where `n` represents current active node. Method `routeForNode()` route the packet towards the following router. The simplified algorithm (inspired of the ANTS Java programming) of this method for our protocol is shown figure 3.

It is impossible to use classical simulators such as Network Simulator 2 (NS2) since they do not take into account essential parameters (such as computing resources). So, in order to validate our new protocol, we developed a simulator. This one was implemented in Java and makes it possible to simulate the sending of flow through an active network.

The server sends flows made up of 25 packets per second (each packet corresponding to a picture) towards a client through a given number of routers. If a router receives an

```
boolean evaluate (Node n) {
 if (n.bandwidthAvalable<ourBandwidthLimit) {
   makeServerDecreaseVideoQuality();}
 if (this.alreadyDone) {
   return n.routeForNode(this,getDst());}
 else {
   if (n != lastNode) {
     if (n.resourcesAvailable<limit) {
       return n.routeForNode(this,getDst());}
     else {
       this.AggregateAllLayers();
       this.alreadyDone=true;
       return n.routeForNode(this,getDst());}
   }
   else {
     if (n.resourcesAvailable<limit) {
       this.AggregateOnlyBasicLayer();
       this.alreadyDone=true;
       makeServerDecreaseVideoQuality();
       return n.routeForNode(this,getDst());}
     else {
       this.AggregateAllLayers();
       this.alreadyDone=true;
       return n.routeForNode(this,getDst());}
   } } }
```

Fig. 3. SouceCode

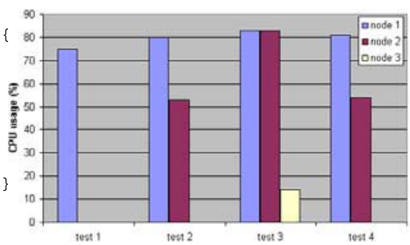


Fig. 4. CPU load

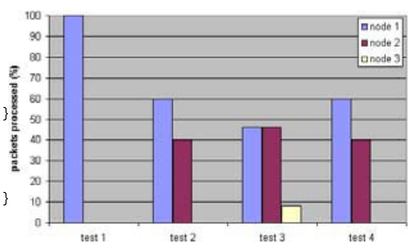


Fig. 5. Processing distribution

already aggregated packet, then it simply forwards it to the following router. In the contrary case, if its resources permit him, then it aggregates the packet which remains a given average time on the router before being able to be sent to the following router. This time is calculated from time measured during the analysis and tests of an algorithm of decompression of wavelet flow (decoding, aggregation, encoding). Once this time is out, the packet is marked "aggregated" and sent to the following router.

On each router, we analyze available resources (in order to check that those do not reach a critical point necessary to carry out the traditional operations of routing of the other traffics), percentage of "aggregated" packets locally as well as percentage of non-aggregated packets for reasons of leak of resources (the remainder corresponding to the packets already treated on a previous router). The critical point is configurable (in our simulation, this limit is fixed at 85%).

Then, we check on the client that all packets were aggregated. If the last router receives "non-aggregated" packets then it orders the transmitter (as well as all the crossed to the server) to decrease the number of layers sent.

The tests carried out set up a video in four layers sent to a client. The client and the server are separated by three active nodes (n=3) according to the figure 1.

The figure 5 shows the distribution of packets on the various crossed active nodes. The figure 4 shows the load of CPU of these active nodes. The two figures have to be observed in parallel.

In test 1, 10 hierarchical flows are sent towards the client (in order to test the load distribution on active routers). One can see (on figure 5) that the first node deals with 100% of the packets and (on figure 4) that this consumes 75% of its CPU. Thus, we are not very far from the limit that an active router can handle.

In test 2, the number of flows are doubled, we can then see that the second node start to aggregate some packets. The first node treats 60% of packets by consuming 80% of its computing resources and the second node treats the remainder by consuming 53% of its CPU. Thus, we can see that during this test, packets are well managed in a priority way on the first nodes. We can also note that we preserve some resources for the remainder of the traffic.

Test 3 brings into play 30 flows, the diagrams show that the first two active routers are not able to manage all packets (those treat 46% of packets each one, by consuming 83% their computing resources). The last active router starts to receive "non-aggregated" packets in a regular way, then it contacts the filter in order to ask for a decrease of throughput emission. This one removes a layer from the sent video. Test 4 corresponds to 30 flows with one removed layer, processing are strongly reduced and the first two routers become again able to aggregate all 30 flows.

These tests show that our protocol makes it possible to manage a greater number of light clients. Packets aggregation is distributed (decision of processing a packet is taken independently of other packets and of other active nodes) on active routers with a priority to first routers while keeping a minimum of resources for remainder traffic. Then, when the last router receives non-aggregated packets in a regular way it contacts the video server in order to make him reduce the video quality.

The fact that all packets are already aggregated on their incoming at the client makes it possible to consider the displaying of 25 frames/s on light clients such as PDA whereas without our protocol, these PDA will not be able to display a hierarchical video with more than 2-3 frames/s if they have the adequate codec.

4 Conclusion

We created a protocol allowing us to display hierarchical video on a light client by distributing layers aggregation on the various active nodes between the video server and the light client.

Computations are realized as soon as possible in order to be sure that the maximum number of packets is aggregated before their incoming to the client. Each active router decides independently to process packets. Moreover, the distribution of computations is done by respecting other traffics, i.e. in allocating them a minimum computation resources.

Our protocol also manages the computation resources of active nodes located on the way of video flow as well as the congestion at active node in order to be able to make the video server decrease the quality of the video in case of overloaded network. We also implemented a simulator in order to validate our protocol.

Our prime future work is to add to our simulator the management of network interfaces queues in order to be able to detect a possible congestion and to be able to test the effectiveness of our network resources management.

References

1. Taylor, K., Polyzos, G.C.: Performance measurements of a simple heirarchically coded image animation over various network testbeds. rapport de recherche S2K-93-39, University of California, Berkeley (1999)
2. Paté, D., Pansiot, J.J.: Hierarchical Video Multicasting and Packet Filtering. In: proceedings of Packet Video'01, Kyongju, Corée. (2001)
3. Tennenhouse, D.L., Smith, J.M., Sincoskie, W.D.,Wetherall, D.J., Minden, G.J.: A survey of active network research. IEEE Communications Magazine **35** (1997) 80–86
4. Bourgeois, J., Mory, E., Spies, F.: Netmovie: An architecture for adaptative multimedia transmission over wireless networks. In: R. Arabnia, editor, Proc. of the International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA'02) CSREA Press. (2002)
5. Fuin, D., Garcia, E., Guyennet, H.: Behavior and Performance of QoS Mechanisms on Different Router OS. In: ICN'2004, 3rd IEEE International Conference on Networking, Pointe-a-Pitre, Guadeloupe, French Caribbean. (2004)
6. Wetherall, D., Guttag, J., Tennenhouse, D.: ANTS: A toolkit for building and dynamically deploying network protocols (1998)

Frequency Cross-Coupling Using the Session Initiation Protocol

Christoph Kurth[1], Wolfgang Kampichler[2], and Karl Michael Göschka[3]

[1] Vienna University of Technology, Institute of Computer Technology,
Gusshausstraße 27-29, 1040 Wien, Austria
kurth@ict.tuwien.ac.at
[2] Frequentis Nachrichtentechnik GmbH, Wolfganggasse 58-60, 1120 Wien, Austria
wolfgang.kampichler@frequentis.com
[3] Vienna University of Technology, Institute of Information Systems,
Distributed Systems Group, Argentinierstraße 8, 1040 Wien, Austria
karl.goeschka@tuwien.ac.at

Abstract. VoIP is increasingly used in voice communication systems, where integration of legacy radios is an important feature. In addition to managing radio channel access, requiring Push-to-Talk (PTT) and Squelch signals, also cross coupling of radio channels is an important feature in air traffic control and public safety applications. Therefore we present a signaling approach based on the Session Initiation Protocol (SIP) in combination with a dynamic master slave model. It provides compatibility with standard SIP-phones and the existing expiry mechanism is used for detection of failure states. We complete our contribution with estimations of the audio and PTT-request delay in a coupled sector.

1 Introduction

Voice Communication Systems (VCS) traditionally have been built on proprietary hardware and protocols, using circuit switched technologies. With the dissemination of VoIP also voice communication centers migrate to packet based transmission. IP-networks together with common-off-the-shelf (COTS) hardware offer greatest flexibility in terms of location independence and convergence with data services.

The integration of extended call-features as well as radio signaling requires additional signaling capabilities, whereas a SIP-based method of radio signaling has already been presented in [3] and [4]. We take these concepts and contribute with signaling extensions for cross coupling of radio channels.

2 System Overview

A Voice Communication System (VCS) offers phone and radio services to a variety of different communication standards together with advanced callcontrol features for its operators. Thus it becomes applicable for communication centers in public safety, disaster operation and air traffic control (ATC) also. A VCS consists of multi-feature

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 680–689, 2005.

operator positions as well as standard phones which are connected peer to peer in a LAN (see Fig. 1). We consider an IP-LAN, signaling by means of SIP [1] and the Realtime Transport Protocol (RTP) [2] for media transport, because this is about to become a global standard. A local SIP-proxy is responsible for registration, authentication, user-mobility and dial-plan.

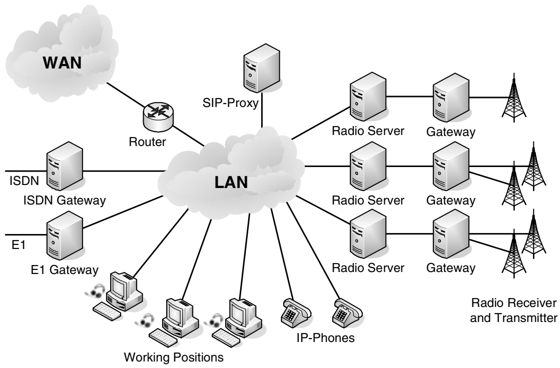


Fig. 1. SIP-based Voice Communication System

Besides gateways to different communication networks, e.g. to public telephony and the Internet, access to analogue or digital radios will be provided by means of radio gateways. Since a radio frequency is a shared media, a server has to control the access. The server can be connected to multiple radio gateways and a gateway can have connections to multiple transceivers to apply best-signal-selection (BSS) for optimization of the signal quality.

Since the whole system is built upon IP-networks, the gateways as well as working positions may reside on remote locations, with connection to the public Internet, or a leased IP-line. The hardware for network-, client- and server components consists of COTS products or dedicated SIP-enabled 3rd party products, like SIP phones, SIP servers and gateways.

3 Radio Service in SIP

Interconnection with analogue radios is one of the key features in many VCS fields of application. Users can key-in at radio channels, which may consist of several transmitters and receivers for a single frequency. Since audio transmission in traditional analogue radio is half duplex, an authority is required, which grants access to transmit messages.

Hence radio servers (one for each radio channel) control the access of the working positions to the radio gateways and grant or deny Push-to-Talk requests (PTT). To support also client connections with low bandwidth, the service offered by radio servers should be kept as slim in network load as possible. Therefore, with several transceivers a radio server offers a single RTP audio stream for the working positions, which are keyed in.

As shown in [4], SIP event notification [6] offers a suitable tool for managing access to a shared media like radio channels. Working positions as well as radio gateways generate PUBLISH requests [5], which contain their current PTT/SQU state. The radio server acts as Event State Compositor and in return distributes NOTIFY messages about the PTT/SQU-state of the channel to the parties, which have previously subscribed this event. Figure 2 shows the signaling sequence, when an operator presses PTT and access to the channel is granted.

For increasing reliability we suggest redundant transmission of the PTT/SQU information. Adding PTT/SQU information to the RTP audio packets offers a redundant signaling path, which may be sent over the IP-network on different routes and may be differently influenced by packet loss. The PTT/SQU is not separated from the audio information, but earns a higher delay due to packetization and jitter buffers. Alternatively RTCP may be used for redundant PTT/SQU transmission. Thereby the delay would be comparable with SIP event notification (tested in [4]).

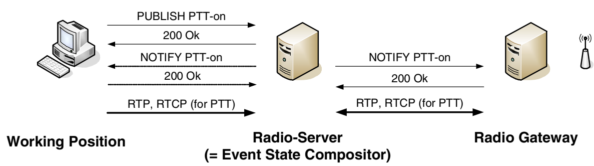


Fig. 2. Request and Grant of Radio Access

4 Cross Coupling of Radio channels

Coupling is a feature that offers definition of virtual radio channels, which consist of 2 or more physical radio channels. Then all operators using one of the channels involved, operate on a single virtual radio channel. A typical application for this can be found in voice communication centers, where the number of operators on duty depends on the time of the day. In times of a reduced volume of traffic, multiple radio channels can be managed by a single operator. Therefore cross coupling is used to provide variable fields of activity.

4.1 Variants of Implementation

The easiest variant of implementing coupling is local coupling. The initiator's client simply forwards the reception of one channel to all others in the coupling group. The radio servers do not need any additional functionality, but signaling and audio delay is high. The SIP, RTP and RTCP traffic between the channels is routed via the initiator's client, which causes CPU load on his working position.

A centralized approach uses a dedicated server for coupling. A coupling server is located at an additional hierarchical level above the radio servers. It receives coupling requests and controls coupled groups of radio servers in terms of PTT-allocation and audio mixing.

An intermediary, distributed solution is to extend the radio servers' capabilities for coupling. Each radio server can become master of a coupling-group or switch itself to slave mode, when another server masters the access to a virtual radio channel.

4.2 The Master-Slave Model

As a result the best suitable solution with the given SIP radio service is a master-slave model. For connecting peer-to-peer a set of working positions and radio gateways, an authority device for controlling access is necessary. Among the radio servers of the channels involved in the coupling, a master is determined, while all others become slave radio servers. Call connections of a star topology are set up and the master takes over the decision of granting or refusing PTT-requests.

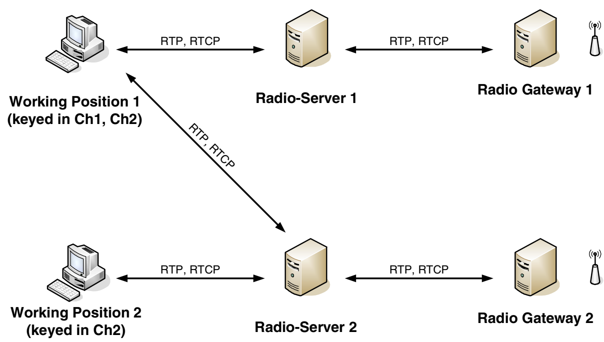


Fig. 3. Scenario before coupling

In an example scenario client 1 is keyed in at channel 1 and channel 2, whereas client 2 only listens to channel 2. Figure 3 shows the respective RTP connections before the coupling request is issued. In the following we go through the signaling steps of every party involved, starting with the initiating client.

4.3 The Initiator's Task

The client, which initiates a coupling group, has to decide, which of the radio servers involved will become the master. Therefore it may choose one by random or uses capability information, which can be retrieved by OPTIONS requests. The coupling request is done by means of a PUBLISH message [5] with the master and the list of slave servers specified in the message body.

If the master radio server is not capable of mastering a coupling group or it has no capacity available for another coupling group, it will refuse the coupling request and the initiator has to try again with a different master. If any of the servers involved supports neither master nor slave mode, cross coupling is not possible with this group of servers. Figure 4 shows the example scenario with the PUBLISH request sent by the initiator. Subsequently the initiator will put all calls to the slave radio servers on hold, because from now on audio mixing and PTT requests for the coupling group will be handled by the master server.

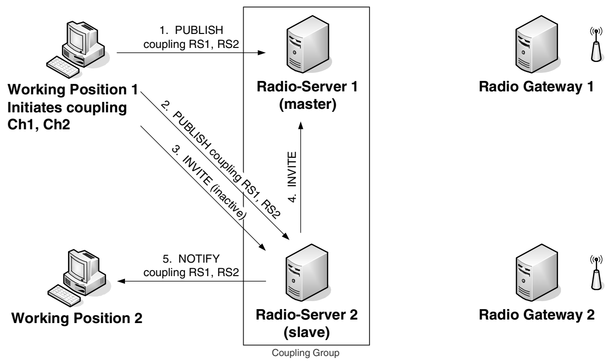


Fig. 4. Operator 1 initiates coupling

4.4 The Master Radio Server's Task

On initiation the master radio server receives INVITE-requests from the slave radio-servers, to set up the direct RTP audio connections and the PTT-signaling (as shown in Fig. 4). These calls represent the star topology between the radio servers mentioned above. If all slaves sent their invitations, the master sends a NOTIFY request about the coupling to its clients (including the initiator). For the initiator this acknowledges the establishment of the coupling.

Subsequently the slave radio servers inform the master about their current PTT state. Since originally every channel had its own radio server for PTT allocation, the master now has to migrate these PTT states into a common PTT state for the coupling group. If more than one PTT state is active, the master has to exclude all but one by signaling PTT-lockout, to reach a consistent common PTT state.

During normal operation a master radio server acts as event state compositor [5]. It receives PUBLISH requests and distributes a corresponding state by means of NOTIFY requests. The master's task includes:

- Grant or refuse PTT-requests: for all operators; also the ones, which are connected to slave servers (PTT request via a slave server)
- Forward SQU publications: from radio gateways; these may origin from its own gateways or being forwarded from a slave server
- Audio forwarding: between clients, slave radio servers and radio gateways
- Audio mixing: for frequency intercom in the coupling group

4.5 The Slave Radio Server's Task

In slave mode a radio server receives the coupling request from the initiator and a call setup request from the master radio server. Depending on the policy, clients will put their calls on hold (as shown in figure 5) or keep them.

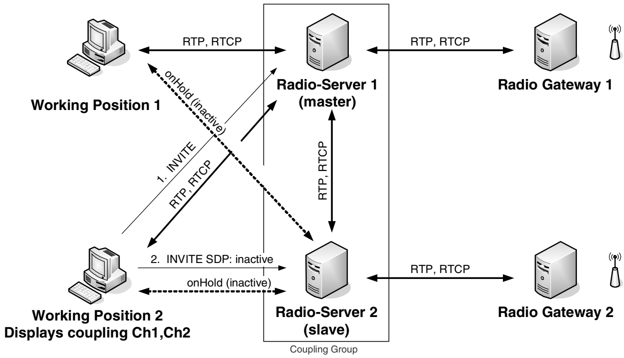


Fig. 5. Coupling Scenario: Final signalling and audio

During normal operation a slave switches back to forwarding of signaling and audio packets. Its task includes:

- PTT requests: are forwarded to the master
- PTT responses: are forwarded to all connected clients
- SQU publications: are sent to the clients and to the master radio server
- Audio forwarding: between clients, master radio_servers and radio gateways
- Audio mixing: for frequency intercom in the coupling group

4.6 The Passive Client's Task

The operator position clients, which did not initiate the coupling, receive a PUBLISH request and thus passively take part of the frequency coupling. Therefore some different scenarios and policies have to be taken into account.

If the client is only connected to one of the involved channels, the simple way is to keep the established call and just tell the user about the coupling, by displaying the information at the user interface.

Another policy, which can be applied also if the client is connected to multiple frequencies of the coupling, is always connecting to the master radio server. Therefore all calls to slave servers are put on hold and the master's channel has to be keyed in. In this case the passive client has the same access to the coupling group as the initiator. PTT requests take the shortest path to the event state compositor and audio as well as signaling delay of slave channels increases.

If the client has additional information about e.g. the capability or the location of the radio servers, it may remain connected to a radio server in its local domain, irrespective of which is the master. In a distributed VoIP environment with voice communication centers interconnected over a WAN, connecting to the master may cause an increase of WAN traffic. This policy helps to reduce the long distance network traffic.

4.7 Operation, Phone Clients

If an operator newly keys in a radio channel, which is member of a coupling, the client will immediately be sent a NOTIFY request with the details of the coupling. In the following the client can choose its connection to the coupling group and acts like a passive client.

For compatibility reasons it should also be possible to use a standard SIP-client or SIP-phone instead of a multifunctional working position. For key-in with a standard SIP-client the radio server must accept missing PTT-information and the reception of error responses to PTT/SQU notifications. For submitting radio messages the server has to generate a PTT-signal from the voice activity (known as vox-PTT). When a radio channel becomes coupled, the client will again respond with an error to the NOTIFY request. The radio server can now try to notify the coupling by means of mixing an acoustic signal or a voice-message into the client's audio stream. The user can be informed about termination of the coupling the same way. If a standard SIP-client shall become capable of initiating couplings, this could be realized by additional DTMF signaling.

4.8 Termination of Coupling

Normally a coupling definition is ended by a PUBLISH request, indicating an expiry-period of 0 (Expires: 0). The initiator has to take care of refreshing the publication in time, if he wants to keep the coupling. If a coupling publication expires (or is set to zero), all involved devices should go back to the state before the coupling has been initiated.

<u>Master radio server</u>:
- terminates calls with slave servers
- sends NOTIFY about end of coupling
- switches back to normal radio server operation

<u>Slave radio server</u>:
- sends NOTIFY about end of coupling
- switches back to normal radio server operation

<u>Client</u>:
- restores the key-in modes of the frequencies concerned

Since this mechanism also is used to resolve failure states and coupling inconsistency, each of the components must have its own timer, to realize expiration of a coupling. To indicate timeout-triggered termination of a coupling, both master and slave server have to publish a reason of termination; the master in the BYE request, the slaves in the NOTIFY request.

5 Performance

To evaluate the feasibility of the frequency coupling approach, we analyzed call setup delay as well as the audio transmission in the master-slave model.

5.1 Signaling

Compared with call setup signaling in a VCS, like for phone-calls or intercom, frequency coupling is less time critical. Also, initiation of a coupling may cause a burst of SIP messages, especially if radio-servers refuse becoming the master and if clients reconnect to the master radio server. Therefore coupling messages may be treated with a lower priority.

On the other hand, PTT/SQU signaling has to be improved, since in the worst case the master radio server is one additional hop away, compared with a non-coupled radio channel. Estimated with the results in [4], the PTT delay, from submitting the PUBLISH request to the NOTIFY reception at the gateway is

$$t_{PTT,coupled} = 1{,}5 * t_{PTT} = 3 * t_{forw} \tag{1}$$

which leads to a PTT delay of about 28,5 ms (based on dissipate, the SIP stack of KPhone). This value exceeds the limit of 25 ms, defined in the requirements for ATC [7]. Since this limit is founded in the low audio-delay of a circuit switched VCS, in a VoIP system with packetization delay this value will not cause loss of syllables.

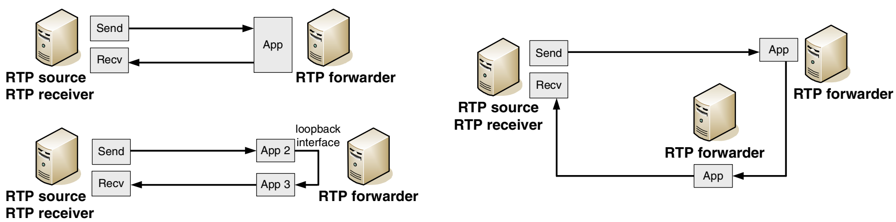


Fig. 6. RTP test scenarios

5.2 Audio Processing

RTP transmission in coupled radio channels should be as fast as possible. Depending on source and target, RTP packets can be forwarded via 3 intermediate servers. Audio received at a slave has to be forwarded via the master and another slave to the listening working position. The master-slave method keeps the increase of hops of the RTP streams consequently low, while using existing radio servers and providing fast signaling for granting or denying access to a coupled radio channel. Additionally accessing radios via standard SIP-phones is supported.

We performed tests to measure the influence of an additional hop in the RTP path. Therefore we compared the scenarios shown in figure 6, audio forwarding by a single radio server, 2 radio server applications on the same machine, whereas in the third test we used separate machines for 2 radio servers. An RTP test-application using jrtp 2.7.0 has been used as generator, forwarder and receiver on a Pentium IV CPU (2 GHz) hardware. The following list summarizes the results, which have been derived as average of a series of 100 packets with 20ms audio payload.

Scenario	delay
1 RTP forwarder	0,323 msec
2 forwarder, loopback	0,520 msec
2 Forwarder, separate machines	0,558 msec

The results show, that with a high-performance RTP stack the processing delay of the RTP packets is low, compared with the delay caused by conferencing and jitter buffering. Jitter buffers will be implemented in both master and slave radio servers, because both have to fulfil conferencing tasks, if an intercom service is provided also. Summing up the delays of the maximum length audio path leads to

$$t_{audio} = t_{packetization} + t_{jitter,slave} + t_{jitter,master} + t_{jitter,slave} + t_{jitter,client} \quad (2)$$

With a packet-size of 20ms and jitter buffer sizes of 20ms a resulting delay of 100ms (stack processing excluded) has to be taken into account. Adding the results of the measurement mentioned above and a delay for the management of the lists of clients and slave servers leads to the audio delay in a real coupling scenario. Thereby it makes only little difference between using the Ethernet loopback in comparison with two separate machines with a fast network connection of sufficient bandwidth.

6 Related Work

A project initiated by the European Organization for the Safety of Air Navigation (EuroControl) called AudioLAN [8], successfully showed the feasibility of an IP-based VCS. In contrast to the protocols proposed in this paper, they used H.323 for signaling. Meanwhile SIP emerges to become the future-standard for Internet telephony.

Another H.323 solution has been implemented at the University of New Hampshire [11]. A conference server provided coupling of several radio stations, but PTT requests were generated by voice activity. Thus no central authority for granting or denying access exists.

Ericson, Motorola, Nokia and Siemens published a specification for PTT signaling in GPS/GPRS networks [9]. They use SIP for initial call-setup and signal PTT by means of RTCP. But since the main fields of application are workgroups with GSM handsets, there is no need for extended radio-features like frequency cross-coupling.

Radio signaling also can be seen as a specific application of floor control [10], where the floor chair is radio server. As the floor control mechanisms are built for conferences, there is no demand for temporarily linking floors on user request.

7 Conclusion and Future Work

As under the possible fields of application of radio services with frequency coupling are public safety and disaster operation, the future focal point will be reliability and availability. Redundant server architectures as well as main and standby gateways will be needed to improve the availability. Therefore mechanisms for failure recognition and failsafe operation will become necessary.

Another important aspect is automatic configuration and management, where especially VoIP systems can show its surplus. Service registration at directory servers like provided by the Service Location Protocol (SLP) is a possible mechanism for setting up ad-hoc SIP environments.

Finally there are some special features, mainly used in ATC, like frequency forward and frequency intercom, which also offer multiple signaling variants and which should be specified for SIP based ATC.

References

1. J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, E. Schooler, ”SIP: session initiation protocol“, Request for Comments 3261, Internet Engineering Task Force, June 2002
2. H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, „RTP: A Transport Protocol for Real-Time Applications“,Request for Comments 3550, Internet Engineering Task Force, July 2003
3. K. Darilion, W. Kampichler, K. M. Goeschka, ”Event-based Radio Communication Signalling using the Session Initiation Protocol“, IEEE International Conference on Networks 2003, ICON ’03
4. K. Darilion, C. Kurth, W.Kampichler, K.M. Goeschka, “A service environment for air traffic control based on SIP”, Proceedings of the 37th Annual Hawaii International Conference on System Sciences 2004, Jan. 2004
5. A. Niemi, “An Event State Publication Extension to the Session Initiation Protocol (SIP)”, IETF Internet Draft, May 2004
6. A.B. Roach, “Session Initiation Protocol (SIP)-Specific Event Notification”, Request for Comments 3265, Internet Engineering Task Force, June 2002
7. EuroControl, “Voice Communication System Procurement Guidelines”, EuroControl, May 2003
8. EuroControl. Audio LAN – radio/telephone voice communication system based on internet technologies, http://www.openatc.org/
9. Ericson, Motorola, Nokia, Siemens, “Push-to-Talk over Cellular (PoC)”, Technical Specification, Aug. 2003
10. P. Koskelainen, J. Ott, H. Schulzrinne, X. Wu, “Requirements for Floor Control Protocol”, IETF Internet Draft, October 2004
11. J.H. Mock, “A Voice over IP Solution to the Problem of Mobile Radio Interoperability”, University of New Hampshire, May 2003

IP, ISDN, and ATM Infrastructures for Synchronous Teleteaching - An Application Oriented Technology Assessment

Mustafa Soy and Freimut Bodendorf

Department of Information Systems, University of Erlangen-Nuremberg,
Lange Gasse 20, 90403 Nuremberg, Germany
Bodendorf@wiso.uni-erlangen.de

Abstract. The quality and diversity of communication scenarios in distance education depend strongly on the available bandwidth. With the increase of existing network capacity teleteaching applications improve enormously in quality and quantity. Synchronous teleteaching which was initially just a video-stream based transmission of lectures is meanwhile enriched with multimedia applications and moves towards multimedia conferencing sessions. Advanced communication infrastructures for synchronous teleteaching applications are introduced. Solutions are based on standard desktop conferencing tools and streaming tools over IP on the one hand and transmission of media streams over ISDN and ATM codecs on the other hand. The pros and cons of each technology are outlined. Recommendations for appropriate usage are given.

1 Introduction

A first approach to synchronous teleteaching is to establish a point-to-point video conference between two remote access points. This permits the exchange of audio-visual signals. However this approach can not be seen as an efficient method for transmitting lectures. The technology should fit the teaching style and support the lecturer with multimedia aids in an appropriate way. Usually a computerized representation of instructional material leads to better results (particularly with regards to quality) in teleteaching courses, but the lecturer should still be able to use traditional blackboards or transparencies.

Quality and number of transmitted audio/video streams are closely tied to available network capacity. Even for teleteaching scenarios of low-quality, approximately 384 Kbps are transmitted per video stream. For high-quality views of the event, up to 15 Mbps are required for one video stream. With respect to the given technical and organizational aspects various types of teleteaching scenarios have been realized and evaluated. They include:

Tele-lecture: A lecture is transmitted to several remote access points. Students can interact with the lecturer and among themselves by using audio/video conferencing tools. For presentation of associated materials the lecturer can use several different remote presentation systems.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 690 – 697, 2005.

Tele-seminar: Different groups of students work in a virtual seminar environment transmitting presentations from each participating access point. This means that conferencing tools have to support many-to-many relationships for communication.

Tele-exercise: Groups of students using PC or workstations interact with an instructor by audio/video conferencing tools and specific groupware tools.

There are several other teleteaching scenarios you can think of, e.g., tele-excursions and distributed colloquiums.

At the University of Erlangen-Nuremberg hundreds of these scenarios have been run, experimenting with different communication infrastructures. Based on eight years and a lot of experiences, positive and negative ones, a substantial technology report on this "case" can be given. This is done in a condensed form in the next sections.

2 IP-Based Infrastructure

Multimedia conferencing over IP-based networks is becoming increasingly prominent. H.323 is the International Telecommunications Union (ITU) standard for IP-based multimedia conferencing. It covers standards for audio/video coding as well as standards for data exchange and control. H.323 conferencing systems consist of four network components: terminals, gatekeepers, gateways, and multipoint control units (MCU).

Terminals are endpoints on a LAN that provide real-time two way communication. The H.323 standard states that all endpoints must support voice, with video and data being optional. Hence, the basic form of an endpoint is an IP-Phone. However, most endpoints are video conferencing systems with additional support for data communication. Although the H.323 standard describes a gatekeeper as an optional component, it is in practice an essential tool for defining and controlling how voice and video communication is managed over the IP network. Gatekeepers are responsible for providing address translation between LAN aliases and IP addresses, call control and routing services to H.323 endpoints, system management and security policies. Gatekeepers provide the intelligence for delivering new IP services and applications. They allow network administrators to configure, monitor, and manage the activities of registered endpoints, set policies and control network resources such as bandwidth usage.

H.323 systems can interoperate with ISDN-based H.320 conferencing systems over a gateway. Essentially, gateways provide translation between a circuit-switched network such as ISDN and a packet-based network such as LAN, enabling the endpoints to communicate. To do this, they must translate between transmission formats and between control protocols. Gateways also have to transcode between various audio/video codecs used in LAN and ISDN devices. Most gateways have multiple ISDN connections and can support several conferences simultaneously. To allow three or more conference participants simultaneously, H.323 systems require a MCU. The H.323 MCU's basic function is to maintain all audio, video, data, and control streams between all the participants in the conference. Main components of an H.323 MCU are multipoint controller (MC) and multipoint processor (MP). MCs handle negotiations between all endpoints to determine common capabilities for audio/video processing. Most H.323 systems support IP multicasting and use this to send just one audio and one video stream to other participants. In contrast MPs perform audio mixing, data distribution, and video switching/mixing. Both (MC and MP) functions can exist in one unit or as part of other H.323 components. Most H.323 MCUs work in conjunction with, or include gatekeeper functionalities.

Streaming is a client/server technology that allows to broadcast live or pre-recorded data in real-time. Streaming technology offers a significant improvement over the download-and-play approach. It allows data delivery as a continuous flow with minimal delay before playback can begin. Hence, multimedia data is buffered before being played, and then is discarded. The video of the lecturer and other lecture material captured by cameras are usually analogous video streams which are fed into an encoding station. Figure 1 shows streaming within a network environment.

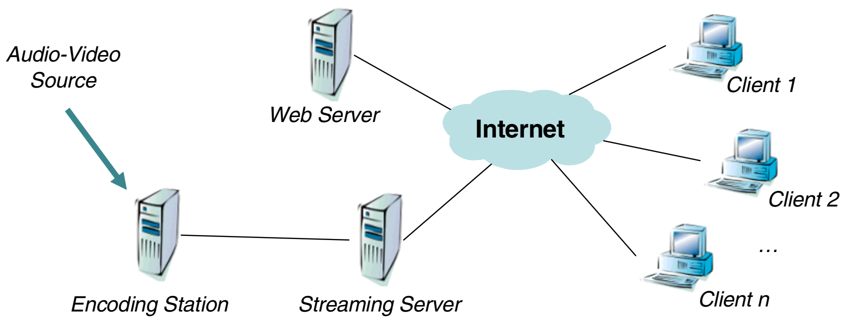


Fig. 1. Infrastructure for multimedia streaming

There are specific transport protocols available for streaming data such as RTP, RTSP, and MMSP. Real-Time Protocol (RTP) was developed by the Internet Engineering Task Force (IETF) to handle streaming audio/video and uses IP multicasting. RTP is a derivative of UDP in which a time-stamp and sequence number is added to the packet header. This extra information allows a receiving client to reorder out of sequence packets, discard duplicates, and synchronize audio/video streams after an initial buffering period. RealNetworks introduced with RealServer its primary server protocol, the RealTime Streaming Protocol (RTSP). To use RTSP, URLs that point to media clips on a RealServer begin with “rtsp://”. In return Microsoft introduced Microsoft Media Server Protocol (MMSP) as its primary server protocol. MMSP has both a data delivery mechanism to ensure that packets reach the client and a control mechanism to handle client requests such as “Stop & Play”. URLs that point to media clips on a Windows Media Server begin with “mms://”.

Most important advantages of IP-based communication infrastructures are:

– good suited for both multimedia conferencing and streaming multimedia data,
– wide spread availability and moderate costs,

On the other hand important disadvantages are:

– heterogeneous bandwidth availability and no quality-of-service for established, connections,
– low or medium quality of transmitted multimedia streams.

3 ISDN-Based Infrastructure

H.320 is the ITU standard for multimedia conferencing between endpoints connected over an Integrated Services Digital Network (ISDN), in contrast to H.323 over IP. Even though the long term prediction for multimedia conferencing is to use IP-based infrastructures, at the moment ISDN-based infrastructures are the easiest and most cost efficient. ISDN supports isochronous (regular timed) data transmission and the bandwidth is guaranteed once the connection is established. With ISDN, all information such as audio, video, and data is transmitted over the public switched telephone network. An ISDN connection has two possible interfaces: a Basic Rate Interface (BRI) or a Primary Rate Interface (PRI). The BRI consists of two circuit-switched B-channels, each of 64 Kbps that are used for data and one D-channel of 16 Kbps that is used for network control. The PRI is similar to the BRI, but with more channels and extra control bandwidth. In Europe, the PRI consists of up to thirty 64 Kbps B-channels, with 1920 Kbps for data transmission and one 64 Kbps D-channel for network control.

ISDN connections usually aggregate BRIs and share the same number for both B-channels. Known as ISDN-2, this provides a line speed of 128 Kbps and is usually used in desktop conferences. For increased bandwidth, ISDN-6 provides a line speed of 384 Kbps and is usually used in room-based conferencing tools. With ISDN-6, the sequence in which the lines are aggregated must be known. To run a multipoint conference over ISDN, participants have to use an H.320 MCU (see fig. 2) that connects and manages all ISDN lines. The basic function of any H.320 MCU is to maintain the communication between all participants in a conference. H.320 MCUs are hardware based as they need to connect to all ISDN lines from each participant. For example, to manage a conference between four H.320 systems, each at 384 Kbps (3 x BRI), a dedicated H.320 MCU needs to connect twelve BRIs. This is usually done as 24 x 64 Kbps within a PRI.

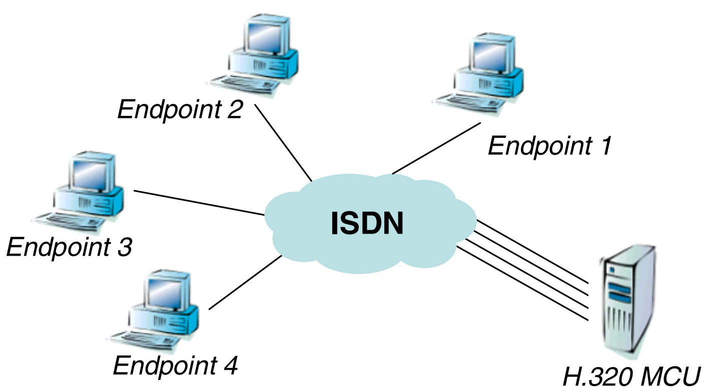


Fig. 2. ISDN-based network setup for H.320 multimedia conferencing systems

In general, dedicated MCUs support simultaneous sessions, more participants, higher bitrates, and more screen layout options. Application sharing within ISDN-based conferences is established just like within IP-based conferences by using the same protocols.

As a result the most important pros of ISDN-based communication infrastructures are:

– guaranteed bandwidth and quality-of-service for established connections,
– good availability and low costs,
– wide variety of commercial room based conferencing systems.

In return important cons are:

– not suited for large scale conferences and streaming applications,
– low or medium quality of transmitted multimedia streams.

4 ATM-Based Infrastructure

Due to the lack of reserving mechanisms, IP-based transport protocols are not the best choice for transmitting real-time media streams. Packets are always forwarded without considering network load. In situations with high network traffic, packets are discarded, thus lowering the quality of transmitted media streams. Another disturbing effect is the possible loss of synchronicity between audio/video streams. A better suited method for transmitting real-time media streams is the utilization of ATM networks. Advantages of ATM networks compared to conventional IP-based transmission include the definition of a quality-of-service parameter, thus reserving bandwidth for real-time applications. By guaranteed network capacity, support for point-to-multipoint broadcasts as well as an unlimited scalability of overall capacity, ATM fulfils all aspects required for synchronous teleteaching.

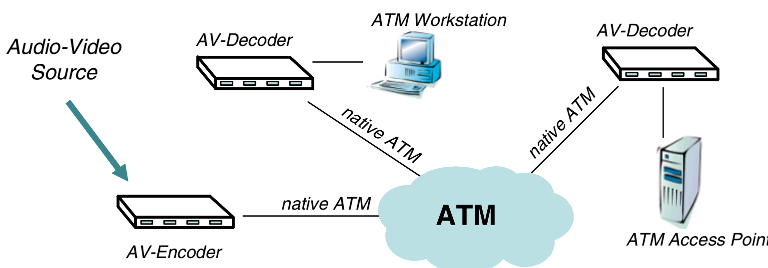


Fig. 3. ATM-based teleteaching infrastructure

Video and sound of the lecturer as well as additionally shown instructional materials are forwarded via ATM codecs (see fig. 3). Bandwidth requirements are approximately 15 Mbps for the video channel and 2 Mbps for audio. In order to facilitate connection setup, a virtual path (VP) has to be established between the access points. The VP should cover a minimum bandwidth of 34 Mbps. Inside the VP two bidirectional virtual connections (VC) – besides the VCs for TCP/IP connectivity for audio/video transmissions need to be configured permanently. This is also referred to as permanent virtual connection (PVC). Both VCs should be configured with constant bitrates. The configuration of 35400 cells per second for the video and 2400 cells per second for the audio stream has been proved to be very effective and robust. During non-transmission hours the whole available capacity is automatically assigned to the TCP/IP channels by the intermediate ATM switches. After starting ATM codecs for transmission, the configured bandwidth is reserved for audio/video streams thus guaranteeing optimal audio-visual quality. In this scenario permanent VCs are routinely used for transmission. Besides this, different setups with switched VCs are possible for ease of usage and flexible adoption of connection parameters like bandwidth. Corresponding solutions have been realized between different university locations in Germany.

Most important pros of ATM-based communication infrastructures are:

- very high and guaranteed bandwidth as well as quality-of-service for
- established connections,
- excellent suited for large scale conferences with high-quality multimedia
- data transmission,

In return most important cons are:

- not appropriate for multipoint conferences and streaming applications,
- poor availability and very high costs.

5 Conclusions

Each of the introduced technologies has its own strengths and weaknesses that should be considered carefully before deciding upon which one to use. The trade-off factors involved in determining the best infrastructure for a specific teleteaching application are:

- expectation levels and acceptable quality,
- available bandwidth,
- quality-of-service (reservation of bandwidth, synchronicity, error rate),
- multicast connectivity,
- number and location of participants,
- costs of installation and usage.

A crucial aspect choosing a communication infrastructure for teleteaching is to discuss and then set the expectation levels of the users. Therefore, a common understanding of media quality has to be established. At the University of Erlangen-Nuremberg three quality levels have been identified:

High-quality: PAL or NTSC resolution with H.264 or MPEG-2, MPEG-4 video coding (6 Mbps), CD-quality audio coding (1.4 Mbps), and data communication for additional material (1 Mbps). This gives a total of 8.4 Mbps. For native ATM video quality can be increase to DVD-quality with a total bandwidth consumption of 34 Mbps.

Medium-quality: CIF resolution with H.261/H.263 or MPEG-1 video coding (1.5 Mbps), near CD-quality audio coding (256 Kbps), and data communication for additional material (256 Kbps). This results in a total of approximately 2 Mbps bandwidth consumption.

Low-quality: QCIF resolution with H.261/H.263 video coding (256 Kbps), better phone-quality audio coding (64 Kbps), and data communication for additional material (64 Kbps). This makes a total of 384 Kbps bandwidth consumption.

Depending on the expansion (e.g. LAN vs. WAN) IP networks offer bandwidth beginning from 768 Kbps up to more then one Gbps. Even though teleteaching applications can be accomplished with IP-based infrastructures at all quality levels, high-quality teleteaching is delimited on LANs. The majority of running teleteaching applications with IP-based infrastructures are medium-quality. As most conferencing and streaming technologies are aligned for IP networks, IP-based infrastructures enable flexible and scaleable solutions. With IP-based infrastructures large scale internet-based broadcasts for tele-lectures as well as ad-hoc video conferences for tele-lectures can be realized. The costs for hardware, software, and network are at a moderate level. In particular streaming technology is becoming much more affordable. For H.323 conferencing systems, the installation costs need to cover probable upgrades to the network infrastructure such as faster switches and better routers. They also need to cover managing the network, endpoints, gateways, and MCUs.

ISDN is used today primarily as replacement for the Plain Old Telephone Service (POTS). It offers up to 1920 Kbps bandwidth and thus cannot be used for high-quality teleteaching. Although multipoint conferences are achievable, ISDN is better suited for point-to-point video conferences with small audience. Streaming over ISDN usually is not used. ISDN is technically mature as it supports isochronous transmission, exclusive access for reserved bandwidth, and error-free connections. High availability and moderate cost are also benefits of ISDN-based infrastructures. For H.320 based systems, the costs need to cover installation of ISDN lines and on-going line rental as well as the initial purchase of the H.320 equipment. With regards to multipoint H.320 conferences, the costs would have to cover the purchase of an MCU. However, it is probably more cost-effective to use systems with ISDN and multipoint options that will allow mixed-mode calls for up to three other sites over ISDN and IP.

ATM is an excellent technical solution for teleteaching applications. It offers many quality-of-service parameters, guaranteed network capacity, support for multicast, and scalability of overall capacity. ATM codecs work with native audio/video streams and do not use the IP stacks three and four. ATM-based infrastructures are highly appropriate for video conferences between two locations as well as for broadcasts of lectures to several locations with large auditoriums. It covers most aspects for tele-lectures. The configuration of multipoint connections is very complex. Multipoint connections are usually realized by multicasting techniques. Therefore no feedback channel is available when a multicast connection is established. Additional hardware is required to deal with this issue. ATM-based infrastructures are the most expensive and sophisticated teleteaching infrastructures. Both hardware components such as ATM codecs or ATM switches as well as network components are very costly. To maintain an ATM-based infrastructure qualified personnel as well as frequent teleteaching applications are necessary.

After eight years of configuring different communication infrastructures, running many teleteaching applications and monitoring performance and acceptance at the University of Erlangen-Nuremberg a combined infrastructure (IP and ATM) has been established for teleteaching courses. Figure 4 shows this infrastructure. As it can be seen, the usage of a multicast capable network for transmission is a core issue.

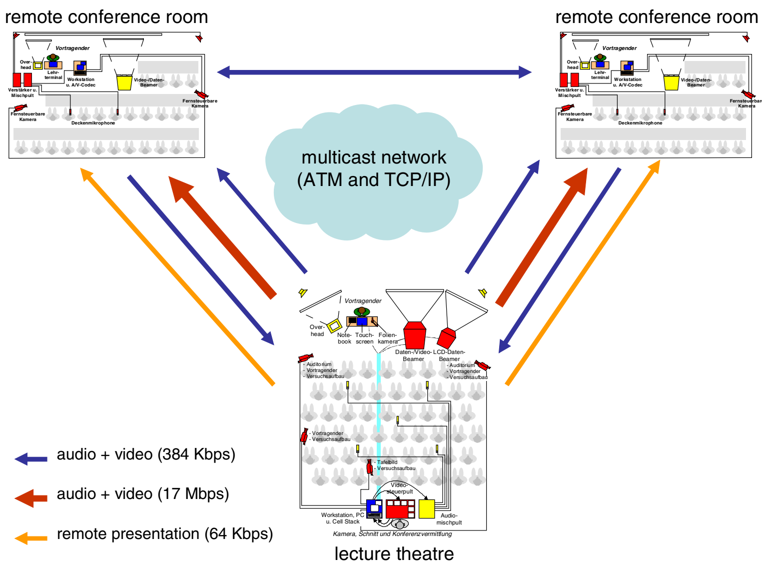


Fig. 4. Combined ATM- and IP-based infrastructure

Meanwhile, technical problems of synchronous teleteaching have been solved in university environments. Now, particular attention has to be paid to new concepts for high capacity network access at home and to growing social implications of home learning.

References

1. ITU-T Study Group: H.320 – Narrow-band visual telephone systems and terminal equipment.In: ITU Recommendation 03/04 (2004)
2. Bouras, C., Gkamas, A., Kapoulas, V., Tsiatsos, T.: Evaluation of teleteaching services over ATM and IP networks. In: Telematics and Informatics, Vol. 20. (2003)
3. Soy, M.: Videoconferencing with Advanced Services for High-Quality Tele-teaching. In: Proceedings of Trans-European Research and Education Networking Association (TERENA) (2001)
4. Liebeherr, J., Brown, S.R., Albertson, R.: An Interactive Telelecture System with Hybrid ATM/IP Networking. In: Multimedia Tools and Applications, Vol. 11 (2000)
5. Klein, A.: Teleteaching Scenarios for High-Bandwidth Networks. In: Computer Networks and ISDN Systems, Vol. 30. (1998)
6. ITU-T Study Group 16: H.323 – Packet-based multimedia communications systems. In: ITU Recommendation 02/98 (1998)
7. Grebner, R.: Use of instructional material in universal teleteaching environments. Computer Networks and ISDN Systems, Vol. 29. (1997)
8. ITU-T Study Group 8: T.120 – Transmission Protocol for Multimedia Data. In: ITU Recommendation 07/96 (1996).
9. CSWL Whitepaper: Basic Streaming Technology and RTSP Protocol, http://www.cswl.com/whiteppr/tech/StreamingTechnology.html (2004)

Two Energy-Efficient Routing Algorithms for Wireless Sensor Networks

Hung Le Xuan, Young-koo Lee, and Sungyoung Lee

Department of Computer Engineering, Kyung Hee Univerity, Korea
lxhung@oslab.khu.ac.kr, {yklee, sylee}@khu.ac.kr

Abstract. Power Conservation is one of the most important challenges in wireless sensor networks. We propose two minimum-energy routing algorithms CODE and SIDE. CODE (**CO**ordination-based **D**ata dissemination for s**E**nsor networks) addresses mobile sinks and considers energy efficiency not only in communication but also in idle-to-sleep state. SIDE (**SI**nk cluster-based data **D**issemination for s**E**nsor networks) addresses numerous stationary sinks and relies on loosely resource-constraints of a sink to ease the cost burden of sensor nodes. Our simulation results show that both algorithms reduce energy and prolong the network lifetime[1].

1 Introduction

A sensor network is randomly deployed by hundreds or thousands of unattended and untethered sensor nodes in an area of interest. These networking sensors collaborate among themselves to collect, process, analyze and disseminate data. Limitations of sensors in terms of memory, energy, and computation capacities give rise to many research issues in the wireless sensor networks. In this paper, we propose two algorithms. The first proposed algorithm, *Coordination-based Data Dissemination protocol* (CODE), addresses mobile sinks. We are motivated by the fact that handling mobile sinks is challenge of large-scale sensor network research. Though many researches have been published to provide efficient data dissemination protocols to mobile sinks [1],[2],[4],[5], they have proposed how to minimize energy consumed for network communication, without considering idle energy consumption. However, energy consumed for nodes while idling can not be ignored. M.Stemm *et al* [7] suggests that energy optimizations must turn off the radio to reduce number of packets transmitted and to conserve energy. In CODE, we take into account of energy for both communication and idle. CODE is based on grid structure and coordination protocol GAF [6] to provide an energy efficient data dissemination path to mobile sinks for coordinating sensor networks. The second proposed algorithm, *Sink Cluster-based Data Dissemination protocol* (SIDE), addresses numerous stationary sinks. We are motivated by the fact that future sensor networks will be composed of numerous sinks,

[1] This research work has been partially supported by Korea Ministry of Information and Communication ITRC Program joint with SunMoon University.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 698–705, 2005.

from several to tens and they are often far away from phenomena. SIDE exploits capacities of not only nodes but also sinks in order to reduce communication cost. Receiving data, a sink can act as a source's Agent to relay data to the other nearby sinks. Since the sink is not as tightly resource-constrained as sensor nodes, they can talk directly to each other or via a few nodes or sinks in order to ease the cost burden for sensor networks. Since the paper is composed of two different approaches which target different sensor network models, we separately present each protocol and its evaluation in Section 2 and Section 3. Section 4 concludes the paper.

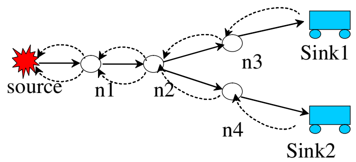
2 CODE Protocol

2.1 Theory

In CODE, we rely on the assumptions that all sensor nodes are stationary and aware of their residual energy and geographical location. Once a stimulus appears, the sensors surrounding it collectively process signal and one of them becomes a source to generate data report [2]. Sink and source are not supposed to know any *a-priori* knowledge of potential position of each other. To make unnecessary nodes stay in the sleeping mode, CODE is deployed above GAF protocol [6]. To establish the grid structure, each sensor node computes its grid ID [CX,CY] based on coordinate (x,y) as $CX = \lfloor x/r \rfloor\ ,\ \ CY = \lfloor y/r \rfloor$ (1) where r is the grid size and $\lfloor x \rfloor$ is largest integer less than or equal to x.

Data Announcement: When a stimulus is detected, the source propagates a *data-announcement* message to all coordinators using flooding. Every coordinator stores a few pieces of information for the data dissemination path discovery, including the information of the stimulus and the source's grid ID. Since the coordinator role might be changed every time, the grid ID is the best solution for nodes to know the target it should relay the query to. To avoid keeping data-announcement message at each coordinator indefinitely, the source includes a timeout parameter in data-announcement message. If this timeout expires and a coordinator does not receive any further data-announcement message, it clears the information of the stimulus and the target's location to release the cache.

Query Transfer: Every node is supposed to maintain a *Query INformation Table* (QINT) in its cache as an example in Fig.1. Each entry is identified by a tuple of (*query, sink, uplink*) (*sink* is a node which originally sends the query; *uplink* is the last hop from which the node receives the query). We define that two entries in QINT are identical if all their corresponding elements are identical. Receiving a query from an uplink node, a node first checks if the query exists in its QINT. If so, the node simply discards the query. Otherwise, it caches the query in the QINT. Then, based on target's location stored in each coordinator, it computes the ID of next grid to forward the query. In case the next grid contains no node (called void grid) or the next grid's coordinator is unreachable, it tries to find a round path. Each node is supposed to maintain a *one-hop-neighbor table*. If a node can not find the next grid's coordinator in this table, it



QINT n1		
Query	sink	uplink
A	sink1	n2
A	sink2	n2

QINT n2		
Query	sink	uplink
A	sink1	n3
A	sink2	n4

Fig. 1. Query transfer and Query Information Table

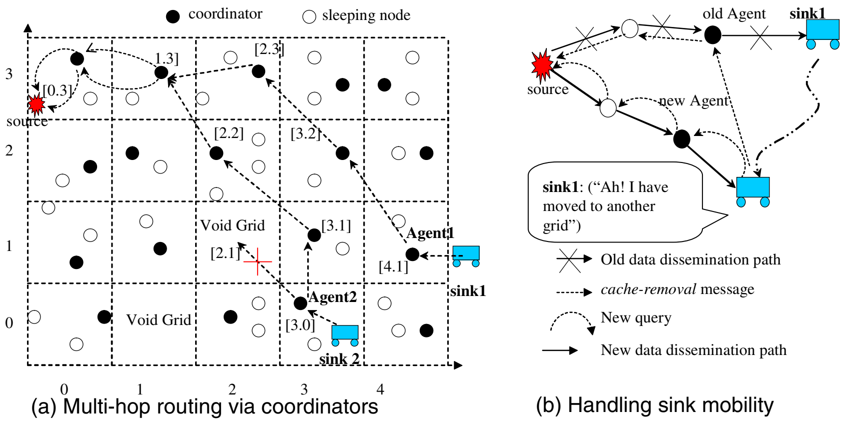


Fig. 2. Multi-hop routing via coordinators and Handling sink mobility

considers that grid as a void grid. For example in Fig.2a, sink1 sends query to *source* along the path [4.1], [3.2], [2.3], [1.3], [0.3]. However, to sink2, grid [3.0]'s coordinator can't find grid [2.1]'s neighbor (due to void grid). Therefore, it finds the round path as [3.1], [2.2], [1.3], [0.3].

Data Dissemination: A source starts generating and transmits data to a sink as it receives a query. Receiving data, a node on the dissemination path first checks its QINT if the data matches to any query and to which uplinks it has to forward. If it finds that the data matches several queries from the same uplink nodes, it forwards only one copy of data. For example in Fig.1, node n1 receives the same query A of sink1 and sink2 from the same uplink node (n2). Therefore, when n1 receives data, it sends only one copy of data to n2. Node n2 also receives the same query A of sink 1 and sink 2 but from different uplink nodes (n3,n4). Thus, it must send two copies of data to n3 and n4.

Handling Sink Mobility: Periodically, a sink checks its current location to know which grid it is locating. The grid ID is computed by the Formula (1). Based on grid ID, if it finds that it has moved to another grid, it first sends a *cache-removal* message to its *old Agent*. The *cache-removal* message is composed of query's information, sink's identification and target's location. The old Agent is in charge of forwarding the message along the old dissemination path as depicted in Fig.3. Receiving a cache-removal message, a node checks its QINT and

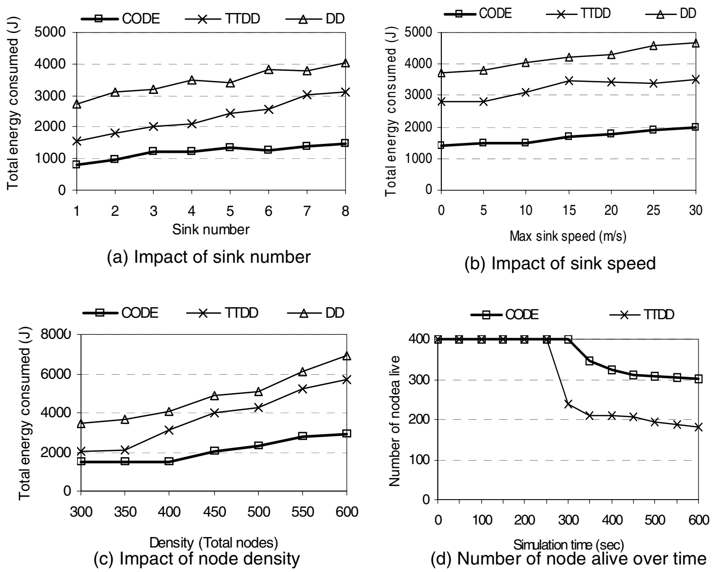


Fig. 3. CODE Simulation Results

removes the matched query. When this message reaches the source, the whole dissemination path is cleared out, i.e. each intermediate node on the path no longer maintains that query in its cache. Consequently, the source no longer sends data to the sink along this dissemination path. After old dissemination path is removed, the sink re-sends a query to the target location. A new dissemination path is established as described above. In case the sink moves into a void grid, it selects the closest coordinator to act as its Agent.

2.2 Simulation

We simulated CODE on SENSE [3] and compared to other approaches [1],[2]. A network comprises 400 nodes randomly deployed in a 2000mx2000m field. We use the same energy model used in ns2 that requires about 0.66W, 0.359W and 0.035W for transmitting, receiving and idling. The simulation uses MAC 802.11 DCF and nominal transmission range of each node is 250m [6]. *Two-ray ground* is used as the radio propagation model. Each data packet has 64 bytes, query packet and the others are 36 bytes long. The default number of sinks is 8 moving with speed 10 m/sec according to *random way-point model.* Two sources generate different packets at an average interval of 1 second. Fig.3a represents the impact of the sink number on CODE. It shows that CODE is more energy efficient than DD and TTDD. This is because efficient query and data dissemination paths are established based on grid structure to find a nearly straight path. Also, CODE exploits GAF protocol to turn off radio to conserver node's energy. Fig.3b plots the simulation results with different sink speeds. It shows the total energy consumed is less than the others. This is because the mobile sink communicates with the closest coordinator to receive data while it is moving thus the query only needs to be resent when it has moved to another grid. To evaluate the impact of node density on CODE, we vary the number of nodes from 300 to 600 nodes. Eight sinks move with speed 10m/sec as default. Fig.3c shows the energy consumption at different node densities. In this figure, CODE demonstrates better energy consumption than the other protocols. As the number of nodes increase, the total energy consumption slightly increases. In final experiment, we study the network lifetime. A node is considered as a dead node if its energy is not enough to send or receive a packet. Fig.3d shows that number of nodes alive of CODE is about 60 percent higher than TTDD at the time 600sec. This is because CODE focus on energy efficiency and rotating coordinators distributes energy consumption to other nodes, thus nodes will not quickly deplete its energy like other approaches.

3 SIDE Protocol

3.1 Theory

SIDE is based on GEAR protocol[4]. In SIDE, we assume that sensor nodes are stationary and aware of its geographical location and residual energy. Sinks are immobile, density (from several to tens) and not resource-constrained. Most current protocols use query and data aggregation to reduce communication cost while disseminating data to multiple sinks as illustrated in Fig.4a. In this case, a source propagates data to multiple sinks along a reverse path or by flooding [1],[2],[4]. Each node is supposed to maintain some routing knowledge so that whenever it receives data, it relays to an appropriate neighbor towards a sink. Our idea, as illustrated in Fig.4b, is that instead of simultaneously disseminating data to all sinks, a source first sends data to the closest sink. If a sink can communicate with others, it can itself relay data directly to them instead of through multi-hop sensor nodes. In Fig.4b, S1 relays data to S2 via a few sensor node hops and S2 can relay directly to S3. Though this approach reduces energy consumed but it may cause longer delay, therefore a source needs to know when it should use *Option1* (Fig.4a) or *Option2* (Fig.4b).

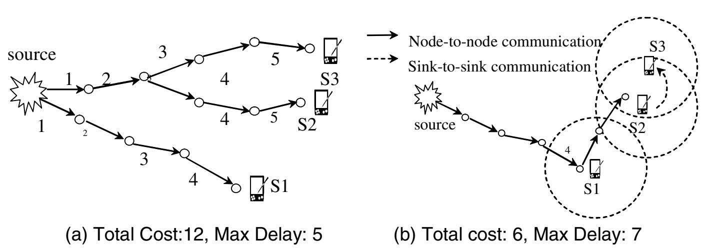


Fig. 4. Two approaches of data dissemination to multiple sinks

How to Select the Minimum-Energy Option: In [1], C.Intanagonwinwat *et al.* defined an *interest* as a list of *attributed-value pairs* that describes a task. In our approach, we also attach some description of the sink which propagates the task into an interest. This includes sink location and its communication range so that a source can decide which option to use to disseminate data. Since sensor network interests may often contain the target geographical location, we use GEAR algorithm to directly propagate interests to the target.

Assuming that there are three sinks and one source as illustrated in Fig.4 (S2 and S3 are within the communication range of each other, so that they can directly talk to each other). Those sinks propagate identical interests to the source along different path using GEAR. Each interest includes its communication cost from the sink to the source. Based on distances between sinks and source and the communication costs,we define an *average cost* as $C_{avg}{=}\sum_{i=1}^{n} cost(s_i)/\sum_{i=1}^{n} dis(s_i)$ (where n is the number of sinks, $cost(s_i)$ is total cost to deliver an interest from sink s_i to the source and $dis(s_i)$ is the geographical distance between sink and the source). A source has no information to compute the actual communication cost between two sinks, thus it computes the *estimated cost* between two sinks based on their position as: $s(s_i, s_j) = \begin{cases} 0; & if\, s_i\, and\, s_j\, can\, talk\, to\, each\, other \\ dis(s_i, s_j).C_{avg} & ; otherwise \end{cases}$
where $dis(c_i, c_j)$is distance between sink $s_i$and s_j , which is computed based on sink locations in the received interests.

We assume that S1 has the minimum communication cost of delivery the interest to the source. Therefore the source first sends data to S1. From S1, data is relayed to S2, S3. We then define the *estimated cost* to disseminate data from the source to a sink s_i as $e(s_i) = \cos t(s_1) + \sum_{i=1}^{n} e\,(s_i, s_{i+1})$.

Finally, the source comes to the conclusion that:

$$\begin{array}{l} C_1 = \sum_{i=1}^{n} cost(s_i); \\ C_2 = cost(s_1) + \sum_{i=1}^{n-1} e(s_i, s_{i+1}); \end{array} \Rightarrow \begin{array}{l} if\ (C_1 > C_2)\ then\ Call\ Option2 \\ else\ Call\ Option1 \end{array}$$

Arbitrary Sink Position and General Cases: So far, we have described the ideal case that all sinks are nearby. In fact, a sink is located at an arbitrary position in a sensor field, and we classify into two general cases as depicted in Fig.5. In case 1 as depicted in Fig.5a, the sink S2 can talk directly to S1 and S3. S1 is the closest sink that the source should send data first. However, by comparing the communication range of S2, the source finds that S2 can talk directly to S1 and S3, thus it first sends to S2. Then the sink S2 is in charge of forwarding data to S1 and S3 directly. In turn, S3 relays data through multi-hop to S4. This is a trade-off between communication cost and delay. Though sending along the path source→ S1 → S2 → S3 may take less cost than sending along source → S2 → (S1,S3) , yet the delay of S3 to receive data is longer. Consequently, it also causes longer delay for S4. In case 2(Fig.5b), some sinks are nearby among themselves but far away from the others. The best solution of this case to minimize communication cost is that the source groups nearby sinks into a cluster using *Sink Clustering Algorithm.* Then, the source only need to

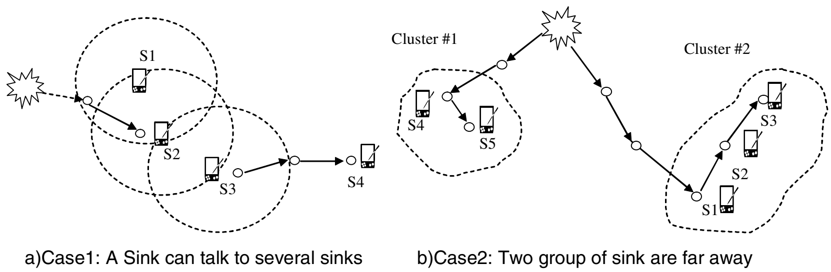


Fig. 5. Arbitrary Sink Position and General Cases

send data to the closest sink of the cluster and this sink is in charge of relaying data to the other sinks.

3.2 Simulation

To evaluate SIDE performance, we simulate a sensor network which consists of 200 sensor nodes randomly deployed in a 2000mx2000m field. The source generates different packets at an average interval of 5 second. Initially, the sink sends a query to the source. As the source receives a query, it calculates on query information to group sinks. Then it starts generating and sends data to the sinks. The simulation result is dependant on the scenario, i.e. the more close the sinks are, the less energy the network consumes. Therefore, we run our simulation on several scenarios with different position of sinks, and then we get the results averaged of all. Fig.6a plots the energy consumed by SIDE. In comparison with GEAR, SIDE reduces the energy consumed significantly as the number of sinks increases. When the number of sinks reaches five, energy consumed by SIDE is almost the same as GEAR. This is because sinks are scatted far away from each other. Therefore, the source has to individually send data to sinks like GEAR. However, as the number of sinks increases, they can talk to others so energy consumed by sensor nodes is reduced. Fig.6b shows the end-to-end delay of SIDE. This delay depends on sink's positions. However, in

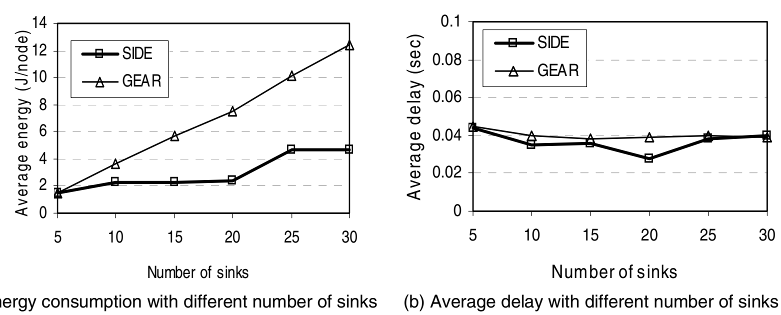


Fig. 6. SIDE Simulation Results

most cases, it is comparable with pure GEAR. This result is because of trade-off between communication cost and end-to-end delay as mentioned above. We do believe that, the approach brings an efficient scheme for sensor networks and can be employed in conjunction with other data dissemination protocols such as GPSR, Directed Diffusion, SPIN, *etc.*

4 Conclusion

We have proposed two algorithms, named CODE and SIDE, to reduce energy consumption for sensor networks. CODE is deployed above GAF to take advantages of coordination protocols. This approach is based on grid structure to find an energy-efficient data dissemination route between a source and mobile sinks. The other approach, SIDE, is based on GEAR. SIDE exploits capacities of sinks to ease the cost burden for sensor nodes. However, SIDE only copes with a large number of stationary sinks, rather than mobile sinks. Through our simulation results, we show that CODE and SIDE achieve energy efficiency and outperform other approaches. For our future work, we will combine CODE and SIDE to provide efficient data dissemination protocol for a large number of mobile sinks.

References

1. C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, F. Silva: Directed diffusion for wireless sensor networking. Networking, IEEE/ACM Transactions on Volume: 11 Issue: 1 , Feb. 2003 Page(s): 2 -16.
2. Fan Ye, Haiyun Luo, Jerry Cheng, Songwu Lu, Lixia Zhang: Sensor Networks: A two-tier data dissemination model for large-scale wireless sensor networks. Proceedings of the Eighth Annual ACM/IEEE International Conference on Mobile Computing and Networks (MobiCOM 2002), Sept 2002, Atlanta, GA
3. Gang Chen et al: SENSE - Sensor Network Simulator and Emulator. http://www.cs.rpi.edu/ cheng3/sense/
4. Yan Yu, Ramesh Govindan, Deborah Estrin: Geographical and Energy Aware Routing: a recursive data dissemination protocol for wireless sensor networks (2001). UCLA Computer Science Department Technical Report UCLA/CSD-TR-01-0023, May 2001.
5. F. Ye, S. Lu, L Zhang: GRAdient Broadcast: A Robust, Long-lived, Large Sensor Network. http://irl.cs.ucla.edu/papers/grab-tech-report.ps, 2001.
6. Y. Xu, J. Heidemannn, and D. Estrin: Geography-informed energy conservation for ad hoc routing. In Proc. of the Seventh Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom 2001), Rome, Italy, July 2001.
7. M. Stemm and R.H Katz: Measuring and reducing energy consumption of network interfaces in hand-held devices. IEICE Transaction and communication, E80-B(8): 1125-1131, Aug. 1997

An Energy Constrained Multi-hop Clustering Algorithm for Wireless Sensor Networks

Navin Kumar Sharma[*] and Mukesh Kumar[**]

[*]Assoc Software Engineer, CA Computer Associates, India
[**]Dept. of Computer Science and Eng., Institute of Technology, B.H.U., India
navin.sharma@ca.com, sharmamukeshkumar@yahoo.co.in

Abstract. A wireless sensor network is a new kind of wireless Ad-Hoc network consisting of a large number of small low cost, power constrained sensors deployed in a large area for gathering information for a large variety of applications. Since sensors are power constrained devices, it is quite important to organize them in some fashion in order to minimize total energy consumption in communication with base station. Clustering sensors into groups, so that sensors communicate information only to respective cluster heads and then cluster heads communicate the aggregated information to the processing center, is such an approach. In this paper, we propose an algorithm to organize sensors into clusters, which tries to extend the lifetime of network by forming balanced clusters as well as minimizing the total energy consumed in communication. In our algorithm, the number of hops is not fixed; rather it depends upon the spatial location of sensors.

1 Introduction

A wireless sensor network can be envisioned as consisting of hundreds or thousands of tiny sensing devices randomly deployed in a terrain of interest (e.g. thrown from an aircraft), called sensor field. These devices are able to continuously monitor a wide variety of ambient conditions (i.e. temperature, pressure, humidity, noise lighting condition or etc) and to detect event occurrences. Although they may have limited sensing and data processing capabilities, once they are empowered with a wireless communication component, they can coordinate among themselves to perform a big sensing task that cannot be achieved by a single sensor node

Since energy is a very scarce resource for sensors, networking of these sensors should be done in an efficient manner in order to increase the efficiency of application and the life time of network. Clustering is such an approach. In the clustered environment, the data gathered by each sensor is communicated to its cluster head directly or via some other sensor in the same cluster. The cluster head performs data fusion to correlate sensor reports and then communicate the data to the sink or processing center directly or through other cluster heads in the network.

There are many critical issues associated with the clustered topology of sensor network. A multi-cluster architecture is required to cover a large area of interest without degrading the service of the system. Since the sensors may not be uniformly dis-

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 706 – 713, 2005.

tributed over sensor field, some cluster heads may be heavily loaded than others, thus causing latency in communication, decrement in lifetime of the network and inadequate tracking of targets or events. In order to avoid such situations, we propose an algorithm that aims to increase lifetime of the network while minimizing the total power consumption in communication. Many clustering algorithms in various contexts have also been proposed in the past but very few have tried to create balanced clusters while minimizing the energy consumption in communication. Due to the space limitation, we are not describing the related work done in this field.

The rest of the paper is organized as follows:

In the next section we describe the energy constrained multi hop clustering algorithm. Description of the simulation environment and analysis of the experimental results can be found in section 3. Finally we conclude the paper and discuss about future research work in section 4.

2 Energy Constrained Multi Hop Clustering Algorithm

The primary goal of our algorithm is to maximize the lifetime of the sensor network as much as possible while minimizing the overall communication cost and keeping the cluster architecture balanced. By balanced cluster architecture, we mean such a cluster architecture in which the energy consumed in communicating unit amount of data by each sensor node to its cluster head, remains almost same for all clusters.

In order to achieve our objective, clusters are formed in such a way that energy of each cluster is almost equal, where energy of a cluster is defined as the total energy required in transmitting unit amount of data from each sensor node of that cluster to its cluster head. First, our algorithm forms clusters at the denser area taking into account the energy constraint, and gradually proceeds toward the sparser area. Intra cluster communication cost reduces the advantage of clustering. Also denser the cluster, lesser the cost be. If the network consists of a high number of dense clusters then a significant amount of intra cluster communication cost will be reduced. Our algorithm achieves this by giving preference to the formation of a denser cluster over that of a sparser cluster. So, the area is chosen in order of decreasing density, and the cluster is formed simultaneously.

For convention, we have assumed certain terms which are as follows:

E_C^i : Total energy dissipated in communication of 1 bit of data from each sensor in cluster i to its cluster head (considering multi - hop structure).

E_N : Total energy dissipated in communication of 1 bit of data from each sensor node to the sink node through clustered network architecture.

E_{th} : Energy Threshold – the maximum permissible E_C^i value in any cluster.

The optimum value of E_{th} is estimated through extensive simulation for each type of configuration such that the E_N value of sensor field attains minimum value (refer to Figures 1, 2 and 3).

We have assumed the same sensor energy model as described in [3] and the same energy cost that is used for data fusion/ aggregation via beamforming in LEACH [1].

We have made some additional assumptions which are as follows:

- All sensors are symmetric and the distribution of sensors in the sensor field may be uniform or non-uniform
- The communication environment is contention- and error- free.
- Cluster heads may follow any suitable routing algorithm to transfer data to the processing center, but for simulation purpose we assume that all cluster head send data directly to the processing center.
- Each sensor knows about its position through the GPS system or other measurement techniques.

As soon as the sensors in the sensor network are powered up, all sensor nodes start sensing and transmitting information about its location and ID with maximum transmission range. By employing an existing MAC protocol, the first sensor node accessing the channel successfully becomes the coordinator node. It then has the information about location of all other sensor nodes. Other sensor nodes keep on sensing the channel until the information about the clustering of network comes. The coordinator node executes the proposed algorithm to decide the clustered architecture of the network. The algorithm starts with dividing the entire sensor field into number of unit square cells of unit length and unit width. Each sensor node is then mapped onto the nearest vertex of the cell. In case it lies in the middle of a cell, it is mapped onto any one of the four vertices of the enclosing cell. For convention, we have assumed that there are three states for sensor nodes. The sensor node that takes the role of cluster head is said to be in state S_{CH}, the sensor that has become part of a cluster is in state S_C and the sensor that has not been a part of any cluster is in state S_I. Initially all sensor nodes are in S_I. We define the degree of a vertex as the total number of S_I nodes mapped onto that vertex. A list L of all vertices with their degree is maintained. A separate list L_S^i for each S_I sensor node i is maintained that contains the distance of other S_I sensor nodes from this sensor node in increasing order. Also, a separate pointer P_i for each list L_S^i is maintained that points to the index upto where the nodes can be included in a cluster formed taking node i as cluster head. Also, a counter is maintained that is updated after each cluster formation. The counter indicates the percentage of sensor nodes that are already a member of some cluster. When a node becomes a cluster head or member of a cluster its L_S^i is deleted and also that node is deleted from L_S^i of all other S_I nodes. The detailed steps of our clustering algorithm that will execute in the coordinator node are as follows:

Step 1. A vertex is chosen randomly by generating a random number (integer number less than or equal to the maximum limit) for X and Y coordinate of the vertex. Let us denote the selected vertex as v_i. (Randomized approach has an edge over serialized approach with respect to complexity and balanced cluster formation).

If (degree of v_i > 0) Go to Step 2.

Else If (Value of counter indicates more than 90%) Go to Step 4

Else Select a vertex with non-zero degree from L that is nearest to the v_i in L. Replace v_i with the new vertex. Go to Step 2.

Step 2. In this step, the sensor node that is neither S_{CH} nor S_C and which is nearest to the vertex selected in Step 1 (v_i) is selected as temporary cluster head (CH) and an auxiliary cluster is formed around it. The algorithm of forming a cluster is as follows.

While (true) {Take a non-cluster node (S_I) nearest to the CH. Find the minimum-cost-path of communication (direct or via other node that is already a member of current cluster) between node and the cluster head.

$E_C^i = E_C^i + E_{cn}$

E_{cn} is the energy consumed in the communication of 1 bit of data between the sensor node and CH through minimum cost path.

If ($E_C^i \le E_{th}$) {Include the sensor node in the current cluster}

Else {Include or exclude the current node in order to minimize $|E_C^i - E_{th}|$ and end the cluster formation process of current cluster }

Break}

It should be noted that mapping of the sensor nodes onto one of the vertices is done only for smooth progression of our algorithm. In cluster formation, actual locations of nodes are taken for all calculation purposes.

Step 3. After step 2, all the vertices within or on the square of side 2K units, centered at v_i are chosen. For all the vertices except v_i lying within or on the square and having degree greater than 0, a sensor node (S_I) nearest to the vertex concerned is selected as temporary CH and an auxiliary cluster is formed around it. The procedure of cluster formation is similar to that of step 2.

The value of K is determined empirically depending upon the average density of sensors in the sensor field. The value of K is determined as $\left\lfloor \frac{\sqrt{1+8D^{-1}}-1}{2} \right\rfloor$ where D is the average density of sensor nodes (nodes per vertex) in the sensor field.

After that, the vertex v_j, for which the auxiliary cluster has maximum number of sensor nodes is chosen as reference vertex. For choosing v_j, comparison is not required as the node ID having maximum number of sensors in its cluster is updated after every choice of vertex.

If ($v_j == v_i$) { The auxiliary cluster formed at vertex v_i is taken as permanent cluster and the states of sensors within this cluster is changed according to their roles, i.e., the cluster head becomes S_{CH} and other nodes within this cluster becomes S_C. Also these sensors are deleted from the vertex mapping list and degree of vertex concerned is adjusted accordingly. Also all required data structures such as degree of vertex, list L_S^i and pointer P_i etc. are updated. Then Go to Step 1}

Else { Take v_j as reference vertex and repeat Step 2 to 3 assuming v_i replaced with v_j .In this way we are trying to form a cluster of maximum density in the nearby area of randomly selected vertex. Moreover, Dynamic Programming Paradigm can be implemented in order to avoid multiple calculations of same cluster formation.}

Step 4. If 90 % of the total sensors are already a member of some cluster then rest of the sensors are distributed to the cluster of nearest cluster head in order to avoid formation of some highly unbalanced clusters, since the remaining nodes are quite likely to be highly sparsely distributed in the sensor field.

The main complexity of the algorithm that is of concern is computational cost incurred in terms of total time taken. Let the total number of nodes are n. Maintaining separate lists for each node require O($n^2 \lg n$) time. The time taken in forming an auxiliary cluster in the Step 2 is O(1). Therefore, the cost of one cluster formation is O(n) in worst case. Updating the data structures like L_S^i 's corresponding to the deletion of one node (here deletion means state change from S_I to S_C or S_{CH}) will take O(nlgn) time. And there can be at most n (no. of nodes) deletions over the complete run of the algorithm, so the amortized cost of deletion of n nodes is O($n^2 \lg n$) . Therefore, the overall time complexity of the algorithm is O($n^2 \lg n$). Although the amortized time cost is O($n^2 \lg n$), the complexity in practical cases goes far less than this. Since in our algorithm only one node is involved in cluster formation, the message complexity and time complexity for all other nodes is zero in Cluster Formation Phase. Also the role of coordinator can be assigned to a high power special sensor node and this assignment can be predefined in order to reduce further burden on network. Since the energy loss in computations is far less than the energy loss in message exchange, our algorithm has a clear edge over other algorithms.

After the clustering of the entire sensor field is completed, the coordinator node broadcasts the whole information about clustered network architecture i.e. the ID of cluster heads and ID's of all nodes lying within its cluster etc. to the all sensor nodes. After receiving the information, all sensors become part of their respective cluster and the cluster heads start controlling their clusters. Also the coordinator node becomes the part of some cluster and starts functioning as normal sensor node.

3 Experimental Evaluation

The performance of this algorithm depends upon the value of E_{th}. So our first objective is to find the optimum value of E_{th}, and then we will compare this algorithm with Load Balanced and Shortest Distance Clustering algorithm given in [3], and will see the effect of various routing algorithms in intra cluster message passing.

For the first part of analysis, we have taken a sensor field of 100 m x 100 m dimension with varying densities. Base station is assumed to be at center of field. We have tested our algorithm for three types of spatial distribution of sensors – Regular, Poisson and Aggregated Distribution. For aggregated distribution we have taken coefficient of aggregation (k) as 1. Fig. 1, 2 and 3 show the plot of E_N vs. E_{th} (defined above) for Regular, Poisson and Aggregated distribution respectively. From Fig. 1, 2 and 3, it is clear that the algorithm performs optimally when E_{th} lies between 2500 nJ to 3000 nJ. After taking the average of all test cases the optimum value of E_{th} is estimated as around 2675 nJ. But any value between above limits may be desirable.

Fig 4 shows the plot of E_N (at optimum value of E_{th}) vs. average density of sensors in the sensor field for all three types of distribution. We have also compared our algorithm with Load Balanced Clustering and Shortest Distance Clustering given in [3] for various routing algorithms used in intra cluster message passing. The various routing algorithms used for intra cluster routing are: Energy-Aware Routing, Minimum-Hop Routing, Direct Routing, Minimum-Distance Routing, and Minimum-Distance Square Routing [3].

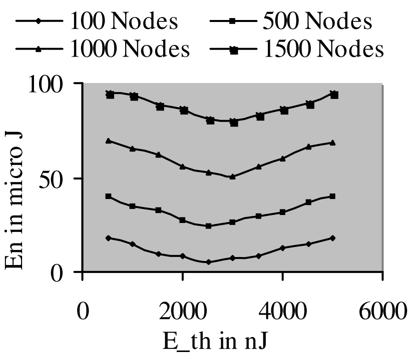


Fig. 1. Regular Distribution

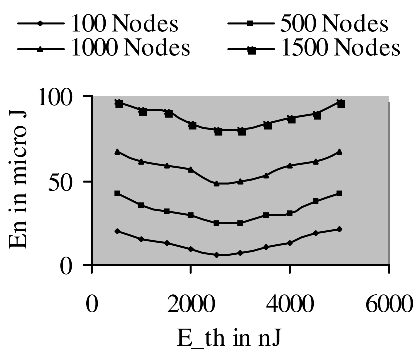


Fig. 2. Poisson Distribution

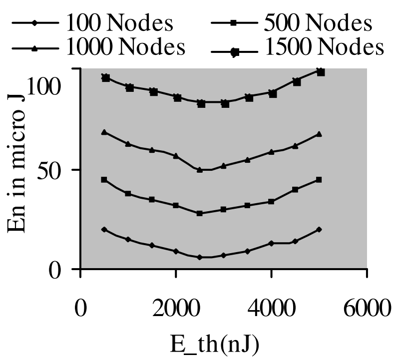


Fig. 3. Aggregate Distribution

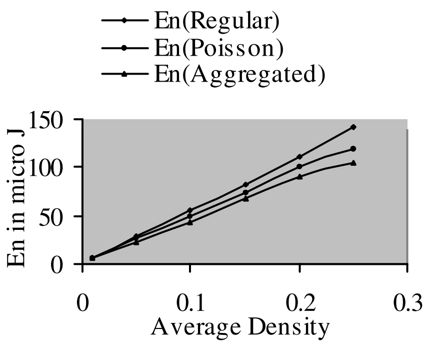


Fig. 4. En vs. Average Density

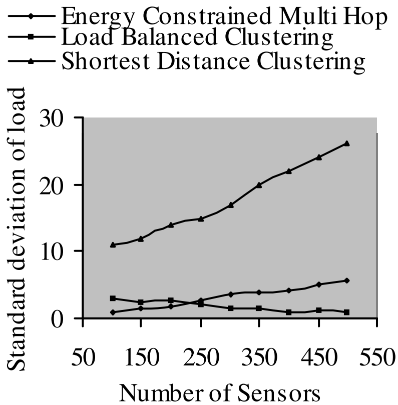


Fig. 5. Std. Deviation vs. No of Sensors

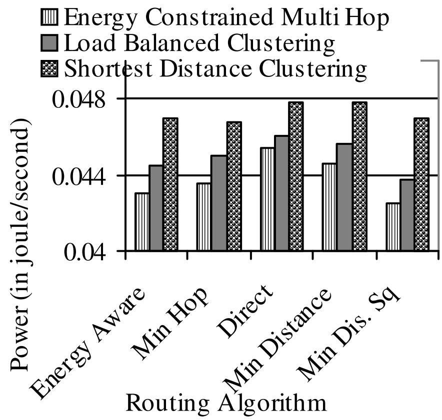


Fig. 6. Avg. Power Consumed in Comm.

The experiment is performed for 100 sensors distributed over the area of 100 m x 100 m. For this, we took the spatial distribution of sensors as poisson distribution. Initial energy of sensors is taken as 0.5 joule. Maximum range of sensor (for Load Balanced) is taken as 20 m. Packet length for data packet and routing packet is taken as 10 Kbit and 2 Kbit respectively. A sensing node produces data packet at the constant rate of 1 packet/second. Fig 6 shows the average power consumed in communication (this metric is an average of power consumed taken at different instance of time. It indicates the power utilized due to message traffic). It is clear that, our algorithm conserves more power than the other two algorithms. Fig 5 shows the standard deviation of load vs. density. For load balanced algorithm, the number of gateway sensor nodes is taken as 5. Number of sensors is varied from 100 to 500. From graph, it is clear that the standard deviation for our algorithm is comparable to that of Load Balanced. The deviation increases slightly with increasing density because the distribution varies with increasing density and our algorithm utilizes it. But the deviation doesn't go very high because the maximum number of permissible sensors in a cluster is bounded by energy E_{th}.

4 Conclusions and Future Scopes

In this paper, we have introduced an approach to cluster sensors by bounding the E_C^i, which in turn leads to energy efficient cluster architecture. Our future plans include extending the clustering model to allow sensor mobility, data aggregation/ fusion at each sensor nodes and appropriate routing protocol. Implementation of advance techniques of fault tolerance and recovery in our clustering model is also one of our future plans. Also, new algorithms of scheduling of sensors in the cluster and routing in the cluster can be implemented in order to increase the efficiency and life time of the network.

References

[1] W. Heinzelman, A. Chandrakasan and H. Balakrishnan, " Energy –efficient communication protocol for wireless microsensor networks", Proc of the Hawaii Conf on System Science, Jan 2000.

[2] Srajan Raghuwansi and Amitabh Mishra, "A self-adaptive clustering based algorithm for increased Energy-efficiency and Scalability in Wireless Sensor Networks", Proc IEEE VTC 2003, October 6-9,Florida

[3] Gaurav Gupta Mohamed Younis, "Load-Balanced Clustering in Wireless Sensor Networks", Proc IEEE ICC 2003, Anchorage, Alaska, May 2003

[4] Gaurav Gupta, Mohamed Younis, "Fault-Tolerance Clustering in Wireless Sensor Networks". Proc IEEE WCNC 2003, New Orleans, Louisiana, March 2003

[5] Seema Bandyopadhyay, Edward J. Coyle, "An Energy Efficient Hierarchical Clustering Algorithm for Wireless Sensor Networks." Proc IEEE INFOCOM 2003, San Franciso, CA, USA, March 30 - April 3, 2003.

[6] Li-Chun Wang and Chung-Wei Wang , "A Cross-Layer Design of Clustering Architecture for Wireless Sensor Networks", Proceedings of the 2004 IEEE ICNSC, Taipei, Taiwan, March 21-23 2004

[7] Malka N. Halgamuge, Siddeswara. M. Guru, Andrew Jennings "Energy Efficient Cluster Formation in Wireless Sensor Networks" International Conference on Telecommunication (ICT '03) Feb, IEEE Press, Papeete, Tahity, French Polynesia

[8] J. Tillett, R. Rao, F. Sahin and T.M. Rao, "Cluster-head Identification in Ad Hoc Sensor Networks Using Particle Swarm Optimization", Proc of *the IEEE International Conference on Personal Wireless Communication*, 2002

[9] Jain Shing Liu and Chunghung Richard Lin, " Power Efficiency Clustering Method with Power-Limit Constraint for Sensor networks" , Proceedings of Workshop on Energy-Efficient Wireless Communications & Networks (EWCN 2003)

[10] Julien Cartigny, David Simplot, and Ivan Stojmenovic, "Localized minimum-energy broadcasting in ad-hoc networks", In *Proc. IEEE INFOCOM 2003* (San Francisco, USA, 2003)

Maximizing System Value Among Interested Packets While Satisfying Time and Energy Constraints

Shu Lei, Sungyoung Lee, Wu Xiaoling, and Yang Jie

Department of Computer Engineering,
Kyung Hee university, South Korea
{sl8132, sylee, xiaoling, yangjie}@oslab.khu.ac.kr

Abstract. Interest is used as a constraint to filter uninterested data in sensor networks. Within these interested data some are more valuable than others. Sometimes among these interested data, we hope to process the more important data first. By using Reward to denote the important level of data, in this paper, we present a packet scheduling algorithm by considering four constraints (Energy, Time, Reward, and Interest) simultaneously. Based on simulation result, we find out that our ETRI-PS packet scheduling algorithm can substantially improve the information quality and reduce energy consumption.[1]

1 Introduction

Conventional research, such as Dynamic Voltage Scaling, has been utilized in all kinds of embedded systems. By extending DVS's concept into communication system, Dynamic Modulation Scaling has been proposed to schedule packet transmission [1]. The key idea is to let radio transmit packets with a lower transmission rate to reduce the energy consumption while still meeting all deadlines. Similar research [2] also follows this approach by applying lazy scheduling algorithm. These researches focus on minimizing energy consumption of a set of packets by delaying the finish of transmission till the deadline. A common drawback is that they only consider packets that already exist in the buffer, but do not provide threshold or constraint to filter and reduce the coming packets. Another research trend is presented in [3]. Cosmin Rusu, *et al.* consider **Energy**, **Time**, and **Reward** these three constraints simultaneously while Reward denotes the important level of task. They believe that in some overload systems, instead of processing several unimportant tasks that just consume a small amount of energy, it is more meaningful to process one valuable task which will consume more energy. In this **ETR** scheduling algorithm, whenever a new task is processed, it must have the highest ratio (reward value / energy consumption of this task) among all waiting tasks. Later on, in paper [4] Fan Zhang *et al.* extend this ETR algorithm for packet scheduling and present three different transmission algorithms. Data filtering is also an important approach to reduce energy consumption. Generally, a huge amount of data can be created by a large sensor network. However, in most of

[1] This work is partially supported by the Ministry of Commerce, Industry & Energy, Korea.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 714 – 721, 2005.

the time only the data of some sensor nodes that related to the users' purpose is really valuable. In [5], data-centric approach is proposed for power efficient data routing, gathering and aggregation. **Interest** is introduced as a constraint which is used to filter and reduce the unnecessary data. In these researches authors simply consider that all these packets have the same important level, but actually among these interested packets, some of them may be more important than others. For example, users are interested in the data of several sensor nodes which are used to monitor one object. The data created by the sensor nodes which are close to the observed object have more valuable information than the data created by the sensor nodes which are far from this object. Therefore, if we can introduce the **Reward** into these interested packets, we are able to select out and process the most important and valuable packet first. In this paper we present a new packet scheduling algorithm, namely, **ETRI-PS (Energy, Time, Reward,** and **Interest)**. Within this algorithm each packet has four parameters as follows: (1) energy consumption; (2) processing time; (3) important level; and (4) interest level.

In section 2 we present problem model. In section 3 we describe ETRI-PS scheduling algorithm. We present simulation work in section 4. And this paper is concluded in section 5.

2 Problem Model

We have one cluster in the heterogeneous sensor networks that is deployed as figure 1. Sensor nodes in area A, B, and C are used to monitor three different objects denoted by the triangles. We suppose that a user wants to know the information about the objects in area A and B. After querying and sensing, only the data collected by the sensor nodes which are located in area A and B can be accepted by the cluster head. Data from the area C will be rejected, because the user is not interested in them. If we look inside area A, we can find that the data sensed by the sensor nodes which are relatively closer to the observed object have the higher valuable information. Therefore, we consider these sensor nodes' data more important than others'. Then, whenever the cluster head receives the packets from sensor nodes, it will receive the most

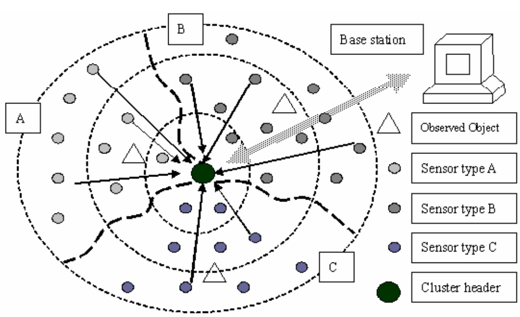


Fig. 1. Different sensor nodes send different packets to cluster head simultaneously

valuable packet among several interested packets first. We define the interested areas as A $\subseteq \{A_1, A_2, \ldots, A_M\}$. From each interested area A_x the cluster head can accept a subset of packets $P_x \subseteq \{ P_{x,1}, P_{x,2}, \ldots, P_{x,N} \}$.The processing time of the packet $P_{x,y}$ is denoted by $T_{x,y}$. Associated with each packet $P_{x,y}$, there is an Interest value $I_{x,y}$ and a Reward value $R_{x,y}$. Interest value is used to distinguish the interested packets from different areas. Reward value is used to denote the important level of this packet. The larger reward value means the higher important level. These four constraints of algorithm are defined as follows:

- The *energy constraint* imposed by the total energy E_{max} available in the cluster head. The total energy consumed by accepted packets should not exceed the available energy E_{max}. Whenever cluster head accept one packet, the energy consumption $E_{x,y}$ of this packet should not exceed the remaining energy *RE*.
- The *time constraint* imposed by the global deadline *D*. The common deadline of this user's data query is *D*. Each processed packet must finish before *D*.
- The *interest constraint* imposed by the interest value threshold *IT*. Each packet that is accepted must satisfy the interest value threshold $IT_{min} \le I_{x,y} \le IT_{max}$.
- The *reward constraint* imposed by the *value ratio* $V_{x,y}$ ($V_{x,y} = R_{x,y} / E_{x,y}$) between reward value $R_{x,y}$ and energy consumption of packet $E_{x,y}$. The larger $V_{x,y}$, the packet has, the more valuable the packet is.

The ultimate goal of ETRI-PS is to find out a set of packets P = $P_1 \cup P_2 \cup \ldots \cup P_M$ among interested packets to maximize the *system value,* which is defined as the sum of selected packets' *value ratio* $V_{x,y}$. Therefore, the problem is to

$$\text{maximize} \qquad {}_{x\in A,\ y\in P}\, V_{x\,y} \tag{1}$$

$$\text{subject to} \qquad {}_{x\in A,\ y\in P}\, E_{x,y} \le E_{\max} \tag{2}$$

$$ {}_{x\in A,\ y\in P}\, T_{x,y} \le D \tag{3}$$

$$IT_{min} \le I_{x,y} \le IT_{max} \tag{4}$$

$$x \in \mathrm{A}\ ,\ \mathrm{A} \subseteq \{A_1, A_2, \ldots, A_M\} \tag{5}$$

$$y \in P_x\, , P_x \subseteq \{1, 2, \ldots, \mathrm{N}\} \tag{6}$$

Because of the P = $P_1 \cup P_2 \cup \ldots \cup P_{M,}$ we can have the following formula as:

$$ {}_{x\in A,\ y\in P}\, V_{x\,y} = {}_{A1,\ y\in P1}\, V_{A1,\,y} + {}_{A2,\ y\in P2}\, V_{A2,\,y} + \ldots + {}_{AM,\ y\in PM}\, V_{AM,\,y} \tag{7}$$

From formula (7), we can find that the real problem of ETRI-PS is to find out the subset of $P_x \subseteq \{1, 2, \ldots, \mathrm{N}\}$ to maximize the *system value* for each interested area A_x. Thus, the problem is to

$$\text{maximize} \qquad {}_{x\in Ax,\ y\in Px}\, V_{x\,y} \tag{8}$$

$$\text{subject to} \qquad {}_{y\in P}\, T_{x,y} \le D \tag{9}$$

$$IT_{min} \le I_{x,y} \le IT_{max} \tag{10}$$

$$E_{x,y} \le RE \tag{11}$$

$$x \in \mathrm{A}\ ,\ \mathrm{A} \subseteq \{A_1, A_2, \ldots, A_M\} \tag{12}$$

$$y \in P_x\ ,\ P_x \subseteq \{1, 2, \ldots, \mathrm{N}\} \tag{13}$$

Inequality (9) guarantees that the *time constraint* is satisfied. Inequality (10) guarantees that only the interested packets are accepted, and inequality (11) guarantees that the energy budget is not exceeded. In order to solve the problem that is presented by (8)-(13), we give the following ETRI-PS algorithm.

3 ETRI-PS Packet Scheduling Algorithm

Before sending the real data of a packet, sensor node can send its packet's parameters to cluster head by including them in a small packet, which just consumes very limited energy. We give a name to this kind of small packet as *Parameter Packet (PP)*. There is a physical buffer that exists inside cluster head to store these *PPs*. After receiving these *PPs*, cluster head can decide which packet to be accepted based on these sent parameters. We can define our ETRI-PS algorithm into these following steps:

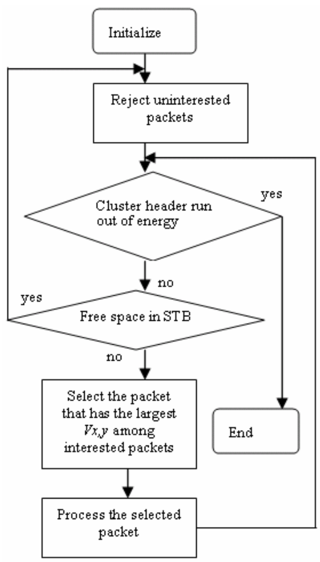
Step 1: Initialization. After receiving PP $\subseteq \{PP_1, PP_2, \ldots, PP_N\}$, we assume that tables exist inside the cluster head for storing parameters of every packet i ($i \in$ PP): *energy consumption* $E_{x,y}$, *processing time* $T_{x,y}$, *reward value* $R_{x,y}$, and *interest value* $I_{x,y}$. For each PP_i, there are energy consumption for checking CE_i and a period of time for checking CT_i. We also use two structure arrays, *considered(i)* and *selected(i)* of size N, to store the information for all received *PPs*. Initially, we start with an empty schedule (*selected(i).status = false*) and no *PP* is considered (*considered(i).status = false*). The set of selected *PPs* (initially empty) is defined as S = {*(i)* | *selected(i).status = true*}. After selecting the *PPs*, cluster head accepts packets that are corresponded to these selected *PPs*. Therefore, packet's parameters can be expressed as *considered(i).*$E_{x,y}$, *considered(i).*$T_{x,y}$, *considered(i).*$R_{x,y}$, *considered(i).*$I_{x,y}$, *selected(i).*$E_{x,y}$, *selected(i).*$T_{x,y}$, *selected(i).*$R_{x,y}$, and *selected(i).*$I_{x,y}$. We define four variables: 1) *checking energy* ($_{i \in PP} CE_i$) is used to store total energy consumption for checked PPs; 2) *checking time* ($_{i \in PP} CT_i$) is used to store total processing time for checked PPs; 3) *processing energy* ($_{i \in PP}$ selected(i).$E_{x,y}$) is used to store total energy consumption for processed packets; and 4) *processing time* ($_{i \in PP}$ selected(i).$T_{x,y}$) is used to store total processing time for processed packets.

Step 2: We Filter and Accept Packets Based on the ETRI Constraints. A packet that can be accepted should satisfy all the following criteria:

- This packet's *PP* is not considered before (*considered (i).status = false*).
- The current schedule is feasible (*checking time + processing time*) $\le D$.
- By accepting this packet to current schedule, the energy budget is not exceeded (*checking energy + processing energy + considered(i).*$E_{x,y} \le E_{max}$).

- This packet is intentionally queried by user ($IT_{min} \le considered(i).I_{x,y} \le IT_{max}$).
- Among all the *PPs* that satisfy the above criteria, select the one that has the largest $considered(i).V_{x,y} = considered(i).R_{x,y} / considered(i).E_{x,y}$.

After choosing the *PP*, cluster head can send Acknowledge back to accept new packet. In addition, for those packets which user is not interested in, their corresponded sensor nodes will discard them.




```
Step1: Initialize: selected(i).status = false, consider(i).status = false,
       ∀i ∈ {1, 2, …, N}; checking_energy = 0; checkin_ time = 0;
       processing_energy = 0; processin_ time = 0;
Step2:
    2.1 If (cluster_header.buffer has free space) = true Then
            Receive_PP(); Calculate_checking_energy();
            Calculate_checking_time();
        End If
    2.2 If (checking energy + processing energy) ≤ E_max Then
            If (checking time + processing time) ≤ D Then
              For i To cluster_header.buffer = N
                If (IT_min ≤ considered(i).I_x,y ≤ IT_max) = false Then
                   Delete_uninterested_PP();
                Else
                   considered(i).status = true;
                End If
              Next
     2.3 For i To cluster_header.buffer = N
             If consider(i).status = true Then
               j = Find_out_the_PP_that_has_the_largest_V_x,y();
               selected(j).status = true;
            End If
            Next
Step3:  For j To cluster_header.buffer = N
             If selected(j).status = true Then
               Receive_and_process_paceket();
               Calculate_processing_energy_and_processing_time();
             End If
            Next
            Else Stop_working()
            End If
        Else Stop_working ()
        End If
Step4: End
```

Fig. 2. Flowchart and source code of ETRI-PS

Step 3: We Transmit Accepted Packets to Base Station by Using Dynamic Modulating Scaling. As the algorithm that has been presented in [1], let radio transmit packets with a lower transmission rate to reduce the energy consumption while still meeting all deadlines.

Another Aspect: Replace or Drop a Packet. A new packet is always accepted if possible. When receiving new *PP* from sensor node, if the buffer is full, we can replace or drop a packet based on the following criteria:

- This packet's *PP* is selected (*selected(i).status = true*).
- Among all selected packet's *PPs*, find out the one that has the smallest $selected(i).V_{x,y} = selected(i).R_{x,y} / selected(i).E_{x,y}$.
- If this found one is not the new packet that is going to be accepted, we use this new packet to replace this found one, otherwise, we drop this new packet.

4 Simulation and Discussion

In simulation, we randomly deploy nine sensor nodes. And we randomly initialize these nodes with: *total energy* (scope: from 111 to 888), *buffer size* (scope: from 6 to

9). In addition, we design 8 different packets that are randomly initialized with the following four parameters: energy consumption (scope: from 3 to 10), processing time, reward value (scope: from 3 to 10) and interest value (scope: from 3 to 10). Eight of these nine sensor nodes are chosen to be the packet generators which randomly create eight different packets and send to the remaining one. The remaining one works as the cluster head. For this cluster head, we design three parameters: *total energy = 666*, *buffer size = 6*, and *interest threshold = 5*. The meaning of threshold is that we just accept the packets when their interest values are belonging to the top 5 among these 8 packets. These eight sensor nodes are organized into three groups based on their packets' interest values. Interest value {8, 9, 10} are considered as group A, {6, 7} are considered as group B, and {3, 4, 5} are considered as group C. Therefore, the cluster head just accepts the packets from area A and B. And the checking energy is designed to be 0.3, which is 10% of the minimum packet consumption 3. Besides ETRI-PS, we provide two different existing packet scheduling algorithms to run on cluster head for comparison:

1) Compared Algorithm one (CA 1) [1]:

a) In FTB: No *interest constraint* and *reward constraint*
b) In STB: Maximizing system lifetime (*Dynamic Modulation Scaling*)

The cluster head doesn't set any threshold to reduce the incoming packets, but just simply receives packets and relays them. Once it gets a packet, it will always process this packet just meeting its deadline.

2) Compared Algorithm two (CA 2) [4]:

a) In FTB: Maximizing reward value, but no *interest constraint*
b) In STB: Maximizing system lifetime (*Dynamic Modulation Scaling*)

The cluster head always accepts the packet that has the largest *value ratio* among several checked packets. Once it gets a packet, it will always process this packet just meeting its deadline.

We design the simulation parameters as follows: 1) *lifetime* of Cluster Head (CH), 2) *checking energy* of cluster head, 3) *processing energy* of cluster head, 4) *energy utilization* of cluster head (*energy utilization = processing energy / (checking energy + processing energy)*), 5) *processed packets number* by cluster head, 6) *total created packets* from sensor nodes, 7) *discarded packets* in sensor nodes, 8) *average interest value* per packet, 9) *average reward value* per packet.

From figure 3, we can find that for a given amount of energy, by using the *Dynamic Modulation Scaling*, the *lifetimes* of three different algorithms are almost same. As the result of the figure 4, the *checking energy* of ETRI-PS is much more than the *checking energy* of others. The reason is that we add the *interest constraint* in this ETRI-PS algorithm. Naturally, the energy that can be used to process packets is lower than others (*checking energy + processing energy = E_{max}*). This consequently causes relatively low *energy utilization* of ETRI-PS, as showed in figure 5. Even though the *energy utilization* of ETRI-PS is relatively lower than others, by using our ETRI-PS packet scheduling algorithm, we can still substantially reduce the energy consumption of whole sensor networks. The saved energy comes from the normal sensor nodes but not from the cluster head. By analyzing the figure 6, we can find that the *processing*

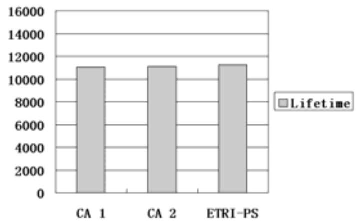


Fig. 3. Lifetime of cluster head

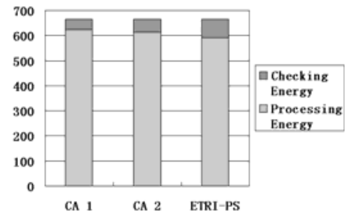


Fig. 4. Checking energy and processing energy

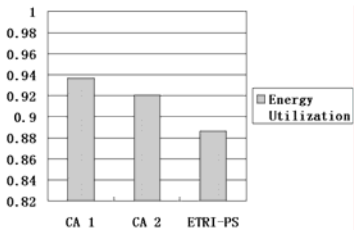


Fig. 5. Energy utilization of cluster head

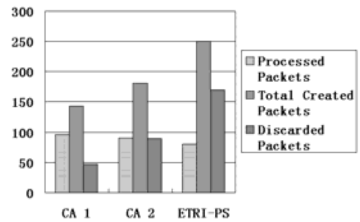


Fig. 6. Total created packets = processed packets + discarded packets

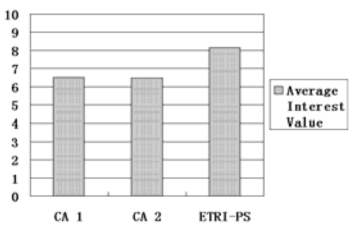


Fig. 7. Average interest value per packet

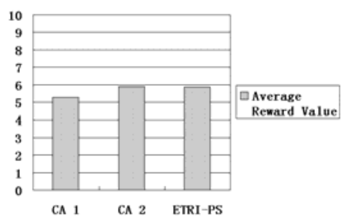


Fig. 8. Average reward value per packet

ratio (*processing ratio = processed packets / total created packets*) of ETRI-PS is much lower than others; inversely, the *discarding ratio* (*discarding ratio = discarded packets / total created packets*) is much higher than others. The lower *discarding ratio* the sensor nodes have, the more uninterested packets the sensor nodes send. Thus, the more unnecessary energy is consumed. In conclusion, by using the ETRI-PS, the sensor nodes can reduce the unnecessary transmission of uninterested data to reduce the energy consumption. As we design the *interest threshold* to just accept the packets that have the larger interest values, therefore, the desired *average interest value* should be larger than that of other algorithms. Figure 7 shows that the *average interest value* of ETRI-PS is much larger than others, that means the ETRI-PS can exactly process the user interested packets well. Figure 8 shows the comparison among three algorithms' *average reward values*. In the algorithm CA 1, because we do not intentionally maximize the *value ratio* ($V_{x,y} = R_{x,y} / E_{x,y}$), as a result, the *average reward value* of CA 1 is relatively smaller than others. Compared with CA 2, even though we add the *interest constraint* to CA 2, the *average reward value*s of two algorithms are almost same. This means the ETRI-PS can inherit the original purpose of ETR well.

5 Conclusion

Packet scheduling algorithm for communication subsystem is a potential approach to reduce energy consumption of sensor networks. ETRI-PS provides us a prioritized transmission scheduling algorithm according to the transmitted data's important level. By using ETRI-PS packet scheduling algorithm, we can achieve the following contributions: (1) Use interest constraint as the threshold to filter the uninterested incoming packets to reduce the energy consumption; (2) Use reward constraint to choose the high quality information and minimize the queried packet number to minimize the energy consumption but still satisfy the minimum information requirement. As the simulation result shows, by using the ETRI-PS packet scheduling algorithms, we can easily reduce energy consumption of sensor nodes and enhance the quality of queried information.

References

1. Schurgers, C., Raghunathan, V., Srivastava, M.B.: Modulation Scaling for Real-time Energy Aware Packet Scheduling. Global Communications Conference, San Antonio, Texas (2001)
2. Prabhakar, B., Biyikoglu, E.U., Gamal, A.E.: Energy-efficient Transmission over a Wireless Link via Lazy Packet Scheduling. IEEE/ACM Transactions On Networking, Vol. 10 (2002)
3. Rusu, C., Melhem, R., Mosse, D.: Maximizing Rewards for Real-Time Applications with Energy Constraints. ACM TECS, Vol. 2 (2003)
4. Zhang, F., Chanson, S.T.: Throughput and Value Maximization in Wireless Packet Scheduling under Energy and Time Constraints. 24th IEEE International Real-Time Systems Symposium (2003)
5. Krishnamachari, B., Estrin, D., Wicker, S.: Modeling Data-Centric Routing in Wireless Sensor Networks. 6^{th} international workshop on Modeling analysis and simulation of wireless and mobile systems (2003)

An Optimal Coverage Scheme for Wireless Sensor Network

Hui Tian and Hong Shen

Graduate School of Information Science,
Japan Advanced Institute of Science and Technology,
{hui-t, shen}@jaist.ac.jp

Abstract. The coverage problem is one of the most fundamental issues in a wireless sensor network, which directly affects the capability and efficiency of the sensor network. In this paper, we formulate this problem as a construction problem to find a topology that covers the required sensing area with high reliability. Deploying a good topology is also beneficial to management and energy saving. We propose an optimal coverage scheme for wireless sensor networks that can maintain sufficient sensing area as well as provide high reliability and long system lifetime, which is the main design challenge in sensor networks. With the same number of sensors, our scheme compares favorably with the existing schemes.

Keywords: sensors, coverage, hexagon, reliability, energy.

1 Introduction

The advancement in wireless communication and sensor technology is expediting the development of wireless sensor networks (WSNs), which have a wide range of environmental sensing applications such as danger alarm, vehicle track, battle field surveillance, habitat monitor, etc. [1, 3]. A WSN consists of hundreds to thousands of sensors and a base station. To gather information from the environment and deliver the processed messages to the base station, each sensor is capable of collecting, storing, processing signal, and communicating with neighbors. The base station decides if an unusual or concerned event occurs in the sensing area after aggregation and analysis of the messages from the sensors.

Similar to mobile ad-hoc networks (MANET), WSNs apply multi-hop communications where the packets sent by the source node are relayed by several intermediate nodes before reaching the destination node. However, they are significantly different in several aspects. First, the communication mode of a WSN is mainly many-to-one, that is, multiple sensor nodes send data to a base station or aggregation point in the network, whereas MANET support communication between any pair of nodes. Second, unlike MANET, data collected by different sensor nodes in a WSN might be the same and needs to be processed in the intermediate nodes. Third, in most envisioned scenarios the sensor nodes are immobile and keep on sensing and monitoring the area assigned beforehand until the system energy is exhausted. Finally, the energy constraint is much more

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 722–730, 2005.

stringent in WSN than that in MANET because the communication devices handled by human users can be replaced or recharged relatively often in MANET, whereas battery recharging or replacement is usually infeasible for a WSN because it's often deployed in hostile or inhospitable places. Thus, maintenance of unattended sensor nodes in a WSN to lengthen the system lifetime becomes extremely important when deploying the WSN. All of these properties of sensor networks, in turn, highlight three goals in designing WSN: energy efficiency, high reliability and low latency.

Topology management and priori topology deployment, if available, are greatly beneficial to reach above goals. Appropriate location of sensor nodes is helpful to save energy as well as keep required reliability which are always contradictory in the previous designs [8]. In this paper we propose an optimal scheme for topology deployment driven by the following concerns. First, a well-designed topology may bring the most desirable effectiveness to a WSN because the sensor nodes are usually immovable after deployment. A well-deployed WSN can benefit to energy saving by avoiding redundancy due to unnecessary overlapping of sensor nodes. Second, to ensure there isn't any "sensing hole", i.e., "blind points", we hope to design a flexible topology which can cover the entire area. Third, we want to configure the sensing area with the required reliability because some abnormal event requires several sensors to sense collaboratively so that guarantee high reliability. By deploying sensors according to an appropriate coverage scheme, we can provide high reliability to the entire sensor network or only the hot spots where the abnormal events may happen frequently. Fourth, we want to manage a WSN more easily. Nodes in WSNs are too constrained in memory and computing resources to afford for complex protocols. The knowledge of topology will simplify its management. Thus, we propose an optimal coverage scheme for WSNs to obtain the expected outcomes.

The coverage problem was stated as a decision problem in [4], which determined if each point in a WSN was covered by at least k-sensor to avoid the redundancy. They tried to calculate how many sensors overlapped with the concerned sensor. It involved complex communication between the sensor and all its neighbors and indirect-neighbors that are beyond the communication range of the sensor though overlapping with it. So heavy communication cost affected its potential applications in practice. However, the knowledge of coverage in WSNs does benefit to energy saving and energy efficient routing as discussed in [6, 7, 8]. Thus we study on how to deploy WSNs in a desirable way and propose an optimal coverage scheme which can reach both goals of energy saving and high reliability. It applies hexagon-based coverage as cellular network which has ever been a landmark novation for mobile communications. Note that unlike discussed in [2], some scenarios preclude the possibility of manual deployment and configuration when building them. We consider the cases that the sensors can be deployed and configured at the start which are required in many applications such as danger alarm, vehicle tracking. Our adaptive reliability design on the knowledge of topology will be applicable to environmental dynamics.

The paper is organized as follows. In Section 2 the problem is stated. Section 3 analyzes different coverage schemes and proposes our optimal coverage scheme. Section 4 discusses the reliability and energy problem in our scheme. Section 5 presents all possible applications in energy saving and reliability design by applying our coverage scheme. Section 6 concludes the paper.

2 Problem Statement

Traditional deployment for WSNs is self-organizing neighboring discovery on random located sensor nodes. The topology information is obtained by periodic communications. However, deploying different topologies at the start affects the nature of sensor networks severely. If the topology is well designed and controlled, the main issues in WSNs which are energy and reliability constrained can be solved. So we formulate the problem as a construction problem which is how to deploy a topology for a WSN that can effectively cover the required area and provide high reliability and energy efficiency. In this section, we use a mathematical model to state the problem.

We assume that a WSN consists of N sensor nodes and 1 base station. Each sensor node has the same sensing and communication range within a disk of radius r. If all nodes can communicate with the base station, we denote the coverage area of this sensor network by S_N. No matter what happens in the area S_N, the responsible sensor node can sense the event and report to the base station via other intermediate sensor nodes. If each point in S_N is sensed by at least k sensor nodes, we define the sensor network as a system with k-reliability.

Thus all the concerned problems are: 1. How can we design a topology with maximal S_N by using N sensor nodes? 2. In a deployed topology, how should we configure it as a k-reliability system and guarantee k-reliability in hot spots? 3. How to save energy and lengthen the network lifetime in our coverage scheme?

3 Sensor Networks Coverage Scheme

In this section, we study the sensing area covered by sensor networks in different topologies with the same given number of sensor nodes N. To reach this goal, we begin with analysis on additional sensing area provided by a sensor node which is enlightened by analysis on rebroadcast beneficial area in [5].

3.1 Analysis on Additional Sensing Area

We define the new sensing area provided by adding a sensor node as the additional sensing area of this node. The shadow area in Figure 1 is the additional sensing area of node j. We denote it by S_{A_j}. Let d be the distance between nodes i and j. Assume the sensing area of sensor node i to be S_i. Thus we can derive $S_{A_j} = S_j - S_i \cap S_j = \pi r^2 - INTS(d)$, where $INTS(d)$ is the intersection area of two circles covered by two nodes whose distance is d.

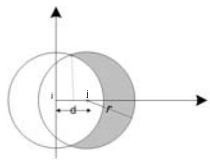


Fig. 1. Additional area provided by sensor node j

$$INTS(d) = 4 \int_{d/2}^{r} \sqrt{r^2 - x^2} dx \tag{1}$$

When $d > r$, i and j cannot communicate with each other. If either i or j doesn't have any else neighbor nodes, the sensor node will be isolated so that it cannot report any sensed event to the base station. Thus, each sensor node in the network must keep at least one neighbor whose distance from it is no more than r as Lemma 1 will give. Even if the additional sensing area provided by j if $d > r$ ($d \leq 2r$) would be great, i and j need the third sensor node to cover both of them, which in turn, results in the actual additional area of j being the additional area under the condition $d \leq r$. Therefore, the additional area of j is studied under $d \leq r$. When $d = r$, the additional area S_{A_j} is the largest, which equals $\pi r^2 - INTS(d) = r^2(\frac{\pi}{3} + \frac{\sqrt{3}}{2}) \approx 0.61\pi r^2$.

Lemma 1. *In order for a WSN to sense any abnormal event in its covered area, each sensor node must have at least one neighbor node whose distance from it is no more than its communication range, i.e.* $d \leq r$.

We then work on how to deploy sensor nodes under the above constraint and cover as large area as possible which may be in any shape. The simplest way is deploying sensor nodes in a linear array, but it is only limited to a strip area to be covered. We have to find a general approach to cover the area in any shape.

As we have derived, the additional sensing area by deploying a new sensor node is maximized when $d = r$ under the condition of their communication availability. In a linear array-deployed network, every two sensor nodes share a sensor node with distance r to maintain communication. A sensor node can have three or more neighbors who communicate via it. To maximize the coverage while minimize the number of sensor nodes, deploying three neighbors who communicate via the same sensor node is the optimal scheme because the arc covered by a neighbor with largest additional area is $2\pi/3$. Thus we obtain,

Lemma 2. *The coverage of a sensor network would be optimized when a sensor node support three neighbors to communicate with each other.*

3.2 Analysis on Hexagon-Based Topology

In the cellular network, it has been proved that a hexagon-based topology is the best topology due to its provision of multiple non-overlapping equal cells and approximation to a circle. Though the WSN, unlike the cellular network, doesn't require to consider the frequency reuse policy, the coverage scheme is applicable. We now deploy a hexagon-based topology for WSNs, which combining with Lemma 1 and 2(also demonstration for these lemmas), would turn out an ideal model. All other possible topologies, triangle-based, quadrangle-based, will be compared with hexagon-based topology.

Figure 2(a) gives a WSN deployed with a 2-layer hexagon-based topology. We denote the cells with solid line to be base cells, and denote the cells with dashed line to be connect cells. Obviously, base cells provide the sensing area, while connect cells are deployed to maintain effective communication of the base cells, i.e., the whole sensor network. Each connect cell right supports three neighbors (base cells) to communicate via it as Lemma 2 shows. Such topology can extend randomly in 2-dimension plane to cover a required area, which is unlimited as linear array deployment that can only extend in 1-dimension.

We note that there are overlapped area between two neighbors. Figure 2(b) describes the real coverage area of each base cell and overlapped area. In Section 2, we have denoted the real coverage area covered by the whole network by S_N, where N is the number of sensor nodes. For comparison, we will study S_N of a WSN in all possible regular topologies consisting of 25 sensor nodes, which are triangle-based, quadrangle-based and hexagon-based topologies.

Denote S_o to be the overlapped area of two base cells in Figure 2(b).

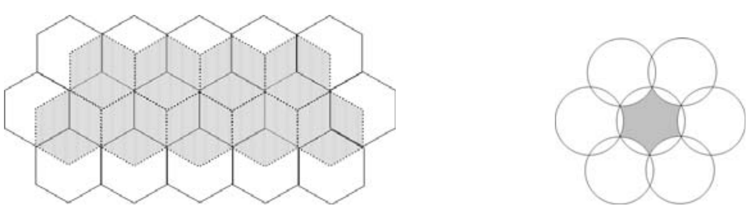
$$S_o = 4\int_{\frac{\sqrt{3}r}{2}}^{r} \sqrt{r^2 - x^2}dx \approx 0.180r^2. \tag{2}$$

There are 16 base cells and 9 connect cells in the sensor work. Let $S_N^{\odot}$ denote the real coverage area in hexagon-based topology. Thus,

$$S_N^{\odot} = 16 \cdot (\pi r^2 - 6S_o) \approx 32.992r^2. \tag{3}$$

Similarly, the real coverage area in triangle, quadrangle based topologies are:

$$S_N^{\square} = 16 \cdot (\pi r^2 - 4 \cdot 4\int_{\frac{\sqrt{2}r}{2}}^{r} \sqrt{r^2 - x^2}dx) \approx 13.728r^2, \tag{4}$$



(a) The 2-layer cellular topology

(b) A cell's real coverage area

Fig. 2. A hexagon-based sensor network

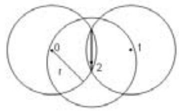


Fig. 3. Node 0 has two perimeter-overlapped neighbors

$$S_N^{\triangle} = S_N^{\odot} \approx 32.992r^2, \tag{5}$$

where $S_N^{\square}$ and $S_N^{\triangle}$ denote the real coverage area of triangle-based and quadrangle-based topologies respectively.

Deployment of triangle-based topology is essentially the same as hexagon-based topology. Due to the hexagon cell is the closest shape to a circle, and the hexagonal cell shape is a simplistic model of the coverage for each sensor node, we deploy the area required to be covered by hexagon cells. The above analysis has proved that hexagon-based topology can obtain much better performance in coverage than quadrangle-based topology. We will compare the hexagon-based coverage scheme with a randomly-deployed scheme in the succeeding section.

3.3 Analysis on Random-Deployed Topology

The WSN deployed the sensor nodes randomly in the previous work. The real coverage area by randomly deployed sensor nodes differs from that in hexagon-based or quadrangle-based topologies. We now have a look at the real coverage area by randomly deploying 25 sensor nodes.

Lemma 3. *To maintain effective communication of the whole sensor network, a sensor node must have one neighbor whose distance is less than r, or two neighbors where one is put anywhere with the distance $r < d \leq 2r$, the other is located on the cross line as Figure 3.*

All the nodes are deployed according to the rule in Lemma 3. We then consider the deployment is performed as the following way. A sensor node is firstly deployed in the required service area, then the first neighbor who provides the additional coverage area S_{A_1} is added, whereafter the second neighbor who provides S_{A_2} is added according to Lemma 3. The rest sensor nodes are deployed in turn as the first neighbor and second neighbor respectively. Considering there must be overlapped coverage area between these first-neighbors and second-neighbors, we have the following inequality,

$$S_N < \pi r^2 + 12S_{A_1} + 12S_{A_2}. \tag{6}$$

Then the expected coverage area with 25 randomly-deployed sensors satisfies

$$E[S_N] < \pi r^2 + 12E[S_{A_1}] + 12E[S_{A_2}]. \tag{7}$$

Because the expected additional coverage area provided by the first neighbor can be derived as

$$E[S_{A_1}] = \int_0^{2r} \frac{2\pi x[\pi r^2 - INTS(x)]}{4\pi r^2} dx \approx 0.591r^2, \tag{8}$$

where the probability of the first neighbor whose distance is x from the original sensor node is $\frac{2\pi x}{\pi(2r)^2}$ because in the area of the circle of radius $2r$, the sensor node can only locate at the perimeter of the circle of radius x for x in $[0, 2r]$.

The expected additional coverage area provided by the second neighbor is

$$E[S_{A2}] \approx 0.19r^2. \tag{9}$$

The derivation for equation (9) can refer to [5]. Thus, we get a upper bound for the expected coverage area with 25 randomly-deployed sensor nodes.

$$E[S_{25}] < 32.589r^2 \tag{10}$$

Comparing (10) with (3), we find that the randomly deployed sensor network provides smaller coverage area than hexagon-based sensor network.

4 Solution to Reliability and Energy Constraints

In order for a WSN to sense important events, it should work with high reliability and as long lifetime as possible. In this section, both goals can be reached in a WSN deployed by our optimal coverage scheme.

As we have defined, if each point in the WSN is covered by at least k sensor nodes, we call it k-reliability sensor network. In a sensor network with hexagon-based topology, we find that each point in the service area is covered by 2 sensors except the marginal place. Without doubt, such a WSN provides higher reliability than a randomly deployed network where much area might be covered by only 1 sensor. If the marginal place isn't taken into account, the sensor network with hexagon-based topology is a 2-reliability sensor network. There are some area covered by 3 sensors, but the area is very small because the overlapped area between cells is small. Thus, in a WSN which is required to be 2-reliability, a hexagon-based topology can not only meet the demand, but also use the minimum number of sensors due to low redundancy.

In some cases, the whole system need to be k-reliable, where $k > 2$. In the hexagon-based sensor network, we can deploy more than one sensor node in the center of each base cell (not in connect cell) according to the required value of k. All these sensor nodes sense the area together to avoid any failure in sensing important events. Thus, the system is easily configured as a k-reliable sensor network. However, because the system in our hexagon-based topology is 2-reliable, keeping one sensor node in each cell is enough in most scenarios. To configure the higher reliability in hot spots, we can add more sensor nodes in the base cells which cover these area.

The approach to save energy and lengthen the lifetime in a hexagon-based WSN is a little different from reliability configuration. We locate more than one sensor in each cell, including base cells and connect cells while keep only one sensor alive. Once a sensor is going to use out its battery, the other sensor in the same cell is waken up. Due to impossibility of recharging the sensors, this kind of configuration can obtain longer lifetime for WSNs.

If quadrangle or random topologies are used to cover the same area with the same lifetime, we have to involve a larger number of sensors. From the energy point of view, each sensor is set to a power which can reach the other sensors with distance of r in a hexagon-based WSN. When deploying them randomly, instead, each node may not need such a power because smaller distance than r between sensors may exist. So unique power configuration may cause redundant energy cost, extra power saving routing and management is necessary. In hexagon-based WSN, simple management and energy saving is reached.

5 Applications of the Hexagon-Based Coverage Scheme

A hexagon-based WSN can be applied in many cases where foreseeable deployment is possible. It provides the following advantages and allows potential applications on these natures.

1. The deployment of each sensor node in hexagon-based topology maximizes the additional sensing area. Randomly deployed sensor nodes cannot guarantee the maximal additional sensing area. To cover the same area, hexagon-based topology needs less sensor nodes than any other kind of topology.
2. There is no blind point in the hexagon-based sensor network, i.e., all the events in the service area can be sensed. Moreover, each event can be sensed by at least 2 sensor nodes due to 2-reliability of the sensor network.
3. It is easy to deploy a wireless sensor network or only those hot spots area with higher reliability than 2-reliability.
4. The system life time can be longer by setting more sensors in each cell. Energy saving can be performed by keeping one sensor node alive in each cell.
5. The power of each sensor node can be strictly set to a certain value according to the required radius of cell, where it is estimated by $P_r \propto \frac{P_t}{r^K}$. Here K is an experience value, usually, $K = 3$. In this formula, the transmitting power P_t can be determined if the required receiving power P_r and the radius of cell r are given. Therefore the redundant power consumption is avoided.
6. The sensors in connect cells can be turned off for saving more energy because the area covered by them have been covered by base cells. Once some important event is sensed and needs to be transferred to the base station, the relevant sensors in connect cells can be waked to act as the routing nodes.
7. Node degree of each sensor is balanced to be 3 in hexagon-based WSN. So congestion and delay caused in a WSN with complex topology may be avoided.
8. The simple routing can be designed for the typical communication mode — data aggregation from many to one in a hexagon-based sensor network.

6 Conclusion

A hexagon-based coverage scheme has been proposed for WSNs in this paper. It can be applied in many scenarios where the required service area can be deployed at the start. By analysis we show that WSNs can benefit from the hexagon-based topology in coverage area, energy saving, reliability control, routing design etc. The potential applications have been discussed, which provides a challenging design to the traditional WSNs where energy saving and reliability are the most significant. Thus our coverage scheme is promising and provides a new view to coverage problem in WSNs.

Acknowledgement

This work was supported by the 21st Century COE Program of Ministry of Education, Culture, Sports, Science and Technology, and by Japan Society for the Promotion of Science (JSPS) Grant-in-Aid for Scientific Research under its General Research Scheme (**B**) Grant No. 14380139.

References

[1] I.F. Akyildiz, Y. Sankarasubramaniam W. Su, and E. Cayirci. Wireless sensor networks: A survey. *Computer Networks*, 2002.
[2] Alberto Cerpa and Deborah Estrin. Ascent: Adaptive self-configuring sensor networks topologies. In *Pro. of INFOCOM*, 2002.
[3] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar. Next century challenges: Scalable coordination in sensor networks. In *Pro. of ACM MobiCom*, 1999.
[4] Chi-Fu Huang and Yu-Chee Tseng. The coverage problem in a wireless sensor network. In *Pro. of ACM Int'l Workshop on Wireless Sensor Networks and Applications(WSNA)*, 2003.
[5] Sze-Yao Ni, Yu-Chee Tseng, Yuh-Shyan Chen, and Jang-Ping Sheu. The broadcast storm problem in a mobile ad hoc network. In *Pro. of MobiCOM*, 1999.
[6] Curt Schurgers and Mani B. Srivastava. Energy efficient routing in wireless sensor networks. In *Pro. of MobiHOC*, 2002.
[7] Curt Schurgers, Vlasios Tsiatsis, Saurabh Ganeriwal, and Mani B. Srivastava. Topology management for sensor networks: Exploiting latency and density. In *Pro. of MobiCOM*, 2001.
[8] Di Tian and Nicolas D. Georgannas. A coverage-preserving node scheduling scheme for large wireless sensor networks. In Pro. of ACM *Int'l Workshop on Wireless Sensor Networks and Applications(WSNA)*, 2002.

Routing Protocols Based on Super Cluster Header in Wireless Sensor Network

Jae-hwan Noh, Byeong-jik Lee, Nam-koo Ha, and Ki-jun Han*

Department of Computer Engineering, Kyungpook National University
{ambitions, leric, adama2}@netopia.knu.ac.kr
kjhan@bh.knu.ac.kr

Abstract. In a variety of applications, wireless sensor networks have received more attention in recent years. Sensor nodes, however, have many limitations including limited battery power and communication range. In this network, data gathering and data fusion help to reduce energy consumption and redundant data. LEACH (Low Energy Adoptive Clustering Hierarchy) is the most representative protocol using data gathering and data fusion, but it has several problems including inefficient energy consumption by many cluster headers to the distant sink and a single-hop routing path. In this paper, we propose two routing protocols called Routing Protocol based on Super-Cluster Header (RPS) and Multi-hop Routing Protocol based on Super-Cluster Header (MRPS) in order to resolve the problems of LEACH. The key idea of our protocols is that only one node sends the combined data to the sink and every node uses multi-hop routing in order to gather data in the cluster. We evaluate performance of our protocols through simulations. Simulation results show that our protocols offer a much better performance than the legacy protocols in terms of energy cost, the network lifetime, and fairness of the energy consumption.

1 Introduction

Wireless micro-sensor networks are expected to have a significant impact on the efficiency a variety of applications that include surveillance, machine failure diagnosis, and chemical, biological detection, since advances in sensor technology, low power electronics, and low-power RF (Radio Frequency) design have led to the development of micro-sensors [1-3]. These sensor networks are, however, such that node's power, computational capacity, memory, and communication bandwidth are significantly more limited than the traditional wireless ad hoc networks. The main aim of routing in these sensor networks is to find ways for energy-efficient route setup and the reliable relaying of data from the sensor nodes to the SINK (is similar to Base Station). It is a very important to use the available bandwidth and energy efficiently so that the lifetime of the network is maximized [2]. Data fusion and gathering help achieve these aims [5-6]. LEACH is a suitable solution that uses data fusion and gathering but, it has several problems. One problem is that average five percent of

* Correspondent author.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 731–739, 2005.

nodes transmit the fused data from cluster to the distant SINK. Another issue is that it uses a single-hop routing path.

In this paper, we propose new two protocols called the RPS (Routing Protocol based on Super-Cluster Header), and the MRPS (Multi-hop Routing Protocol based on Super-Cluster Header). The key idea of our protocol is that only one designated cluster header node, which is defined as a Super Cluster Header, sends the combined data to the sink and every node uses multi-hop routing in order to gather data in clusters. Therefore, our protocols can reduce energy cost significantly and increase the life time of the sensor network.

The remainder of this paper is organized as follows: In Section 2, we review the LEACH protocol. In Section 3, we present our protocols. Section 4 contains the performance evaluation of our protocol through simulations. Finally, Section 5 is the conclusion.

2 LEACH

In the sensor network, all data sensed from the nodes have to be collected and sent to a distant SINK, where the end-user can access the data. A simple approach to accomplishing this task is for each node to transmit its data directly to the SINK. Since the sink is typically located far away and the energy cost is proportional to the distance in transmission, the cost for transmission to the sink from any node is high; therefore the nodes will die very quickly. In addition, the SINK receives redundant data which may be unnecessary. Data fusion and data gathering tries as few transmissions as possible to the SINK to reduce the amount of data that must be transmitted to the SINK and redundant data [5-6]. Furthermore, if all nodes in the network deplete their energy levels uniformly, then the network can operate without losing any nodes for a long time.

LEACH [1-2, 7-8] is one of the most popular hierarchical routing algorithms for these approaches in sensor networks. In LEACH, since a small number of clusters are formed in a self-organized manner, it is a suitable solution for energy efficiency in the sensor network. One nice property of LEACH is that it is completely distributed and the sensor nodes in each cluster are organized to fuse their data, eventually transferring it to the SINK without global knowledge of the network. A designated node in each cluster collects and fuses data from the nodes in its cluster and transmits the result to the SINK. It uses randomization to rotate the cluster headers. Therefore, the nodes die randomly, the lifetime of the system increases and the energy consumption level is fair. Although, LEACH is a sound solution in gathering data, it does have LEACH have several problems:

- Clusters formed randomly in each round not only may not produce good clusters to be efficient but also may be no cluster formation.
- The signal overhead cost for forming the clusters is expensive. In every round, average five percent nodes act as ‘Cluster Headers (CHs)’ [1-2, 7-8] and these nodes must broadcast a signal to all nodes to determine their cluster members.

▪ Average five percent nodes (CHs) of nodes transmit the fused data from the cluster to the distant SINK [7-8]. If only one node transmits the fused data to the distant SINK per round, the energy cost will be greatly reduced [10].
▪ LEACH uses single-hop routing where each node can transmit directly to the CH and then the CH can transmit directly to the SINK. Therefore, it is not applicable for networks deployed in large regions.

3 Super Cluster Header Routing Protocol

As previously described, LEACH has some problems. Therefore, in this section, we present two new protocols in order to solve these problems. The protocols optimize to energy cost when gathering data. In addition, it distributes energy consumption fairly. Our protocols are based on the following assumptions:

▪ Every sensor node has power control and the ability to transmit data to any other sensor node or to the SINK directly [4].
▪ Every node has location information and there is no mobility.

These assumptions are reasonable due to technological advances in radio hardware and low-power computing.

3.1 RPS: Routing Protocol Based on Super Cluster Header

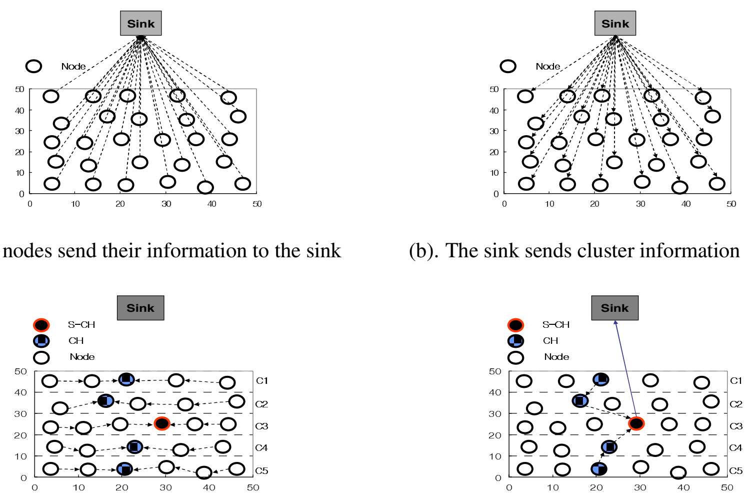
The key idea of the RPS is that only one node which is defined as a 'Super-Cluster Header (SCH)', sends the combined data to the SINK. Therefore, the RPS can significantly reduce energy cost and increase the life of the sensor network. The RPS is similar to operate as LEACH. We will only use the energy level information of the CH (Cluster Header) and the node ID. When selected the CHs broadcast advertisement messages including energy level information and node ID to the rest of the nodes, Each CH compare itself to the energy level information of other headers in order to select only one node which is defined as a SCH.

The CH which has the most powerful energy level is selected as the SCH. If the energy level of the CH is the same, the CH with the lowest node ID will be selected as the SCH. These operations don't require additional overhead when being compared with LEACH.

3.2 MRPS: Multi-hop Routing Protocol Based on Super Cluster Header

Although the RPS is more efficient than LEACH in terms of cost by using the SCH, it still has problem using a single-hop routing path. For energy calculation in a sensor network, the transmission distance is a very important factor. In this aspect, using a multi-hop routing path is very efficient in that reducing energy cost in a sensor network. We present new protocol called the MRPS. It is only one node (SCH) sends the combined data to the SINK and every node uses a multi-hop routing path instead of single-hop routing path used in LEACH and the RPS. Therefore, the MRPS can significantly reduce energy cost and increase the life time of the sensor network. The operation of MRPS is as follows.

Initially, we randomly place the nodes in the playing field. Each node directly sends information about its location and energy level to the SINK only one time after the initial placing of the nodes. Fig.1.(a) shows this operation. After the SINK receives this information, it makes cluster information which consists of a header of each cluster, an SCH and cluster ID. It investigates the energy level of nodes to find the CHs. The node which has the highest energy in each cluster is selected as the CH. Among the CHs, the most powerful header is selected as a SCH. At this time, the cluster ID is determined in such a way that the number of groups will be five percent of nodes, since average five percent of nodes are a good choice for efficient data gathering in sensor network [1, 7-8]. Following this, the SINK makes an advertisement message, which consists of cluster information, and then it broadcasts this message to all nodes. Fig.1.(b) shows this operation. In this way, each node knows which cluster it belongs to and its own CH. In addition, all CHs can determine the SCH.



(a). All nodes send their information to the sink

(b). The sink sends cluster information to nodes

(c). Data gathering in each cluster

(d). After data gathering from each CH to the SCH, the SCH sends all information to the sink

Fig. 1. Multi-hop routing through Super Cluster Header

This operation is performed every round to maintain the affair energy consumption level. We can employ a time slot approach, since all nodes know their positions and group information. Since radio is inherently a broadcast medium, transmission in one cluster will affect communication in a nearby cluster. To reduce this type of interference, each cluster uses different CDMA codes. Each cluster can operate a time slot approach separately in each round, due to CDMA [7-8].

Fig.1.(c) shows data gathering in each cluster which uses the CDMA codes. The node which is scheduled by the time slot in each cluster, receives data from the previous node on the path, fuses the received data and its own data with its energy level information, and transmits the fused data to the next node. As mentioned above, energy level information is utilized by the SINK to make an advertisement message.

Fig.1.(d) shows data gathering from the non-SCH to the SCH. The non-SCH gathers all of the information in its own cluster and sends the gathered information and its own information to the SCH through the transmission path which uses a multi-hop. At this time, if the distance of the transmission path is farther away than the direct distance to the sink, the CH does not send information to the next hop thorough the transmission path, but sends it directly to the sink. For example, if the distance between the CH_1 and the CH_2 is farther away than the distance between the CH_1 and the sink, the CH_1 does not send information to the CH_2, but sends it directly to the sink. When the SCH gathers all of the information, it transmits the information to the sink and then one round is finished.

4 Simulation Results

In this section, we evaluate the effectiveness of our protocol through simulations. For simulations, we use a radio model for energy in the sensor network. This is the same radio model as discussed in LEACH, which is the first order radio model [7-8]. In this model, a radio dissipates $E_{ELEC} = 50nJ/bit$ to run the transmitter or receiver circuitry and $E_{AMP} = 100pJ/bit/m^2$ for the transmitter amplifier. There is also a cost of *5nJ/bit/message* for a 2000bit messages in data fusion [10]. The radios have power control and can expend the minimum required amount of energy to reach the intended recipients. The radios can be turned off to avoid receiving unintended transmissions. An r^2 energy loss is incurred due to the channel transmission [9]. The following equations show radio transmission costs and radio receiving costs for a k-bit message at a distance d. Equation (1) and (2) are used to obtain the transmission cost and the receiving cost, respectively.

$$E_{TX}(k,d) = E_{ELEC} * k + E_{AMP} * k * d^2 \quad (1)$$

$$E_{TX}(k,d) = E_{ELEC} * k \quad (2)$$

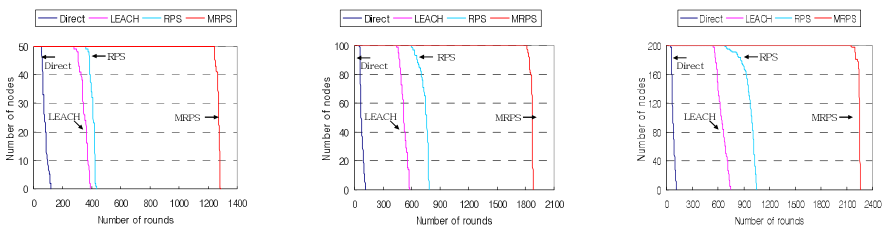
We make the assumption that the radio channel is symmetric. For our experiments, we also assume that all sensors are sensing the environment at a fixed rate and thus, they always have data to send to the end-user. We introduce some parameters for performance evaluation as shown in Table 1. Simulations are carried out in different network topologies. In each network topology, the N nodes are randomly scattered in a fixed area. The distance between the SINK and any one node is not less than 100m. The packet size is fixed. We assume that all nodes have the same initial energy level.

First, we evaluate the network life time by examining the number of rounds until all nodes die. For this simulation, both packet size and initial energy level are fixed at 2000 bits, and 0.25J, respectively. Fig. 2 ~ Fig. 4 show that our proposed protocols offer a much longer life time than LEACH or direct transmission. Here, direct transmission means that each node transmits its data directly to the distant SINK.

Table 1. Simulation parameters

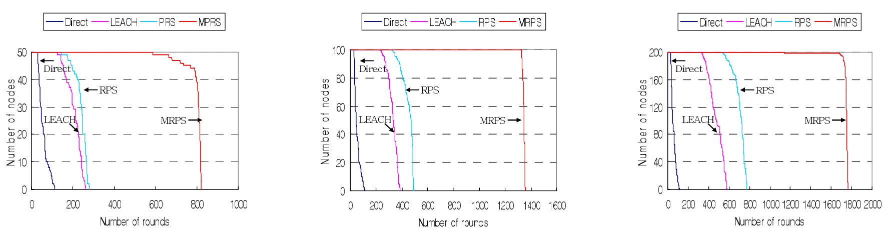
The number of nodes	50, 100, 200
The size of the network	50m x 50m, 100m x 100m, 200m x 200m
Packet size	2000bit, 5000bit, 10000bit
The location of the SINK	(25,150) , (50, 200), (100, 300)
Initial energy level	0.25J, 05J, 1J

In particular, the MRPS is better than other protocols in terms of fairness of energy consumption, since the rounds of the MRPS achieved until the first node are much longer than that of direct transmission, LEACH or the RPS.



(a). The number of nodes is 50 (b). The number of nodes is 100 (c). The number of nodes is 200

Fig. 2. Life time for a 50m x 50m network when the SINK is located at (25, 150)

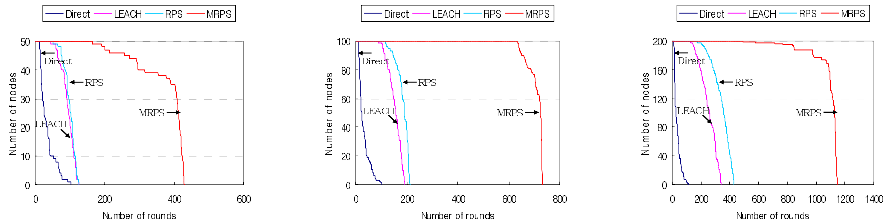


(a). The number of node is 50 (b). The number of node is 100 (c). The number of node is 200

Fig. 3. Life time for a 100m x 100m network when the SINK is located at (50, 200)

Again, we evaluate the life times of the sensor network in another way. We investigate the life time when different energy levels are given to the nodes initially. The size of the network is 100m x 100 m, the location of the sink is (50,200), the number of nodes is 100 and the packet size is 2000 bits. We summarize the results in Table 2. We can see that the rounds of the MRPS achieved until the first node and the last node die are longer than those of the LEACH and the RPS. More specifically, the

MRPS offers a longer life time than LEACH by approximately 6 times until the first node died and by approximately 4 times longer until the last node died.

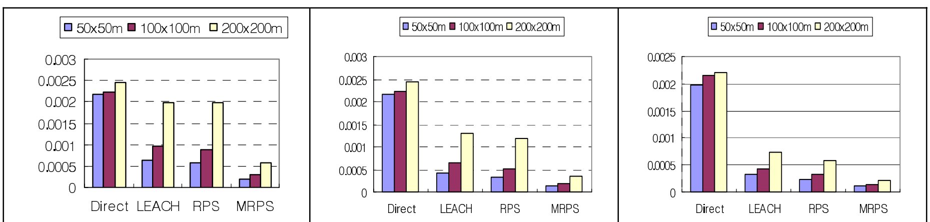


(a). The number of node is 50 (b). The number of node is 100 (c). The number of node is 200

Fig. 4. Life time for a 200m x 200m network when the SINK is located at (100, 300)

Table 2. Life times when different energy levels are given to the nodes initially

Energy(J)	Protocol	Number of rounds until the first node dies	Number of rounds until t he last node dies
0.25 J	Direct	33	116
	LEACH	235	387
	RPS	330	495
	MRPS	1327	1360
0.5 J	Direct	61	226
	LEACH	489	781
	RPS	669	1009
	MRPS	2621	2704
1 J	Direct	122	452
	LEACH	1051	1592
	RPS	1375	2034
	MRPS	5358	5412



(a). The number of node is 50 (a). The number of node is 100 (a). The number of node is 200

Fig. 5. Values of N_{ADE} when the initial energy level is 0.25J and the packet size is 2000 bits

We investigate the N_{ADE} defined as the amount of the average depleted energy of each node per each round. The N_{ADE} is given by

$$\frac{E_{INI}.N}{\mathrm{N}_{\mathrm{ADE}}.N} = R_{TOTAL} \tag{3}$$

$$\mathrm{N}_{\mathrm{ADE}} = \frac{E_{INI}}{R_{TOTAL}} \tag{4}$$

where E_{INI} is the initial energy level of nodes, N is the number of nodes and the R_{TOTAL} is the number of rounds achieved until the last node dies. Fig. 5 shows the values of N_{ADE} obtained through simulations. From this graph, we can see that the higher the density is, the better performance our proposed protocols provide.

5 Conclusion

In this paper, we propose new two routing protocols using 'Super-Cluster Header (SCH)' for efficient data gathering in a sensor network. In the proposed routing protocols, only one node (SCH) sends the combined data to the SINK and a multi-hop routing path is used.

Simulation results show that our protocols offer a much better performance than LEACH or direct transmission in terms of the energy cost, the life time of the sensor network and fairness of the energy consumption. Further more; our protocols are suited for a sensor network with high density. In our future work, we will study an efficient energy dissipation algorithm through data gathering in a mobility sensor network.

Acknowledgement

Academic Research Program supported by Ministry of Information and Communication in Republic of Korea.

References

[1] I.F. Akyildiz et al., "Wireless sensor networks: a survey", *Computer Networks* 38 (4) (2002) 393–422. K.

[2] Akkaya and M. Younis, "A Survey of Routing Protocols in Wireless Sensor Networks, " *in the Elsevier Ad Hoc Network Journal,* Sep, 2003

[3] L. Blazevic, L. Buttyan, S. Capkun, S. Giordano, J.-P. Hubaux, and J.-Y.L. Boudec, "Self-organization in mobile ad hoc networks: The approach of 'Terminodes'," *IEEE Commun. Mag., vol. 39*, pp. 166–174, June.2001

[4] R.Ramanathan and R. Hain, "Topology Control of Multi hop Wireless Networks Using Transmit Power Adjustment,"*In Proceedings Infocom 2000*,2000

[5] W.R. Heinzelman, A.C. Chandrakasan and H. Balakrishnan, "Energy Efficient Communication Protocol for Wireless Micro sensor Network." *In Proceedings of the IEEE Hawaii International Conference on System Sciences*, Jan, 2000.

[6] Wei Yuan, S.V.Krishnamurthy, S.K.Tripathi, "Synchronization of multiple levels of data fusion in wireless sensor networks." *Global Telecommunications Conference, 2003. GLOBECOM2003*, Dec, 2003.

[7] W.Heinzelman, A. Chandrakasan, H. Balakrishnan, "Energy-Efficient communication protocol for wireless sensor networks", in: *Proceeding of the Hawaii International Conference System Sciences*, Hawaii, January 2000.
[8] Heinzelman, W.B.; Chandrakasan, A.P.; Balakrishnan, H.; " An Application-Specific Protocol Architectures for Wireless Microsensor Networks", *Communications, IEEE Transactions* on , Volume: 1 , Issue: 4 , Oct. 2002
[9] T.S Rappaport, "Wireless Communications", *Prentice-Hall*, 1996
[10] Lindsey, S.; Raghavendra, C.; Sivalingam, K.M.; " Data gathering Algorithms in Sensor Networks Using Energy Metrics", *Parallel and Distributed Systems, IEEE Transactions* on , Volume: 13 , Issue: 9 , Sept. 2002

An Automatic and Generic Early-Bird System for Internet Backbone Based on Traffic Anomaly Detection

RongJie Gu[1], PuLiu Yan[1], Tao Zou[2], and Chengcheng Guo[1]

[1] Department of Electronic Information, WuHan University, 430072 WuHan, China
[2] Beijing Institute of System Engineering, 100101 Beijing, China
grj1116@hotmail.com,
{ypl, netccg}@whu.edu.cn, zoutao814@sina.com

Abstract. Worm and Dos, DDos attacks take place more and more frequently nowadays. It makes the internet security facing serious threat. In this paper, we introduced the algorithm and design of ESTABD, an internet backbone Early-bird System of Traffic Anomaly Detection Based. By observing the raw variables such as packets count of protocol, TCP flags and payload length distribution etc., ESTABD analyzes real-time traffic to discover the abrupt traffic anomalous and generate warnings. A traffic anomaly detection algorithm based on Statistic Prediction theory is put forward and the algorithm has been tested on real network data. Further more, Alerts correlation algorithm and system policy are addressed in this paper to detect the known worms& Dos attacks and potentially unknown threats.

1 Introduction

With the astonishingly rapid adoption of network computing and its e-Commerce derivatives, internet has already penetrated to every corner of modern society. Frequently exploded worms make the internet security facing serious threats. In July of 2001, worm Code-Red infected 250,000 computers in less than nine hours. [1] The direct economic loss came up to 2.6 billion dollars. January 2003, SQL Slammer worm caused a loss at 1.2 billion dollars in the first five minutes of bursting. Compared with the spreading rate of Code-Red worm which population doubled every 37 minutes, it only needs 8.5 seconds [1] [2]. The greedy nature of worms determines that most worms disseminate by the way of fast port scanning [3]. Many important services depending on network communication suffered a lot from the huge traffic jam. ATMs could not dispense money and flight could not take off just because of the paralyzed network communication. [4] [5] [6]. Besides worms, Denial of Service Attack, Distributed Dos Attack and traffic jams caused by unsuitable network installation, they are severely influencing the normal circulation of Internet as well. Traffic detection is one of the important tasks in network management and traffic anomaly detection on internet backbone is different than on others. It has following features: 1) traffic volume is too huge that it is impossible to process and store the detailed information; 2) detection algorithm should be compact and effective enough to meet the demand of realtime; 3) backbone has a large sample capacity and statistics theory can be applied to predict and measure the network behaviors.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 740–748, 2005.

2 Related Works

The network traffic anomaly has been widely studied in recent years. P. Barford et al. [7] apply signal analysis theories including Wavelet Analysis in traffic anomaly detection and give the analysis results of 4 classes of traffic anomaly. The detection method is complex and not quite suitable for real-time detection. A. Lakhina et al. [8] argued that it is not appropriate to model all connections simultaneously. They concentrate on the structure of Origin-Destination(OD) flows and decomposed the flow structure space into three main constituents by using of PCA method. But it is not an easy job to obtain OD flows in a real network. Observation can't be able to provide enough information that PCA method need to cover all situations. The deployment of Gbit or even 10Gbit signature-based IDS makes it possible to detect known worms on internet backbone. But its main drawback lies in: 1) high false positive; 2) passively depend on knowledge base. It can't obtain unknown abnormal knowledge about new threaten automatically. Some other researchers tried to make up this drawback. They put forward the idea of withdrawing frequently occurring string automatically from traffic to form characteristics of unknown worms (Madhusudan et al. [9]). But this will obviously influence the performance of system. Furthermore, there is no way to verify the accuracy of the signature. Throttan M. et al. [10] research on MIB variables got via SNMP protocol. MIB is easy to use but it can not provide more detailed information about traffic than protocol analysis from the raw traffic. We follow the latter.

3 Architecture Design

3.1 Introduction of Architecture Design of ESTABD

Fig. 1 demonstrates the main architecture design of ESTABD. The data collection platform is composed of many data-engines deployed in subnets of different ISPs and these data-engines collect traffic information according to policies received from the data center. The data-engines produce traffic data in two steps: a) protocol analyzing on traffic; b) generate the raw variables. The traffic raw variable is defined as following:

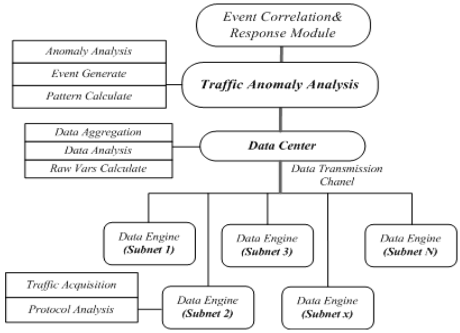


Fig. 1. The Architecture design of ESTABD

Definition 3.1 ***Raw variable***: The data-engines count the packet number of certain protocol variable in a fixed time interval after protocol analysis. This count is defined as ***raw variable*** including:

- ✧ Packet count variable of each protocol: count all packets of different ports(protocol)
- ✧ Packets length count variable of packet payload: We can divide the packet length into several intervals (1~100 bytes, 101~200 bytes… 1501~1600 bytes).
- ✧ Packet count variable of SYN and FIN flags: TCP SYN or FIN flag
- ✧ Packet count variable of ICMP unreachable messages: worm and many DoS will cause the abrupt of this value

The raw variables from all subnets are aggregated as total numbers based on same types and sent to the data center via a data transmission channel. Traffic anomaly detection module will check the incoming traffic time series. Alerts will be generated and sent to alert correlation module.

3.2 Event Definitions

This part we will introduce the event definition used in ESTABD. The traffic event is the anomalous result generate by anomaly detection algorithm. We divide traffic event into two categories, namely: *direct event* and *secondary event*.

Definition 3.2 The traffic anomalous, detected by anomaly detection module, as taken from the raw variable time sequence, is defined as the ***direct event***.

Definition 3.3 The traffic anomalous, detected by anomaly detection module, as taken from the new variable sequence which based on the algebra operation of raw variables (take for example: Difference, ratio operation) is defined as the ***secondary event***.

The behavior of worm-explosion can be observed from one or more direct events and secondary events. We can get the characteristics of the known or unknown anomalous by correlating these events.

3.3 Traffic Anomaly Detection Algorithm

As we discussed above, the traffic on backbone has a large population of hosts, the abrupt individual activity such as downloading a DVD via network will have no significant effect on the total traffic. The traffic variable series is continuous on time order, and has relationship between its history and future trend. The variable series will not be abrupt in usual time.

We have taken a long-period observation on an internet backbone continuously for 15 months and get a great deal of firsthand data and experiences. Based on our empirical knowledge, all kinds of traffic can be divided into two categories: Non-periodic traffic and Periodic traffic (Fig. 2).Non-periodic traffic is the most common type among all monitored traffic. Non-periodic traffic appears stationary (Fig.2-a) in usual time. Some traffic appears evidently periodic is for that some daily services such as HTTP, SMTP, FTP, have tight relationship with people's everyday life. The frequency of the service used presents a periodic change following human's work and rest timetable. The period is approximately 24 hours (Fig.2-b). Based on our observation of several completed periods, we can predict the data profile of the next period as Fig.2-c and define this profile as a *dynamic traffic pattern*. Then we compare this pattern with incoming traffic of next day, if the actual traffic volume of 9 o'clock am deviates from the value of 9 o'clock in pattern greater than a certain threshold, we say it is abnormal. The threshold will be given automatically by the detection algorithm.

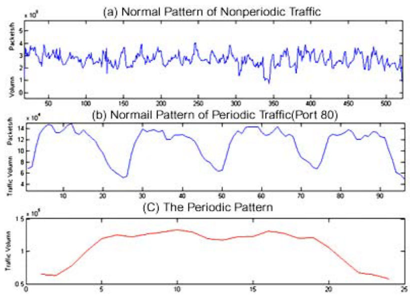


Fig. 2. Periodic and Non-periodic Traffic Pattern

The anomaly detection algorithm takes different method to process periodic traffic and non-periodic traffic. We use time series analysis and statistic prediction in detection algorithm.

3.3.1 Non-periodic Traffic: SESP (Single Exponential Smoothed Prediction)

SES (Single exponentially smoothing) is a very popular method to produce a smoothed Time Series. It evolves from the simple Moving Weighted Average Method [11]. Recent observations are given relatively more weight in forecasting than the older observations.

Let the time series be: $x_1, x_2 \ldots x_n$, equation (1) is the SES model. (S_n — the smoothed value of time $\boldsymbol{n}$, α —the smoothing constant., $\hat{x}_n$ -- the prediction value of time n) SESP, the prediction base on exponentially smoothing, its model shows as equation (2):

$$S_{n+1} = \alpha x_n + (1-\alpha) S_n \tag{1}$$

$$\hat{x}_{n+1} = S_{n+1} = \alpha x_n + (1-\alpha) S_n = \alpha x_n + (1-\alpha)\hat{x}_n \tag{2}$$

Error Measurement:

$$\text{MSE} = \frac{\sum_{t=1}^{n} e_t^2}{n} = \frac{\sum_{t=1}^{n} [x_t - \hat{x}_t]^2}{n} \quad \text{or} \quad \text{MAE} = \frac{\sum_{t=1}^{n} |x_t - s_{t-1}|}{n} \tag{3}$$

Here, we introduce a sliding time window method to calculate the allowable range (threshold) deviation from the prediction value. Denote the current predict value as $\hat{x}_{n+1}$ (in this paper all variable with a cap means predictive value), the length of sliding time window is L, then sequences covered by sliding window is $x_{n-L+1}, x_{n-L+2}, \ldots, x_n$. Let

$$\sigma_{n+1} = \sqrt{MSE} = \sqrt{\frac{\sum_{t=0}^{L-1} e_{n-t}^{\ 2}}{L}} = \sqrt{\frac{\sum_{t=0}^{L-1} [x_{n-t} - \hat{x}_{n-t}]^2}{L}} \tag{4}$$

Before start, algorithm set the first predict value equal to the first observation value, then begin an initialization of time length L. The traffic anomaly detection algorithm based on SESP can be described as following pseudo-codes:

Table 1. Pseudo-codes for Non-periodic Traffic Anomaly Detection

BEGIN: (1) $\hat{x}_1 \leftarrow x_1$, $m \leftarrow 1$
(2) WHILE (m<=L)
DO {
$\hat{x}_{m+1} \leftarrow \alpha x_m + (1-\alpha)\hat{x}_m$
$e_m \leftarrow (\hat{x}_{m+1} - x_{m+1})$
$m \leftarrow m$+1}
(3) n ← L+1
(4) WHILE (TRUE)
DO { $S \leftarrow$ 0
FOR $m \leftarrow$ 1 TO L DO{
$S \leftarrow S$+ e_m^2 }
σ_{n+1} = SQRT (S/L)
$\hat{x}_{n+1} \leftarrow \alpha x_n + (1-\alpha)\hat{x}_n$
delta ← Abs($x_{n+1} - \hat{x}_{n+1}$)
RISKLEVEL ← 0 (Normal)
	IF delta > 8* σ_{n+1}
	THEN
	RISKLEVEL ← 1 (High)
	ELSE IF (delta > 5* σ_{n+1})
	THEN
	RISKLEVEL ← 0.5(Middle)
	ELSE IF (delta >3* σ_{n+1})
	THEN
	RISKLEVEL ← 0.2(Low)
	IF(RISKLEVEL ≠ 0)
	THEN
	Call Alert-Generator (RISKLEVEL)
	$x_{n+1} \leftarrow \hat{x}_{n+1}$
	n ← n+1}
	END.

After algorithm initialization, system handles a new coming traffic value as following steps:

a) Calculate MSE, let σ =$SQRT$ (MSE/L)
b) Predict the moment traffic value: $\hat{x}_{n+1} \leftarrow \alpha x_n + (1-\alpha)\hat{x}_n$
c) Estimate the inequation $x_i > \hat{x}_i + 3\sigma$, if TRUE turn to (d), FALSE turn to (e)
d) Anomalous handle: replace the anomaly value with the predict value: $x_n \leftarrow \hat{x}_n$
e) Calculate the current prediction error e_i, update the error series, observation series and predict series and move the sliding window forward for one step.

3.3.2 Periodic Traffic: Winters Level Seasonal Exponential Smoothed Prediction

Periodic traffic anomaly detection is different than non-periodic ones. Periodic traffic have significant traffic wave in usual time, but they can also be predicted. We use Winters Method and level seasonal exponential smoothed prediction (multiplicative model). This model decomposes the time series into 2 components: longtime trend index T and S. We regard these two indexes related each other, so we choose the multiplicative model.

The prediction model is:

$$y_{t+\tau} = T_t * S_{t+\tau-L}, (\tau = 1,2,3,...,L) \tag{5}$$

$$T_t = \alpha \frac{x_t}{S_{t-L}} + (1-\alpha)T_{t-1} \tag{6}$$

$$S_t = \gamma \frac{x_t}{T_t} + (1-\gamma)S_{t-L} \tag{7}$$

τ : the prediction steps, T_t: trend value predicted by last *t-L* periods S: seasonal index *L*: period's number α: smoothing weight (0.05≤α≤0.3) γ: Smoothing weight (0.5~0.6)

Before start, we need to assign initial values to T and S.

$$T_L = \frac{1}{L}\sum_{i=1}^{L} x_i, (i = 1,2,...,L) \tag{8}$$

$$S_i = \frac{x_i}{T_L}, (i = 1,2,...,L) \tag{9}$$

It will wait for 5 periods to get enough information to predict the future data profile. The error measurement is the same with equation (3) while the difference is that every interval on the data profile of periodic traffic has an error series and the series length is 5.

After algorithm initializaiotn, system will take following measure to detect periodic traffic anomalous when a new observation value comes:

- Calculate the MSE of prediction error based on the past 5 periods:

$$\sigma_i = \frac{\sum_{n=k-4}^{k-1} e_{nL+i}^{\ 2}}{4} = \frac{\sum_{n=k-4}^{k-1} [x_{nL+i} - \hat{x}_{nL+i}]^2}{4}, \ i = 1,2,. \ . \ . ,L \tag{10}$$

- Calculate the degree of current value deviation from the predict valueδ_{nL+i}:

  $\delta_{nL+i} = | x_{nL+i} - \hat{x}_{nL+i} |$, if $\delta_{nL+i} > 3\sigma_i$, turn to next step

  - If$\delta_{nL+i} > 8\sigma_i$, High risk, turn to anomaly handle
  - If$\delta_{nL+i} > 5\sigma_i$, Middle level risk, turn to anomaly handle
  - If$\delta_{nL+i} > 3\sigma_i$, Low level risk, turn to anomaly handle

  Anomaly handle: replace the current value: $x_{nL+i} \leftarrow \hat{x}_{nL+i}$

- Calculate the trend and seasonal index of current periods as equation (6-7):

3.4 Data Analysis and Result Discussion

The traffic anomaly algorithm developed in this paper has been tested on real network environment. The experiments were conducted on internet backbone. All traffic data is collects via ESTABD data center from 20 ISPs (see Fig. 1).

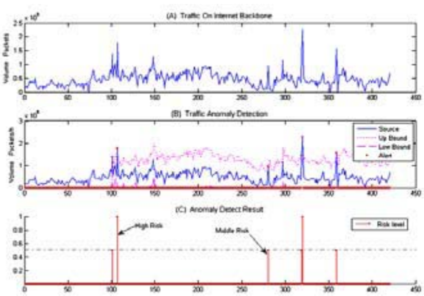


Fig. 3. Traffic anomaly detection on Port 1433

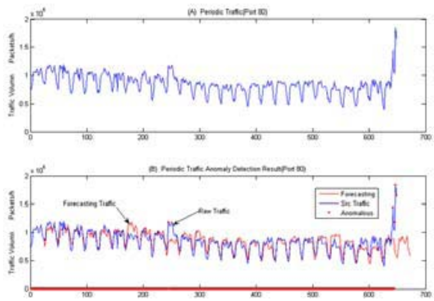


Fig. 4. HTTP traffic of 28 days from backbone

***Example A*: None-periodic traffic anomaly detection case — Port 1433 Traffic Analysis**

Fig. 3 (A) is the observed traffic of port 1433 on the backbone continuing for 18 days. The sampling interval is 1 hour. We can conclude from the figure: 1)the none-periodic traffic has no regular traffic pattern like periodic traffic, it changes from time to time randomly; 2) the common traffic is stationary and the meaning of the normal traffic is near about 0.5 x 10^6packets. Fig. 3(B) shows the anomaly detection process where blue line denote the raw traffic, dotted pink line denote the upper threshold and red dots are detected traffic anomalous. Fig. 3(C) is the detection result where y axis denotes risk levels. We detect 2 high risk warnings and 3 middle risk warnings based on our algorithm and the detection result is obviously successful.

***Example B*: Periodic traffic anomaly detection case — HTTP Traffic Analysis**

We have monitored the HTTP traffic data on the internet backbone for several months, and quote traffic of 28 days as Fig. 5 before a traffic anomaly incident happening. As it shows in Fig. 4(A), the data profile of real traffic is significantly periodic. There is no obvious linear trend of ascending or descending and the change of traffic is very slow. Fig. 4(B) shows the anomaly detection process, where blue line denotes raw traffic and red line denotes prediction values while red dots are those anomaly traffic detected algorithm.The risk level is defined in 3.3.1. Our algorithm detects the traffic anomalous as we expected. The risk levels of traffic anomalous are not marked on Fig. 4.

Table 2. The worm data officially published by CERT/CC recent years [12]

Worm	Affected Port	Vulnerabilities used	Target OS
Nimda	80,139,600	IIS, Code Red II and the backdoor left by Sadmind	Windows
Code Red I	80	IIS 4.0/5.0 Index Service	Windows
Code Red II	80	IIS 4.0/5.0 Index Service	Windows
Adore	23,53,111,515	Bind,LPRng, RPC.statd, wu-ftpd	Unix
Sadmind/IIS	80,111	IIS,Solstice,Sadmind	Win/Unix
Lion	53,10008	BINDservers	Unix
Ramen	27374	LPRng, rpc.statd, wu-ftpd	Unix(Redhat)
Cheese	10008	Backdoor left by Lion	Unix
Slapper	80,1433	OpenSSL,Apache	Unix
SQL Slammer	1433	Microsoft SQL Server	Windows
Witty	4000	ISS products(Black Ice, Realsecure)	Windows
MS Blaster	4444,69	RPC	Windows

3.5 Event Correlation and Monitoring Policy

As we discussed in 3.2, we can forecast the worm by detecting one or more associated events. Based on the table, we find that: the behavior of worm-explosion can be observed from one or more associated direct events (protocol-count variables). Each worm has its distinct features different than others. Taken worm Adore for example, if we have detected traffic anomaly on port (23, 53, 111, 515) and we also detected tremendous traffic ascending of secondary events ratio (SYN/FIN), we can say worm Adore is flooding. Because most of worm spreading using fast scan method, a lot of SYN packet will be generated to probe other hosts online, but many IP address is empty or not online or maybe there is no such service on the destination host, their will no FIN response, thus ratio (SYN/FIN) will increase tremendously. We define ratio (SYN/FIN) as a secondary events to detect the random scanning worms and Dos attacks such as SYN flood. By far we can detect the known worms and DDoS attacks by correlating alerts based on predefined pattern profiles. Based on knowledge we have, we can roughly define the pattern profile of unknown threats, and setup the ESTABD to monitor relevant direct and secondary events defined in pattern profile.

4 Conclusion and Future Works

In this paper, we present the traffic anomaly detection algorithm and introduce a framework design of traffic early-bird system for internet backbone namely ESTABD, which based on the previous algorithm.

ESTABD comprises three components: data center, traffic anomaly detection module and event correlation module. Data has been collected into data center from data engines deployed in all subnets of ISPs. Traffic anomaly detection module then detects anomalies from the data series provided by ESTABD data center. Firstly, we divided all traffic into two categories according to the nature of the traffic: periodic and non-periodic, it is an improvement to those algorithms generally process on all traffic with same data model simultaneously. Time series analysis and statistic prediction method are used in algorithm. Secondary, we use Single Exponentially Smoothed Prediction method to make prediction for non-periodic traffic while using Winters Level Seasonal Exponential Smoothed Prediction method for periodic traffic. Our algorithm provides the dynamic thresholds automatically based on the nature of history data, it is also an improvement over simple thresholding methods. Data analysis from real backbone network supported our algorithm. We studied the characteristics of traffic anomalies and come up with the conceptions of direct event and secondary event. Our algorithm has a very low cost of computation and does not need to maintain all history data in buffer except a fixed length of the sliding window for each traffic variable. Finally, we have discussed how to detect known worms and Dos attacks by defining the pattern profile of known threats and also provide with a practical method to monitor the coming and underlying unknown threats.By now, we have finished building the data center, the traffic anomaly detection module and the basic event correlation module of ESTABD, and plan to enhance the detection ability of traffic anomaly detection module by improve the detection algorithm performance in that: 1) better fitting with the all kinds of network environments; 2) reduce the false positive rate. As part of future work, we will improve the detection performance of event correlation module towards known threats and provide with an effect early warning ability for the subsequent security modules.

References

1. Moore D., Shannon et al. "Code-Red: a case study on the spread and victims of an Internet worm", In IMW, 2002
2. Moore D., Paxson V. et al. "The Spread of the Sapphire/Slammer Worm", CAIDA, ICSI, Silicon Defense, UC Berkeley EECS and UC San Diego CSE, 2003
3. N. Weaver, V. Paxson, S. Staniford and R. Cunningham, "A Taxonomy of Computer Worms", Proc. ACM CCS Workshop on Rapid Malcode, October 2003.
4. http://www.cnn.com/2001/TECH/internet/10/31/new.nimda.idg/
5. "Computer worm grounds flights, blocks ATM",
6. http://www.cnn.com/2003/TECH/internet/01/25/internet.attack/
7. "Security firm: MyDoom worm fastest yet"
8. http://edition.cnn.com/2004/TECH/internet/01/28/mydoom.spreadwed/index.html
9. Bharath Madhusudan, John Lockwood, et al. "Design of a System for Real-Time Worm Detection", In 12th Annual IEEE Symposium on High Performance Interconnects (Hot-I), August, 2004, Stanford, CA, pp. 77-83
10. A. Lakhina, K. Papagiannaki, M. Crovella, C. Diot, E. Kolaczyk, and N. Taft, "Structural analysis of network traffic flows," in Proc. ACM SIGMETRICS, June 2004.
11. P. Barford, J. Kline, D. Plonka, and A. Ron, "A signal analysis of network traffic anomalies," Internet Measurement Workshop 2002.
12. Throttan M., Ji C. "Adaptive Thresholding for Proactive Network Problem Detection", In: IEEE International Workshop on Systems Management, Newport, Rhode Island, 1998. 108-106
13. http://www.itl.nist.gov/div898/handbook/pmc/section4/pmc42.htm
14. http://www.cert.org/

On Network Model Division Method Based on Link-to-Link Traffic Intensity for Accelarating Parallel Distributed Simulation

Hiroyuki Ohsaki, Shinpei Yoshida, and Makoto Imase

Department of Information Networking,
Graduate School of Information Science and Technology, Osaka University,
1-5 Yamadaoka, Suita, Osaka 565-0871, Japan
{oosaki, s-yosida, imase}@ist.osaka-u.ac.jp

Abstract. In recent years, requirements for performance evaluation techniques of a large-scale network have been increasing. However, the conventional network performance evaluation techniques, such as mathematical modeling and simulation, are suitable for comparatively small-scale networks. Research on parallel simulation has been actively done in recent years, which might be a possible solution for simulating a large-scale network. However, since most existing network simulators are event-driven, parallelization of a network simulator is not easy task. In this paper, a novel network model division method based on link-to-link traffic intensity for accelerating parallel simulator of a large-scale network is proposed. The key ideas of our network model division method are as follows: (1) perform steady state analysis for the network model that is to be simulated, and estimate all traffic intensities along links in steady state, (2) repeatedly apply the minimum cut algorithm from graph theory based on the estimated traffic intensities, so that the simulation model is divided at the link that has little traffic intensities in steady state.

1 Introduction

In recent years, demand for a technique to evaluate the performance of large-scale networks has heightened [1, 2] along with the increasing size and complexity of the Internet. The Internet today is a best-effort network, and communication quality between ends is in no way guaranteed. Of course, robustness to some extent has been achieved through use of dynamic routing like OSPF and BGP even with the current Internet. However, the Internet itself is indispensable as society's communication infrastructure, so a technique to evaluate the performance of large-scale networks is in strong demand to ensure the reliability, safety, and robustness of networks, to allow future network expandability and design, and to assess the impact of terrorism and natural disasters.

However, conventional techniques to evaluate the performance of a network such as numerical analysis techniques and simulation techniques are directed toward relatively small-scale networks. As an example, queuing theory [3] as has been widely used in performance evaluation of conventional computer networks is not readily applied to performance evaluation of the large-scale and complex Internet. When strictly analyzing

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 749–757, 2005.

the performance of a network using queuing theory, the number of states for analysis increases tremendously together with the increase in the number of nodes connected to the network.

Techniques to approximately analyze interconnected networks like Jackson networks have been proposed even in queuing theory [3], although the packet arriving at a node is assumed to be a Poisson arrival. However, TCP/IP, the communication protocol for the Internet, is a complex, layered communication protocol with a complex traffic control algorithm and routing algorithm. As an example, the Internet uses various underlying communication protocols such as Ethernet, FDDI, and ATM, and creation of a rigorous numerical model of a complex system like this is not possible in realistic terms. Of course, numerical analysis techniques are extremely advantageous in terms of calculating time, so their use as a method of complementing other performance evaluation techniques is vital.

Simulation techniques, as opposed to numerical analysis techniques, allow performance evaluation of complex networks [4]. Performance evaluation of medium-scale networks in particular has become possible through the increasing speeds and capacities of computers in recent years. However, communication protocols for the Internet are extremely complex, so massive computer resources are required for simulation of networks, and simulation of large-scale networks is still difficult. The majority of network simulators widely used today simulate behavior at the packet level, so they use an event-driven architecture. A technique for faster speeds of network simulators operating on a single computer has also been proposed [5], although a different approach is needed to simulate a large-scale network.

Research with regard to parallel simulations as technology to allow simulation of large-scale networks has been conducted in recent years [6, 7, 8]. Construction of relatively inexpensive cluster computers has become easier through the faster speeds and lower prices of desktop computers and the spread of high-speed network interfaces such as Gigabit Ethernet. In addition, Grid computing using a wide-area network to integrate computer resources around the world has also attracted attention. However, the majority of network simulators have an event-driven architecture, so parallelization of network simulators is difficult.

Thus, this paper proposes division of a network model based on the traffic volume between links in order to run a simulation of a large-scale network at high speeds in a distributed computing environment and evaluate its effectiveness. The basic idea for the proposed division of a network model is as follows:

(1) Steady state analysis as proposed in the literature [9] would be performed on a network model (simulation model) to simulate, and the traffic volume passing through links in a steady state would be estimated.

(2) The simulation model would be divided by links with a low traffic volume using the minimum cut algorithm in the literature [10] based on the estimated traffic volume.

(3) The simulation model would be divided into N portions by repeatedly performing (1) and (2) so that the total traffic volume passing through nodes would be equal.

The simulation model would be divided into N portions via the aforementioned division and respective sub-network models would be run on N computers.

The composition of this paper is as follows. First, Section 2 describes related research regarding parallel simulation of networks. Section 3 explains division of a network model based on the traffic volume between links proposed. Section 4 indicates examples of the proposed division of a network model. In addition, Section 5 describes evaluation via a simple experiment of how much faster the parallel simulation would be through the proposed division of a network model. Finally, Section 6 describes this paper's conclusions and future topics for research.

2 Related Research

QualNet [11], OPNET [12], and PDNS [13] are typical network simulators that support parallel simulation. QualNet is a commercial simulator from Scalable Network Technologies and can be run on a single SMP (Symmetric Multi-Processing) computer [11]. Division (Smart Partitioning) of a simulation model, load dispersion (Load Balancing) per CPU, and maximized simulation look-ahead (Maximization of Lookahead) are techniques used to increase the speed of parallel simulation. However, it cannot be run on multiple computers such as cluster computers and cannot be used for simulation of large-scale networks.

OPNET is a commercial simulator from OPNET Technologies, and it can be run on a single SMP computer, although it cannot be run on multiple computers like cluster computers [12]. In addition, parallel simulation is only possible for specific modules for wireless networks, and the simulator cannot be used for simulation of large-scale networks.

PDNS [Parallel/Distributed NS] is a network simulator that was developed by the PADS research group at the Georgia Institute of Technology [13]. PDNS is an extension of the ns2 simulator [14] as is widely used in performance evaluation of TCP/IP networks and is run on parallel computers. With PDNS, simulation nodes can be distributed and run on different computers. As a parallel simulator, however, only extremely limited features have been implemented. When simply running a simulation of a large-scale network on multiple computers, simulation speed slows substantially due to overhead from communication between computers performing the simulation, a problem that has been pointed out [13]. Accordingly, performing simulation of large-scale networks is also difficult using PDNS as-is.

3 Division of a Network Model Based on the Traffic Volume Between Links

An overview of the proposed division of a network model will be explained. Below, the model of the network as a whole to simulate is called the "network model," and the models obtained by division of the network model are called "sub-network models." First, the network model to simulate is expressed in a weighted, undirected graph. The graph's vertices correspond to nodes (routers or terminals) and edges of the graph correspond to links between nodes. The traffic volume passing through a link in a steady state is used as the weight of the graph's edges. The basic idea is (1) to perform steady state analysis as proposed in the literature [9] on a network model (simulation model) to simulate and estimate the traffic volume passing through links in a steady state, (2) to divide the simulation model with links with a low traffic volume using a minimum cut algorithm in the literature [10] based on the estimated traffic volume, and (3) to divide the simulation model into N portions by repeatedly performing (1) and (2) so that the total traffic volume passing through nodes would be equal.

Specifically, the traffic volume passing through each link in a steady state is first derived using steady state analysis proposed in the literature [9] in instances where a network model to simulate and traffic demands between nodes are given. Moreover, several potential cuts in the network model are determined using the minimum cut algorithm proposed in the literature [10]. Of these, the cuts used were those with a small capacity (traffic volume passing between sub-network models) and simulation calculation time for two sub-network models (estimated by the total traffic volume in sub-network models) that is equal to the extent possible.

Next, a division algorithm like that mentioned above is again applied to a network model of individual sub-network models considered to have the maximum simulation calculation time. N sub-network models are obtained by repeating division like that mentioned above $N-1$ times to have a low traffic volume passing between sub-network models (i.e., slight overhead in parallel simulation) and to have an equal simulation calculation time (i.e., the loads on the computers performing the simulation would be equal) for each sub-network model.

Next, the algorithm for the proposed division of a network model is explained. Preceding an explanation of the algorithm, several forms of notation will be defined. A network model is thought of as undirected graph $G = (V, E)$. Here, $V = \{v_1, v_2, \ldots, v_n\}$ and $E = \{e_1, e_2, \ldots, e_m\}$. The weight of an edge (v_i, v_j) is $w_{i,j}$. Furthermore, the total number of divisions of the network model (the number of sub-network models) is N. In addition, the traffic model used in simulation is denoted by traffic matrixes $L = (l_{i,j})$ and $M = (m_{i,j})$. Here, $l_{i,j}$ is the transfer rate for UDP traffic from vertex v_i to vertex v_j and $m_{i,j}$ is the number of TCP connections from vertex v_i to vertex v_j. This paper deals with TCP traffic and UDP traffic to continuously transfer data for the sake of simplicity.

The algorithm for the proposed division of a network model is as follows:

1. Derivation of the traffic volume between links by steady state analysis
Steady state analysis of network model G and traffic matrices L and M are performed, and the traffic volume between links in a steady state is derived. The analysis technique proposed in the literature [9] is used for steady state analysis of the network. Thus, throughput for TCP traffic $T_{i,j}$ in a steady state and throughput for UDP traffic $L_{i,j}$ are determined. Here, $T_{i,j}$ and $L_{i,j}$ are throughput for TCP and UDP traffic passing through an edge (v_i, v_j) in a steady state.
2. Determination of the weight of the edges $w_{i,j}$
The weight $w_{i,j}$ of an edge (v_i, v_j) is defined as follows:

$$w_{i,j} = \sum_l \sum_m C_{l,m} T_{l,m} + \sum_l \sum_m D_{l,m} L_{l,m} \tag{1}$$

Here, if $m_{i,j}$ passes through edge (v_i, v_j) or edge (v_j, v_i), $C_{i,j}$ is 1; otherwise, it is 0. If, in addition, $l_{i,j}$ passes through edge (v_i, v_j) or edge (v_j, v_i), $D_{i,j}$ is 1;

otherwise, it is 0. Thus, weight $w_{i,j}$ means the sum of the throughput for all traffic passing through edge (v_i, v_j) and edge (v_j, v_i) in a steady state.

3. Initialization of the set M of subgraphs
   The set of subgraphs obtained by division is initialized via network model G.
$$M \leftarrow \{G\} \tag{2}$$
4. Model division using a minimum cut algorithm
   The number of divisions of the network model is N. The following process is performed repeatedly until $|M| = N$.
   (a) A sum of the weights of the edges $W(E)$ from the set M of subgraphs where the maximum subgraph $G' = (V', E')$ is selected. The sum of the weights of the edges $W(E)$ in a weighted, undirected graph $G = (V, E)$ is defined by the following equation.
$$W(E) = \sum_{(v_i, v_j) \in E} w_{i,j} \tag{3}$$
   (b) A minimum cut algorithm proposed in the literature [10] is run on weighted, undirected graph G. Thus, $|V'| - 1$ cuts $(S, \overline{S})$ are obtained. Here, the cut capacity is denoted as the n th small cut $(S_n, \overline{S}_n)$ $(1 \leq n \leq |V'| - 1)$
   (c) Subgraphs with a small cut capacity and equal sum of the weights of the edges to the extent possible are selected from $(S_n, \overline{S}_n)$. Specifically, Subgraphs S_n, $\overline{S}_n$ are selected so that
$$\frac{|W(S_n) - W(\overline{S}_n)|}{W(S_n) + W(\overline{S}_n)} \leq \alpha \tag{4}$$
   (α is a constant) is fulfilled and n is a minimum (i.e., a minimum cut capacity). Then, G' in the set M for subgraphs is replaced by $\{S_n, \overline{S}_n\}$.

The division of a network model as proposed in this paper has the following characteristics. First, the algorithm for the proposed division is a heuristic algorithm and uses a minimum cut algorithm in graph theory. In addition, calculations required for simulation of each sub-network model are estimated by calculating the sum of the weights of all edges $W(E)$ during division into sub-network models. Thus, calculations required for simulation of each sub-network model can be expected to be equal, as opposed to division simply using the traffic volume between links $T_{i,j}$ and $L_{i,j}$. The proposed division of a network model assumes steady, continuous TCP and UDP traffic and cannot handle traffic in bursts. In addition, calculations required for simulation of a sub-network model are estimated using weight $W(E)$, although validation of this method of estimation is required.

4 Examples of the Division of a Network Model Proposed

This section indicates examples of the division of a network model proposed. Here, an example of a network model with 16 nodes and a mean degree of 3 as in Fig. 1 is

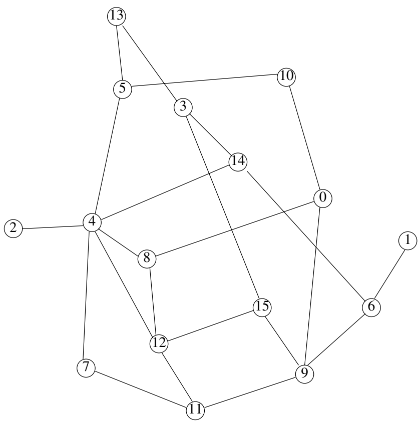


Fig. 1. Example of division of a network model (before an algorithm is run)

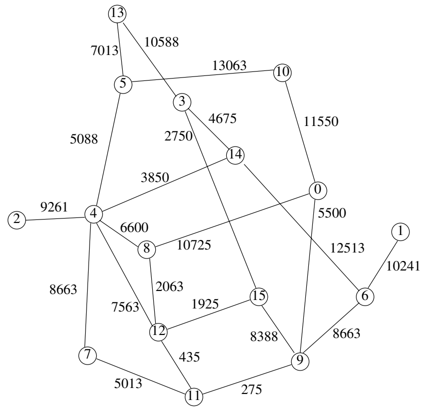


Fig. 2. Example of division of a network model (the weight of the edges $w_{i,j}$ is calculated from steady state analysis; the value for each edge is the traffic volume passing through a link [Kbyte/s])

used. The bandwidth for each link is a random value from 1 to 100 [Mbits/s], and the propagation delay for each link is a random value from 10 to 200 [ms]. In addition, the network model's number of divisions is $N = 4$ considering the fact that the simulation was performed on four parallel computers. Here, results are shown for when 1000 TCP connections were generated randomly.

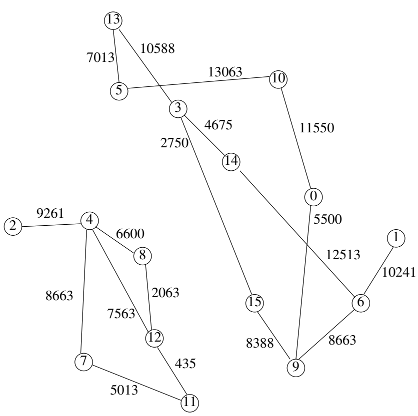


Fig. 3. Example of division of a network model (divided into two sub-network models using a minimum cut algorithm. The cut $(S, \overline{S}) = (\{2, 4, 7, 8, 11, 12\}, \{0, 1, 3, 5, 6, 9, 10, 13, 14, 15\})$ is applied)

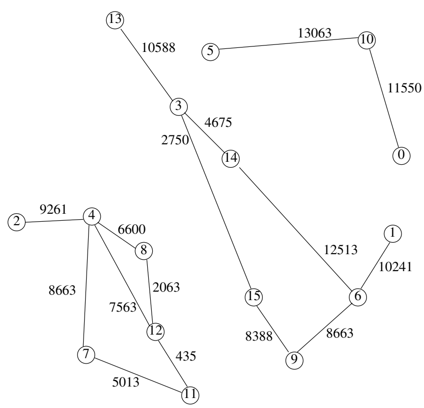


Fig. 4. Example of division of a network model ($W(S) < W(\overline{S})$, so $\overline{S} = \{0, 1, 3, 5, 6, 9, 10, 13, 14, 15\}$ is further divided into two sub-network models. The cut $(T, \overline{T}) = (\{1, 3, 6, 9, 13, 14, 15\}, \{0, 5, 10\})$ is applied)

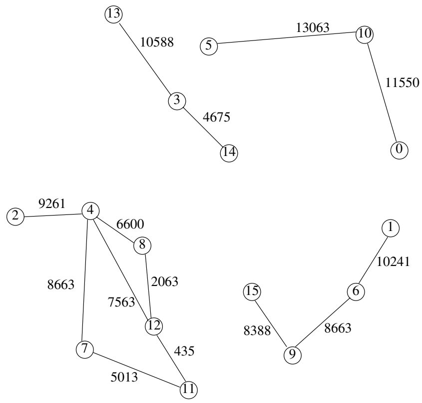


Fig. 5. Example of division of a network model ($W(T) > W(\overline{T})$, so $T = \{1, 3, 6, 9, 13, 14, 15\}$ is further divided into two sub-network models. The cut $(U, \overline{U}) = (\{3, 13, 14\}, \{1, 6, 9, 15\})$ is applied)

With respect to Fig. 1, steady state analysis from the literature [9] is performed, and the throughput of respective traffic $T_{i,j}$ and $L_{i,j}$ passing through each link in a steady state is derived. Based on this, the weight of each edge $w_{i,j}$ is calculated (Fig. 2) from Eq. (1).

The minimum cut algorithm in the literature [10] is run on the weighted, undirected graph $G' = (V', E')$ in Fig. 2, and $|V'| - 1 = 15$ cuts $(S, \overline{S})$ is obtained. Of these cuts, those for which the sum of the weight of the edges $W(S)$ and $W(\overline{S})$ fulfills Eq. (4) in subgraphs S and $\overline{S}$ those with a minimum cut capacity is applied (Fig. 3). In this example, $(S, \overline{S}) = (\{2, 4, 7, 8, 11, 12\}, \{0, 1, 3, 5, 6, 9, 10, 13, 14, 15\})$ and the cut is applied so that the cut capacity will be 21,863 [Kbyte/s], $W(S)$ = 39,598 [Kbyte/s], and $W(\overline{S})$ = 94,944 [Kbyte/s].

In Fig. 3, $W(S) < W(\overline{S})$, so $\overline{S} = \{0, 1, 3, 5, 6, 9, 10, 13, 14, 15\}$ is further divided into two sub-network models (Fig. 4). In this example, $(T, \overline{T}) = (\{1, 3, 6, 9, 13, 14, 15\}, \{0, 5, 10\})$, and the cut is applied so that the cut capacity will be 12,513 [Kbyte/s], $W(T)$ = 57,818 [Kbyte/s], and $W(\overline{T})$ = 24,613 [Kbyte/s].

Moreover, N (= 4) sub-network models are obtained by repeating the same procedure. Here, $W(T) > W(\overline{T})$, so $T = \{1, 3, 6, 9, 13, 14, 15\}$ is further divided into two sub-network models (Fig. 5). In this example, $(U, \overline{U}) = (\{3, 13, 14\}, \{1, 6, 9, 15\})$, and the cut is applied so that the cut capacity will be 15,263 [Kbyte/s], $W(U)$ = 15,263 [Kbyte/s], $W(\overline{U})$ = 27,292 [Kbyte/s].

5 Evaluation of the Division of a Network Model Proposed

This section describes evaluation via a simple experiment of how much faster the parallel simulation would be through the proposed division of a network model. In testing, the running time for parallel simulation (time from the start of simulation until the simulation ended) was measured when a network was divided into several sub-network models

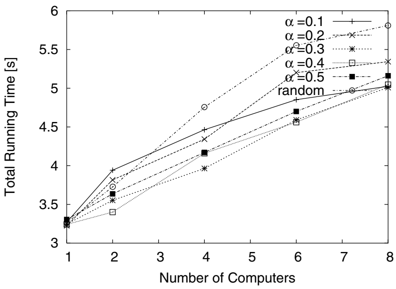


Fig. 6. Total running time required for completing all simulation events (10 nodes, degree 2, link bandwidth 1–100 [Mbit/s], link propagation delay 0.1–100 [ms], and 10 TCP connections)

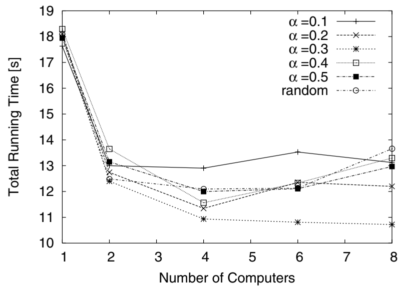


Fig. 7. Total running time required for completing all simulation events (100 nodes, degree 2, link bandwidth 1–10 [Mbit/s], link propagation delay 0.1–100 [ms], and 100 TCP connections)

using the proposed division method and when a network was randomly divided into sub-network models for balancing the number of nodes in each sub-network model.

In testing, a network model was generated by a random graph of 10 or 100 nodes with a mean degree of 2. Bandwidth for each link in the network model was a random value from 1 to 10 or 100 [Mbits/s], and the propagation delay for each link was a random value from 0.1 to 100 [ms]. In addition, 10 or 100 TCP connections were randomly generated between nodes. Under these conditions, 10 network models were generated, and these were respectively evaluated with regard to when the model was divided into two sub-network models using our proposed division method and when the model was divided simply so that the number of nodes in sub-network models would be equal.

PDNS [13] version 2.27-v1a was used as a parallel network simulator, and simulation was performed for 30 [s]. PDNS version 2.27-v1a's default values were used for the packet length, TCP parameters, router buffer size, and the like. 8 computers with the same performance as shown below were used in testing:

- CPU: Pentium III 1,266 MHz
- Memory: 1,024 Mbyte
- Hard disk: 120 Gbyte
- Network: 1 Gbit/s Ethernet
- Operating system: Linux version 2.4.20

Figure 6 shows the total running time required for completing all simulation events for 10 nodes and 1–100 [Mbit/s] link bandwidth. Figure 7 shows the total running time required for completing all simulation events for 100 nodes and 1–10 [Mbit/s] link bandwidth. In these figures, the number of computers running a parallel distributed simulator is changed as 1, 2, 4, 6, and 8, and the parameter α is as 0.1, 0.2, 0.3, 0.4, and 0.5. The results with a random division method are labeled as "random". These figures show with our proposed division method, the total running time becomes about 78%–96% (Fig. 6) and 78%–94% (Fig. 7) of the case with a random division method, indicating significant performance improvement with our proposed division method.

6 Conclusions and Future Topics

This paper proposed division of a network model in order to simulate large-scale networks in a distributed computing environment at high speeds. The proposed division of a network model first derived the traffic volume between links through use of steady state analysis of a network model to simulate. This technique then applies a minimum cut algorithm from graph theory several times in accordance with the traffic volume between links in a steady state and divides a network model into N portions.

Various extensions of the division of a network model for faster parallel simulation as proposed in this paper may be possible in the future. First, this paper dealt with TCP and UDP traffic where data is continuously transferred. Thus, division of a network model can be expanded so as to handle TCP traffic to transfer data in bursts. In addition, this paper dealt with unicast traffic alone, although expansion so as to handle multicast traffic is needed in order to simulate an actual large-scale network.

References

1. Large Scale Networking (LSN) Coordinating Group Of the Interagency Working Group (IWG) for Information Technology Research and Development (IT R&D), *Workshop on New Visions for Large-Scale Networks: Research and Applications*, Mar. 2001. available at http://www.nitrd.gov/iwg/lsn/lsn-workshop-12mar01/index.html.
2. S. Floyd and V. Paxson, "Why we don't know how to simulate the Internet," Oct. 1999. available at http://www.aciri.org/floyd/papers/wsc.ps.
3. D. Bertsekas and R. Gallager, *Data Networks*. Englewood Cliffs, New Jersey: Prentice-Hall, 1987.
4. A. M. Law and M. G. McComas, "Simulation software for communications networks: the state of the art," *IEEE Communications Magazine*, vol. 32, pp. 44–50, Mar. 1994.
5. V. S. Frost, W. W. Larue, and K. S. Shanmugan, "Efficient techniques for the simulation of computer communications networks," *IEEE Journal of Selected Areas in Comuunications*, vol. 6, pp. 146–157, Jan. 1988.
6. H. T. Mouftah and R. P. Sturgeon, "Distributed discrete event simulation for communications networks," *IEEE Journal on Selected Areas in Communications*, vol. 8, pp. 1723–1734, Dec. 1990.
7. G. F. Riley, R. M. Fujimoto, and M. H. Ammar, "A generic framework for parallelization of network simulations," in *Proceedings of the Seventh International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, pp. 128–135, Oct. 1999.
8. S. Bhatt, R. Fujimoto, A. Ogielski, and K. Perumalla, "Paralell simulation techniques for large scale networks," *IEEE Communications Magazine*, vol. 38, pp. 42–47, Aug. 1998.
9. D. Dutta, A. Goel, and J. Heidemann, "Faster network design with scenario pre-filtering," in *Proceedings of the International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, pp. 237–246, Oct. 2002.
10. M. Stoer and F. Wagner, "A simple min-cut algorithm," *Journal of the ACM*, vol. 44, pp. 585–591, July 1997.
11. Scalable Network Technologies, Inc., "QualNet." http://www.scalable-networks.com/.
12. Opnet Technologies, Inc., "OPNET." http://www.opnet.com/.
13. PADS (Parallel and Distributed Simulation) Research Group, "PDNS - Parallel/Distributed NS." http://www.cc.gatech.edu/computing/compass/pdns/.
14. "The network simulator – ns2." available at http://www.isi.edu/nsnam/ns/.

Network Traffic Sampling Model on Packet Identification

Cheng Guang, Gong Jian, and Ding Wei

Department of Computer Science & Engineering, Southeast University,
Jiangsu Provincial Key Lab of Computer Network Technology, Nanjing, 210096
(gcheng, jgong, wding) @njnet.edu.cn

Abstract. A new sampling model for measurement using IP packet identification (IPID) on IP network is provided in this paper under a principle of PSAMP, a working group of IETF, that a good sampling model should work for all purposes of measurement applications at the same time with a simple way. In the paper, and a multi-mask sampling model on the identification field can not only control sampling precise to 1/65536, but also use different sampling parameters among different measurement points. The randomicity and coordination of sampled packets can be assured automatically, and both network traffic performance and statistical characters are analyzed.

1 Introduction

With the rapid development of Internet applications, network behavior problems grow quickly and become more and more complex, which makes the study on it a hot focus of relative research field now [1]. Network measurement [2] is the foundation of network behavior research. Passive measurement is applied to research traffic statistics behavior, such as accounting and traffic management. In recent years the passive measurement technology is also used in network behavior, such as end-to-end network behavior [3] and routing behavior [4] .

PSAMP [5] suggests sampling model should work for all purposes of measurement applications at the same time with a simple way. I. COZZANI [3] used bit pattern checksum sampling model for end-to-end QoS in ATM network, and N. DUFFIELD [4] analyzed routing behavior with hash function sampling model. Unfortunately, without guarantee of randomicity, these two methods couldn't be used in sampling traffic and the statistics behavior analyzing on it. Cheng [6] finds high randomicity of bits in IPID and proposes a single-mask sampled model on IPID. But both the single-mask sampling model and other distributed sampling models face two same problems. The first problem is that sampling ratio cannot be controlled arbitrarily, and the second problem is that there must control same sampling parameters in distributed measured points. Cheng [6] devised a multi-mask sampling model that can control sampling ratio to 1/65536, but cannot solve the second problem.

A new multi-mask sampling coordination model on IPID designed in this paper to control sampling precise 1/65536, and use different sampling parameters among different measurements points. Sampled packets have both randomicity and

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 758–765, 2005.

coordination with the model. The multi-mask sampling algorithm is on the randomicity of IPID. In the following sections, first the single-mask sampling measurement model is described. Second, randomicity of IPID is compared by NLNAR/PMA monitoring traffic. Third, a multi-mask sampling model proposed in this paper, solves the two problems of distributed sampling model. Fourth, the multi-mask sampling model is compared with other sampling models. Last, the conclusion is given out.

2 Single-Mask Sampling Model on IPID

Entropy [7], an important concept of information theory, is extended to estimate bit randomicity in this paper. Bit Random metric is defined as following.

Definition 1, Bit Random Metrics, a metric of bit randomicity is represented by the ratio between H(b) and Hmax(b). Hmax(b)=1, $E = H(b)/H_{\max}(b) = H(b)$, $0 \le E \le 1$. A bit is random, while E approaches to 1, and vice versa. Where $H(b) = -(p_0 \log_2 p_0 + p_1 \log_2 p_1)$, $H_{max}(b) = 1$.

Choosing some fixed bits in an IP packet as the measuring sample, coordination can be assured simply. Suppose these chosen bits can be proved to assure statistical randomicity, they can be used as the sampling mask bits in measured model for traffic statistical analysis. If probability whose masking bit appears to 0 or 1 is 0.5 separately, and these mask bits obey independency identity distributing, then the theoretic sampling ratio is equal to $1/2^n$, where n is the mask length. Actually, it is very difficult to assure that each mask bit has equal probability and keep independent identity distributing. As Bit Random Metric E approaches to 1, the sampling ratio approaches to theoretic one.

It is easy to assure measuring coordination with the sample model, but randomicity can’t be proved by mathematics method, and can only be analyzed from network traffic statistically. CERNET backbone traffic is analyzed [6], and a single-mask sampling model is established on the IPID field.

3 Randomicity Comparison of Identification Field

Cheng [6] has analyzed the IPID randomicity in CERNET backbone. IPID randomicity in some other monitors is analyzed as following.

NLANR is a distributed organization with three parts: application/user support, engineering services, measurement and analysis. The Measurement and Network Analysis team, located at UCSD/SDSC, conducts performance and flow measurements for HPC sites. Two projects form the core of this work group: the Passive Measurement and Analysis (PMA) project and the Active Measurement Project (AMP).

PMA establishes many passive measurement Internet data monitors in the HPC network. The data measured from AIX and TXS on Sep. 30, 2002, and its data format is TSH, whose packet header is 44 bytes length. TSH data format includes both IP header and TCP header of measured packets, so we can obtain the character of identification field. Table 1 is the identification statistics in AIX monitor, and Table 2 is the identification statistics in TXS monitor. The two tables show the identification randomicity, and we also find that identification 0 value is larger than 1/65536 of the theory value.

The random metric of IPID bit is larger than 99%. In TXS1, the ratio of IPID 0 is 11.29%, and that of TXS5 is 7.93%, so their random metric are less than other traffic data. The experiment result show that random statistics of identification field is a common rule in IP packet network, so the sampling model on the identification field can be applied various IP networks.

Table 1. Measured Traffic in both TXS and AIX Monitors

Monitor	Number	Packet Number	IPID Random Value	0 Value of IPID	Ratio of IPID 0
TXS	1	465849	0.9124	52589	11.29%
	2	369929	0.9750	9243	2.50%
	3	506047	0.9814	11584	2.29%
	4	386250	0.9735	11935	3.09%
	5	370527	0.9384	29389	7.93%
	Sum	2098602	0.9632	114740	5.47%
AIX	1	632047	0.9830	12679	2.01%
	2	626495	0.9804	15279	2.44%
	3	405447	0.9754	11213	2.77%
	4	322694	0.9729	9177	2.84%
	Sum	1986683	0.9839	48348	2.43%

4 Distributed Multi-mask Sampling Model

The Multi-mask sampling model [8] can control the sampling ratio to 1/65536, but every monitor must have a same sampling mask at least, so it is very difficult to adjust self-sampling ratio. Especially, if many monitors must be coordinated, then many interactions between monitors will be appeared to assure a same sampling mask at least.

If we find another multi-mask model, which can solve a problem that measured sample of a bigger ratio includes that of a less ratio, then the second problem also be solved automatically, and sampling ratio among monitors need not be coordinated. We will describe the multi-mask sampling model idea as following.

A sampling ratio can be decomposed into $1/2^{a1}$, $1/2^{a2}$, …, $1/2^{ai}$, …,$1/2^{an}$, $ratio = \sum_{i=1}^{n} 1/2^{ai}$, where a_i is the length of a sub-mask. a_1, a_2, …, a_n are arranged from small to big, and their masks are b_1, b_2, …,b_n. Mask b_i is defined as following:

b_i mask length is a_i, from the begin to end of identification field, its offset is 0. Except the bits in a_1, a_2,…, a_{i-1} positions are set into 0, other bits from 0 to a_i positions are set into 1. According to the mask definition, the multi-mask sampling algorithm can assure that a big sampling ratio sample include a small sampling ratio sample. The model will be analyzed as following in detail.

The positions of less the mask b_i, a_1, a_2, a_{i-1} are set into 0, and others positions among 0 to a_i are set 1, every mask among n sub-masks will not have interaction with other sub-mask. If $\Omega(b_i)$ is the aggregation of b_i mask measured sample, so there are equation (1) and (2).

$$\Omega(b_i) \cap \Omega(b_j) = \Phi \; (1 \le i \ne j \le n) \tag{1}$$

$$\Omega(ratio) = \Omega(\sum_{i=1}^{n} b_i) = \sum_{i=1}^{n} \Omega(b_i) \tag{2}$$

Second problem is if Aratio >= Bratio, then there is equation (3).

$$\Omega(Aratio) \supseteq \Omega(Bratio) \tag{3}$$

Sampling ratio Aratio and Bratio can be decomposed into $\sum_{i=1}^{a} 1/2^{a_i}$ and $\sum_{i=1}^{b} 1/2^{b_i}$ respectively. Due to Aratio => Bratio, while $a_i = b_i \quad (i = 1, j)$, 0 <= j <=a, then $a_{j+1} < b_{j+1}$, or j = b and b < a. If the two sub-masks of both Aratio and Bratio are same, then their measured sample on the two sub-masks certainly are same. If all sub-mask of both Aratio and Bratio are same, and the number of sub-mask in Aratio are bigger than that in Bratio, so the sample with Aratio sampling ratio will include the sample with Bratio sampling ratio. If $a_{j+1} < b_{j+1}$, the mask of both a_{j+1} and b_{j+1} can be expressed as equation (4) and (5).

$$mask\_a_{j+1} = \{1 \cdots 10(a_1) 1 \cdots 10(a_2) 11 \cdots 0(a_j) 1 \cdots 1(a_{j+1}) \{0|1\}_{16-a_{j+1}}\} \tag{4}$$

$$mask\_b_{j+1} = \{1 \cdots 10(b_1) 1 \cdots 10(b_2) 11 \cdots 0(b_j) 1 \cdots 1(a_{j+1}) \cdots 1(b_{j+1}) \{0|1\}_{16-b_{j+1}}\} \tag{5}$$

Because the front j items are same, so the equation (6) can be obtained.

$$mask\_a_{j+1} \supseteq mask\_b_{j+1} \tag{6}$$

So the mask from b_{j+1} will belong to $mask\_a_{j+1}$. $mask\_a_{j+1} \supseteq mask\_b_k \; (j+1 \le k \le b_b)$

If Aratio >= Bratio, then sample with Bratio sampling ratio will be included into sample with Aratio sampling ratio. So the sample with their minimal sampling ratio can be measured in all monitors, and can be applied to analyze the network performance.

Multi-mask Sampling Algorithm

```
Sub-mask i length is mask_length(i), (i=0,length-1),
  length is the number of sub-mask;
  cur_mask = 0; // specify the current mask length.
  for(i=0; i < 16; i ++)
   {if (identification[i] == 0)
    {if (i > mask_length(cur_mask))
      return sampling;       // sample the packet.
     else if (i < mask_length(cur_mask))
      return unsampling;   //don't sample the packet.
      else if (i = mask_length(cur_mask))
      {if (cur_mask == length-1)//the end of sub_mask i.
         return unsampling; //don't sample the packet.
      else
       cur_mask++; //continue the next sampling mask. }}}
```

The maximal loop times is 16, so time complexity of the algorithm is O(0), and the time complexity is relation to the number of sub-masks.

For example: 0.356 sampling ratio is compared with 0.41 sampling ratio.

$$0.356 = 1/2^2 + 1/2^4 + 1/2^5 + 1/2^7 + 1/2^8 + 1/2^{11} + 1/2^{15} + 0.00001245$$

$$0.41 = 1/2^2 + 1/2^3 + 1/2^5 + 1/2^9 + 1/2^{10} + 1/2^{11} + 1/2^{12} + 1/2^{14} + 1/2^{16} + 0.00001160$$

The decomposed error of 0.356 is 0.00001245, and the error of 0.41 is 0.00001160. their relative errors are 3.5×10^{-5} and 2.83×10^{-5} respectively. Table 2 is the sub-masks of both 0.356 and 0.41. The table shows that the second item mask "101" of 0.41 includes these masks from second item to sixth item of 0.356.

Table 2. Sub-masks of both 0.356 and 0.41

Number	Mask length	Sampling ratio (fraction)	Sampling ratio (decimal fraction)	Sampling mask
0.356	2	$1/2^2$	0.25	11
	4	$1/2^4$	0.125	1101
	5	$1/2^5$	0.0625	10101
	7	$1/2^7$	0.0078125	1101101
	8	$1/2^8$	0.00390625	10101101
	11	$1/2^{11}$	0.00048828125	11100101101
	15	$1/2^{15}$	0.000030517578125	111101100101101
0.41	2	$1/2^2$	0.25	11
	3	$1/2^3$	0.125	101
	5	$1/2^5$	0.03125	11001
	9	$1/2^9$	0.001953125	111101001
	10	$1/2^{10}$	0.0009765625	1011101001
	11	$1/2^{11}$	0.00048828125	10011101001
	12	$1/2^{12}$	0.000244140625	100011101001
	14	$1/2^{14}$	0.00006103515625	11000011101001
	16	$1/2^{16}$	0.0000152587890625	1101000011101001

5 Comparison of Algorithm Performance

Threshold algorithm

Let threshold be F, sampling ratio be p, and the IPID of the passing packet be x, then F=65536×p. If 0<x<=F, then the packet is captured, else the packet is lost. The threshold algorithm absolute error is 1/65536, and relative error 1/65536*p. If a packet of a host is sent, then its IPID is added 1. If its IPID is less than the threshold F, then all packets will be captured, and else all packets are lost.

Modulus Algorithm

Let modulus be *M*, sampling ratio be *p*, if 0 < value mod M <= p*M, then the packet will be captured, else the packet is lost. Sampling precise of the modulus algorithm is decided by modulus M, whose the least sampling precise is 1/M. All monitors need coordinate the same modulus with the threshold algorithm.

Randomicity Comparison of Three Sampling Algorithms

Definition 2: Sampling Random Metric. Let n+1 events, whose probability is p_0, $p_1,\ldots, p_n$, so sampling entropy H(s) can be defined as $H(s) = -\sum_{i=0}^{n} p_i \log_2 p_i$. If n+1 sampling events have same sampling probability, that is to say $p_0=p_1=\ldots=p_n=1/(n+1)$, the sampling entropy value will arrive to maximum, and $H_{\max}(s) = \log_2(n+1)$, so the sampling random metric is defined as $E=H(s)/H_{max}(s)$, and $0<=E<=1$. The metric can be used to evaluate randomicity of sampling algorithm.

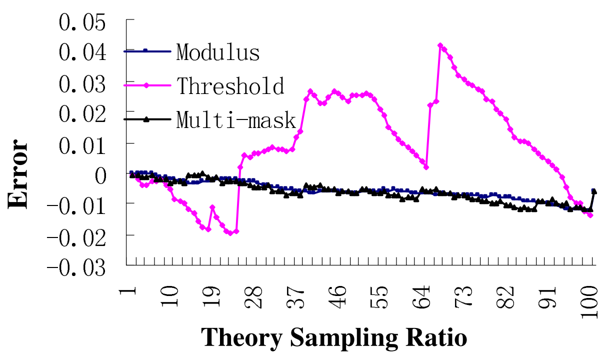


Fig. 1. Error of theory and measured sampling ratio. The result shows that the error of both modulus model and multi-mask model is less than the error of threshold model. Due to IPID 0, the two curves of both modulus model and multi-mask model departure from the axis distinctly

All packets measured during 60s divides into 100 components. Due to IPID 0 packets larger than other packets, so the 100 packet components don't include these identification 0 packets. The sampling random metric mean of modulus is equal to 0.999996, and the sampling random metric mean of threshold is equal to 0.999106. So randomicity of the modulus model is larger than that of the threshold model.

10^6 packets are analyzed using three sampling model of multi-mask model, threshold model, and modulus model, and the error series between theory ratio and measured ratio from 1/100 to 99/100 ratio the theory ratio is described in figure 1.

6 Conclusion

The sampling measurement on network traffic is a hot focus in network behavior research field. After three groups of traffic in CERNET, TXS, and AIX are analyzed, we verify that bits in IPID aren't changed during the transport process and the randomicity of them are very high. In this paper, three sampling algorithm on IPID are provided, that is the multi-mask sampling model, the threshold sampling model, and the modulus sampling model. The sampling randomicity is compared among the three sampling model. We find that the multi-mask sampling model has three advantages: randomicity, 1/65536 of sampling ratio, and without coordinating the sampling parameter among monitors.

It is very easy also to apply the sampling model in a router or measurer. Because IPID isn't changed while it is transported, so the coordination of sample in distributed monitors can be assured, that means the multi-mask sampling model is used not only for traffic behavior analysis, but also in network behavior research.

Acknowledgments

The project is supported by the National Natural Science Foundation of China under grant No. 90104031, and the 973 program of China under grant No. 2003CB314803.

References

1. Kevin Thompson, Gregory J. Miller, and Rick Wilder, Wide-Area Internet Traffic Patterns and Characteristics (Extended Version), IEEE Network, November/December 1997.
2. CAIDA Homepage, http://www.caida.org. November 2004.
3. Cozzani, I.; Giordano, S, A passive test and measurement system: traffic sampling for QoS evaluation, Global Telecommunications Conference, 1998. GLOBECOM 1998. The Bridge to Global Integration. IEEE , Page(s): 1236 –1241, Volume: 2 , 1998
4. Nick Duffield, Matthias Grossglauser. Trajectory sampling for direct traffic observation [J]. IEEE/ACM Transactions on Networking, June 2001, Vol. 9, No. 3: 280-292.
5. Packet Sampling (psamp) Bof, Minutes of the Packet Sampling (PSAMP) BOF IETF 53, Minneapolis, Tuesday March 19, 2002; 9:00-11:30, http://www.ietf.org/proceedings/02mar/164.htm

6. Cheng Guang, Gong Jian, Ding Wei, A Traffic Sampling Model for Measurement Using Packet Identification, ICON 2002. Proceedings 10th IEEE International Conference on Networks, in Singapore, pp: 409-413, August 27-30, 2002.
7. Jin Zhenyu, Information Theory, Beijing University of Science and Technology Press, pp: 11 - 47, 1991.12, BeiJing. (in Chinese)
8. Cheng guang, Research on Traffic Sampling Measurement and Behavior Analysis in Large-Scale High Speed IP Networks, Ph.D. dissertation Southeast University, 2003, pp: 24-44. (in Chinese)

An Admission Control and Deployment Optimization Algorithm for an Implemented Distributed Bandwidth Broker in a Simulation Environment

Christos Bouras and Dimitris Primpas

Research Academic Computer Technology Institute,
61 Riga Feraiou Str., 26221 Patras, Greece
Department of Computer Engineering and Informatics,
University of Patras, 26500 Rion, Patras, Greece
Tel: +30-2610-{960375, 960316} Fax: +30-2610-960358
{bouras, primpas}@cti.gr

Abstract. This paper describes and tests a distributed bandwidth broker that has been implemented in NS simulator. It focuses on the admission control algorithm, its advantages and drawbacks. Also, the bandwidth broker is tested, managing the IP Premium service and we compare 2 different implementations of the service. Finally it approaches the problem of the optimal location of a bandwidth broker in a backbone network. For this purpose, a new model is proposed that evaluates each node and finally selects the most capable node where the base bandwidth broker should be located.

1 Introduction

A bandwidth broker [1][2] is an entity that operates in a backbone network and is responsible to manage QoS service. Actually, it receives demands for bandwidth allocation; it processes them and decides if it can satisfy them. In case that the answer is positive, the bandwidth broker configures the network devices (routers, switches etc) to provide the bandwidth guarantees. This area is a widely open research issue, where several research team works on. There are many scientific papers on this area, where several architectures and algorithms have been presented [4][5][6][7].

We have implemented such a bandwidth broker in a simulation environment. The bandwidth broker as it has been implemented follows a generic architecture and is consisted of various modules. Those modules are: an admission control module that also contains a decision module, which describes the algorithm that runs in order to check each request. Additionally, the admission control module has a second module that stores all the necessary information for bandwidth broker's operation and also updates them whenever it is necessary. Besides, there is a module that is available to end users to make their requests. Finally, the implemented bandwidth broker has a module that describes the QoS service that it supports (classification, queue and scheduling algorithms etc) and it configures the network devices accordingly in each accepted demand. The bandwidth broker has been implemented using an independent implementation of each module and now it will be tested in order to evaluate its performance.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 766–773, 2005.

The rest of the paper is organized as follows: Section 2 has a description of the implemented bandwidth broker focusing on the admission control algorithm. Section 3 presents the QoS service that the bandwidth broker manages and describes the simulation tests that we performed, comparing 2 alternative implementations of the same IP Premium QoS service (using different queue management mechanisms). Next, section 4 approaches the problem of the selection of the optimal node to host the bandwidth broker where we propose a new model that selects the best node using various criteria. Finally, section 5 describes our conclusions as well as the future work that we intend to do on this area.

2 Bandwidth Broker Implementation in NS Simulator

Simulation has always been a valuable tool for experimentation and validation of models, architectures and mechanisms in the field of networking. It provides an easy way to test various solutions in order to evaluate their performance without needing a real network dedicated for experiments. In our case, a bandwidth broker has been implemented and tested on simulation environment (NS-2 [12]) in order to evaluate its performance characteristics and its used mechanisms.

The implemented bandwidth broker [11] on NS-2 followed the classic architecture of a bandwidth broker. The implementation required several changes and additions in the NS structure and source code. In particular, the bandwidth broker that was implemented is based on two new agents, the Edge Bandwidth Broker (BBedgeAgent) and the Base Bandwidth Broker (BBbaseAgent). BBbaseAgent creates BBB packets and consumes BBE packets created by the BBedgeAgent. BBedgeAgent creates BBE packets and consumes BBB packets created by the BBbaseAgent. A BBedgeAgent, which represents a client (user) can send a RAR requesting guaranteed bandwidth between the node it is running and another node The BBedgeAgent that exists on every node simulates a situation where a BB client is connected to a router on a real network. This agent operates as client that makes the communication with the base BB and updates its local router with the configuration modifications according to new admissions. In our case, this agent also stores data regarding the adjacent nodes of the node and communicates with the base BB every time the base BB needs this information. So, the architecture is somewhat distributed as some information is stored locally on every “client” and not centrally on the base BB.

2.1 The Admission Control Algorithm

A very important module in a bandwidth broker is the admission control. There are several algorithms that has been proposed for efficient admission of requests [8][9][10]. But in our case, where the operation is distributed, we designed and implemented a simple distributed admission control algorithm, where the base bandwidth broker agent runs only the main part of the algorithm, in order to ensure the coordination and the proper whole operation.

The system's operation begins when an Edge Bandwidth Broker makes a request asking guaranteed bandwidth of x bps from the node it is running to some other network node. Then, the Base Bandwidth Broker begins to serve the request by running the admission control. It searches the routing tables to find the next hop from the node n0 that made the request to the other end-node nk. Then, the Base Bandwidth Broker sends a query to the Edge Bandwidth Broker that runs on node n0 asking if there is available bandwidth between the nodes n0 and n1. If the answer is positive, the Base Bandwidth Broker finds the next hop n2 from node n1 to node nk and sends a query to node n1 asking if there is available bandwidth between the nodes n1 and n2. If all the answers are positive, this procedure continues until node nk is reached. This means that there exists available bandwidth from node n0 to node nk and the Base Bandwidth Broker will send a positive answer to the Edge Bandwidth Broker that made the request so that node n0 is notified that it is allowed to begin sending data. The procedure will be completed after the Base Bandwidth Broker sends to all the Edge Bandwidth Brokers that lay on the path n0,n1,…,nk, messages informing them to reduce by x bps the available bandwidth to the links that lay on the path. In case one of the Edge Bandwidth Brokers sends a negative answer, because there is not sufficient available bandwidth on a link, the Base Bandwidth Broker sends a negative answer to the node that made the initial request and the procedure ends there.

Sequentially, after the successful admission of a new request, the bandwidth broker should run the resource allocation module that configures properly the backbone routers across the path to provide the admitted guarantees.

2.2 Advantages and Disadvantages of Admission Control Algorithm

The implemented admission control algorithm has many advantages and some drawbacks. In particular, this module (admission control) is operated distributed, as parts of this algorithm run in the clients and a part and the basic synchronization in the base agent. Also, this admission control algorithm only needs simple data structures in the base and edge bandwidth broker agents. Each edge bandwidth broker must store information only for its links that manages. Initially, this information is the maximum bandwidth of the link that is available for the QoS service and the reserved bandwidth. The maximum available bandwidth for reservation on the link is determined by the network dimensioning. On the other hand, the base bandwidth broker agent needs to store more information as the nodes that are managed by the bandwidth broker, the links that each node manages and some data structures that should be used during the processing of every request. The nodes that participate in the bandwidth broker operation can be stored using only an array that should be updated each time a node introduce itself in the bandwidth broker operation or delete itself from the bandwidth broker. Also, this makes the algorithm highly extensible due to the fact that the necessary information for a new node and link is stored locally (in the client agent) and therefore, the bandwidth broker operation can cover new nodes simply when the new node (client agent) introduce itself by an appropriate message. Finally, during the process of every request, the base bandwidth broker has access to network modules, as the routing tables (routing information) and uses temporarily (for the process of each specific request) some information from there.

The drawback of this algorithm is that is works based on the current routing schema and does not provides any kind of load balancing that might be necessary when it operates in a large backbone network. In particular, the base bandwidth broker uses the classic OSPF routing protocol that is configured normally (uses the classic Dijstra algorithm that calculates the minimum path without using costs for the edges). Therefore, this module might lead to rejection of requests in case that the basic minimum path is full and alternative paths are not taken into account. This problem can be solved by running an optimization algorithm when the network approaches such situations. This optimization algorithm can run additionally in bandwidth broker's operation, reconfigure periodically the admitted requests and examine again the rejected requests searching for alternative paths. Such an optimization algorithm is in our future plans to implement. The basic idea of the algorithm is to reroute some of the admitted requests from alternative routing paths, when of course the guaranteed bandwidth and delay characteristics are satisfied.

Also, the admission control algorithm exchanges many packets of 64 bytes (from the base bandwidth broker agent to the edge bandwidth broker agents and vice versa) that are crucial for the whole operation. These packets use TCP transport protocol and therefore their transmission is as secure as possible. Also the packets have been marked appropriately to use the high priority QoS service in order to achieve minimum delay and jitter and therefore accelerate the whole operation of the bandwidth broker. The general responding time of the admission control module depends on the request parameters (how far in the topology is the 2 edge nodes of the request) and also on the location of the base bandwidth broker as it coordinates the whole operation. Therefore, in cases where the base bandwidth broker is located on a node that is included in the routing path between the 2 nodes, the packets that should be exchanged traverse less links and the processing time is reduced accordingly. This problem, of the most suitable location of the base bandwidth broker (in that distributed operation), is approaches in section 4, where we propose a model that can select the node that should host it.

3 Description and Testing of Bandwidth Broker's QoS Service

The implemented bandwidth broker manages a QoS service (the IP Premium) that tries to provide bandwidth guarantees as well as minimum delay and jitter. The original ns-2.26 [12] functionality supports a limited number of features for packet classification and queue management, therefore, we have already enhanced the simulator with additional functionality [3][13] in order to simulate the IP premium service's operation. In particular, the classification is done using the DSCP field of the IP header and also we implemented the Modified Deficit Round Robin Scheduling Algorithm (MDRR) [3] and changed the whole queue management mechanism to enqueue packets based on DSCP. The QoS service, as it has been implemented, classifies the packets for each class that has been admitted by the bandwidth broker with DSCP value 46. Then, when the packets are inserted in the network, we apply strict token bucket policy in order to be sure that the transmitted rate agrees with the admitted rate. Next, on all the network nodes, the queue management mechanism is properly configured. The used queue management mechanism is a high priority queue on every node that is used for all the admitted traffic classes. Additionally, instead of priority queueing, the MDRR mechanism can be used.

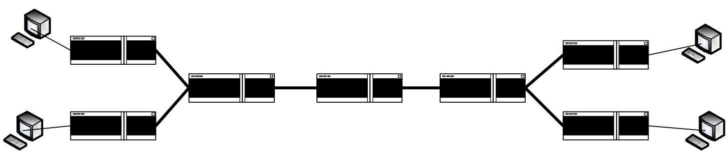
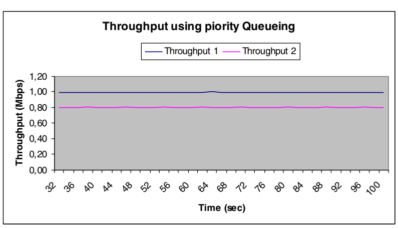


Fig. 1. The network topology

We conducted several tests aiming to evaluate the bandwidth broker's operation when the QoS service (IP Premium) is implemented using the Priority queueing mechanism first and after the MDRR. The topology that was used for those experiments is presented in **Fig. 1**. Each router has an edge client operating locally and also the middle one also contains the base bandwidth broker agent.

3.1 Testing the BB Using the Priority Queueing Algorithm

The bandwidth broker has been configured to manage the IP Premium QoS service, implemented using the priority Queueing as the queue scheduling algorithm. In this case, we performed a set of tests to investigate the operation and finally the guarantees that can provide. For this purpose, the measures that are performed are concentrated on the achieved throughput, delay and jitter. Therefore, we simulated the scenario were the backbone links are all 10Mbps and the bandwidth broker manages 2Mbps on each link for QoS requests. At this point, 2 sources requested 1Mbps and 800Kbps respectively and were successfully admitted by the network as the total bandwidth was available. Finally, the throughput that the 2 flows experienced was exactly the requested and the packet's delay was extremely low.

[1] 

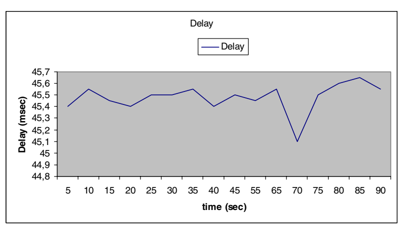
[2] 

Fig. 2. Throughput and Delay using the IP Premium service with Priority Queuing

3.2 Testing the BB Using the MDRR Algorithm

The MDRR is an alternative queue scheduling mechanism that can provide various operations as it has many characteristics. The bandwidth broker has been tested to evaluate the operation of the IP Premium QoS service using MDRR. The topology that was used is again the topology presented in Fig. 1 and the measurements also focus on the achieved throughput, delay and jitter. The final results are the same as in Priority Queuing for the throughput but the delay was measured a little lower as Fig. 3 shows.

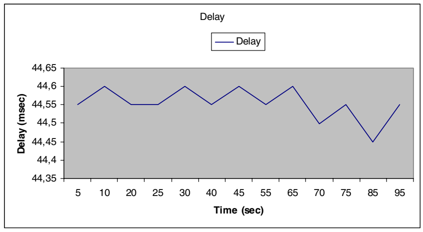
[1] 

Fig. 3. Delay using the IP Premium service with MDRR

Comparing the results from the experiments with the two mechanisms, it is obvious that the bandwidth broker manages very well the IP Premium service that provides the absolute guarantees either with MDRR either with Priority Queueing. The only noticeable difference is that the delay is a little bit smaller when the IP Premium service is provided using the MDRR mechanism. In order to take a decision about the implementation of the IP Premium service and next test it in a backbone network, we should take into account other advantages of the above mechanisms. In this case, the MDRR mechanism seems more powerful than Priority Queueing, due to the fact that except from a high (strict) priority queue, it can support many other queues that can guarantee specific bandwidth (without delay and jitter guarantees).

4 Optimization of Bandwidth Broker's Operation in a Backbone Network

A very important point in the operation of a bandwidth broker is to decide which node should host the base bandwidth broker agent. This decision is more crucial for the efficient operation of the implemented bandwidth broker, when the operation is distributed and the base bandwidth broker agent communicates with all the clients collecting information from the processes that are executed there. In addition, the selection of the location of the base bandwidth broker agent should also take into account the traffic that pass through each node, the importance of each node etc. For this reason, we tried to approach this problem by creating a model that evaluate each node and the adjacent links and according to the weights tries to find the best node to locate the base bandwidth broker. In other words, the problem is to find the root of the graph, where the root is the most important node in the network and most of the packets for the operation of the bandwidth broker will reach it quickly, without passing many links.

This model uses 6 criteria to evaluate the importance of each node in the network operation that are:

- Users. It represents the number of sub-networks and therefore the number of the users that are connected in this node.
- Node equipment. This criterion approaches the capabilities of the specific node. In particular, the grade for this arises from the evaluation of the technology of the routers and the technology and capacity of the backbone links on this router.
- Adjacent nodes. This criterion specifies the importance of the node, taking into account the number of backbone links that are connected on this router.
- Servers. Each node is evaluated by the number of the servers that are connected on it and runs critical and famous services of the network. Except from servers, they can be GRID clusters, VoIP gateways, gatekeepers or any other machine that implies that there is strong possibility for many requests targeted in this node.
- Routing. In this case, the node is evaluated for its importance in the whole routing in the network.
- Interconnection. Finally, the last criterion is used for the condition that this node is an interconnection point with a bigger backbone network and therefore there will be requests from the adjacent bandwidth broker.

Each criterion should be evaluated in the scale from 1 to 10. The evaluation should be done in the same time for all the nodes and the gradation in each one should be analogical. Finally, the weight of each node arises as the sum of all the criteria. In case that there are 2 or more nodes with the same weight, the criteria are taken into account with the following order: Routing, Interconnection, Servers.

Next, for each node, we create the "routing" graph for this node to all the others in the network. In particular, we place each node as root and we create all the paths to all the other nodes, using the network's routing scheme. Therefore, there are N graphs (where N is the number of nodes in the network) that should be examined. Then, we define a new metric for every node, called "special-weight" of node that arises as the weight of this node (that was produced by the above criteria) multiplied with its depth in the graph. In this case, the root of each node has "special-weight" equal to 0. Next, the "special-weight" of the whole graph is the sum of the "special-weight" of all of its nodes. Finally, the problem is to find the graph that has the minimum "special-weight". We run this model for all the nodes, we create all the N graphs and calculate the "special-weight" for each one. Then, we select the graph that has the minimum calculated "special-weight" and the node that is graph's root is the node that must host the base bandwidth broker.

5 Conclusions – Future Work

This paper deals with the Bandwidth broker idea and its operation. It focuses on the distributed admission control algorithm that we implemented, mentioning the advantages and drawbacks that we noticed. Also, the paper describes the IP Premium QoS service that the bandwidth broker manages, which we tested with 2 alternative implementations, using the Priority Queuing and the Modified Deficit Round Robin.

The results showed similar behaviour for both mechanisms and also both achieved the requested guarantees. Finally, we tried to approach the operation of a distributed bandwidth broker in a backbone network where there is specific routing schema and also the nodes have different importance (due to the sub networks and the services that they run). There, the most crucial problem that we faced is where the base bandwidth broker should be located, as it affects the efficiency of its operation. Therefore, we propose a model that evaluates the importance of each node, taken into account several parameters and finally select the most suitable node to host the bandwidth broker.

The simulation tests as well as the algorithms that we propose indicated some points for further investigation. Therefore, we have plans for future work that mainly focuses on the simulation and mathematical evaluation of the proposed “host selection” model in order to optimize it. Also, we plan to study and implement an optimization algorithm that will extend the existing admission control algorithm, in order to provide load balancing.

References

1. RFC 2475 “An Architecture for Differentiated Services”, S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, W. Weiss, December 1998
2. RFC 2905 “AAA Authorization Application Examples”, J. Vollbrecht, P. Calhoun, S. Farrell, L. Gommans, G. Gross, B. de Bruijn, C. de Laat, M. Holdrege, D. Spence, August 2000
3. C. Bouras, D. Primpas, A. Sevasti, A. Varnavas “Enhancing the DiffServ architecture of a simulation environment”, 6th IEEE International Workshop on Distributed Simulation and Real Time Applications, Fort Worth, Texas, USA, October 11 – 13, 2002
4. Manzoor Hashmani and Mikio Yoshida "ENICOM's Bandwidth Broker", Saint 2001 Workshops, pp 213-220, Jan 8-12, 2001, San Diego, USA
5. Bandwidth broker report, University of Kansas
6. QBone Bandwidth Broker Architecture, http://qbone.internet2.edu/bb/bboutline2.html
7. Active Resource Management (ARM) For The Differentiated Services Environment, http://www.caip.rutgers.edu/TASSL/Projects/Adaptive_QoS/ananth/bandwidth.html
8. Przemyslaw Jaskola, Krzysztof Malinowski “Two methods of optimal Bandwidth allocation in TCP/IP networks with QoS differentiation”, 2004 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS’ 04), San Jose, California, USA, July 25 - 29 2004
9. C. Bouras, K. Stamos, “An Adaptive Admission Control Algorithm for Bandwidth Brokers”, 3rd IEEE International Symposium on Network Computing and Applications (NCA04), Cambridge, MA, USA, August 30 - September 1 2004
10. L. Burchard, H. Heiss, “Performance Evaluation of Data Structures for Admission Control in Bandwidth Brokers”, April 2002
11. C. Bouras, D. Primpas, K. Stamos, N. Stathis, "Design and implementation of a Bandwidth Broker in a simulation environment", 7th International Symposium on Communications Interworking - INTERWORKING 2004, Ottawa, Canada, November 29 - December 1 2004
12. http://www.isi.edu/nsnam/ns/ns-build.html
13. http://ouranos.ceid.upatras.gr/diffserv/start.htm

Impact of Traffic Load on SCTP Failovers in SIGTRAN

Karl-Johan Grinnemo[1] and Anna Brunstrom[2]

[1] TietoEnator AB, Lagergrens gata 2, S-651 15 Karlstad, Sweden
karl-Johan.grinnemo@tietoenator.com
[2] Karlstad University, Dept. of Computer Science, S-651 88 Karlstad, Sweden
anna.brunstrom@kau.se

Abstract. With Voice over IP (VoIP) emerging as a viable alternative to the traditional circuit-switched telephony, it is vital that the two are able to intercommunicate. To this end, the IETF Signaling Transport (SIGTRAN) group has defined an architecture for seamless transportation of SS7 signaling traffic between a VoIP network and a traditional telecom network. However, at present, it is unclear if the SIGTRAN architecture will, in reality, meet the SS7 requirements, especially the stringent availability requirements. The SCTP transport protocol is one of the core components of the SIGTRAN architecture, and its failover mechanism is one of the most important availability mechanisms of SIGTRAN. This paper studies the impact of traffic load on the SCTP failover performance in an M3UA-based SIGTRAN network. The paper shows that cross traffic, especially bursty cross traffic such as SS7 signaling traffic, could indeed significantly deteriorate the SCTP failover performance. Furthermore the paper stresses the importance of configuring routers in a SIGTRAN network with relatively small queues. For example, in tests with bursty cross traffic, and with router queues twice the bandwidth-delay product, failover times were measured which were more than 50% longer than what was measured with no cross traffic at all. Furthermore, the paper also identifies some properties of the SCTP failover mechanism that could, in some cases, significantly degrade its performance.

1 Introduction

Since Voice over IP (VoIP) roared into prominence during the latter part of the 1990s, the idea of a converged network based on IP technology for voice, video, and data has gained strong momentum. However, in spite of all prospective advantages with IP it would be naive to think that the transition from the traditional circuit-switched network to IP would happen overnight.

In light of this, the IETF Signaling Transport (SIGTRAN) working group has defined an architecture, the SIGTRAN architecture [1], for seamless Signaling System #7 (SS7) signaling between VoIP and the traditional telecom network. The SIGTRAN architecture essentially comprises two components: a new IP transport protocol, the Stream Control Transmission Protocol (SCTP) [2], specifically designed for signaling traffic; and an adaptation sublayer. The adaptation sublayer shields SS7 from SCTP and IP, and depending on how much of the SS7 stack is run atop SCTP, different adaptation protocols are used. Examples of adaptation protocols include: M2PA [3] for adaptation of the SS7

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 774–783, 2005.

MTP-L3 [4] protocol to IP, and M3UA [5] for adaptation of SCCP [6] and user part protocols such as ISUP [7].

It is widely recognized that to gain user acceptance, the SIGTRAN architecture has to perform comparable to the traditional circuit-switched telecom network [8]. In particular, it has to provide the same level of availability as a traditional SS7 network. Considering that ITU-T prescribes an availability level of 99.9988% [9], i.e., no more than 10 minutes downtime per year, and that many telecom networks have an even higher availability level [10], this is indeed a great challenge.

To meet the stringent requirements of SS7, several availability mechanisms have been included in the SIGTRAN architecture of which the SCTP failover mechanism is one of the more important ones – if not the most important one. It corresponds with the MTP-L3 changeover procedure, and enables rapid re-routing of traffic from a failed signaling route within a SIGTRAN network. In particular, the SCTP failover mechanism constitutes part of SCTPs multi-homing support.

Although, the SCTP failover mechanism plays a key role in the availability support of the SIGTRAN architecture, very few results are available on its actual performance in this context. Jungmaier et al. [11] have studied the SCTP failover performance in an M2PA-based network, and showed that it only meets ITU-T requirements provided it is configured very aggressively, and provided the network path propagation delays are very short. A similar result was also obtained by Grinnemo et al. [12] when they performed measurements on SCTP failover performance in an M3UA-based network.

Both the study in [11] and in [12] took place in unloaded networks, i.e., under quite unrealistic conditions. This paper advances the work in [12], and partly the work in [11], by studying the impact of traffic load on the SCTP failover performance in an M3UA-based SIGTRAN network. The main contribution of the paper is that it demonstrates that cross traffic, especially bursty cross traffic such as SS7 signaling traffic, could indeed significantly deteriorate the SCTP failover performance. Furthermore, the paper stresses the importance to keep the router queues in a SIGTRAN network relatively small. In fact, the paper shows that bursty traffic in combination with ill-dimensioned router queues may well cause the SCTP failover mechanism to not comply with the ITU-T requirement on the MTP-L3 changeover procedure [9]. Furthermore, the paper identifies some issues regarding the design of the SCTP failover mechanism which in some cases negatively affect the failover performance.

The remainder of the paper is organized as follows. Section 2 gives a brief description of the SCTP failover mechanism. Then, in Section 3 follows a description of the design and execution of the experiment that underlies our study. Next, in Section 4, we elaborate on the results of the experiment. Finally, in Section 5, the paper ends with some concluding remarks and words on future work.

2 Failovers in SCTP

While IP networks have many virtues, high availability and reliability have traditionally not been seen as two of them. Unlike circuit-switched paths, which exhibit changeover and failover times on the order of milliseconds, measurements show that it may take

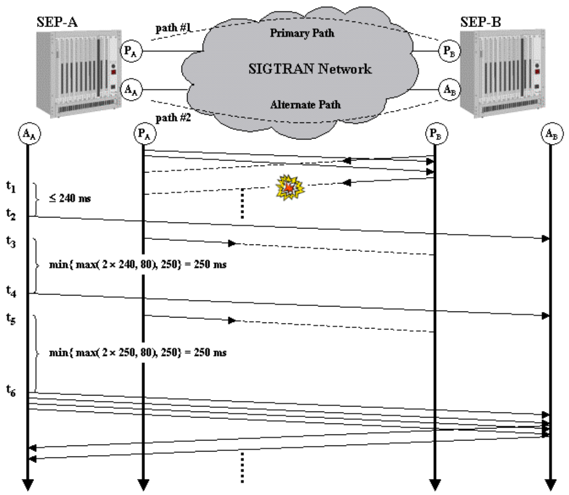


Fig. 1. Failover scenario between two dual-homed signaling endpoints

well over ten seconds before the routers in the Internet reach a consistent view after a path failure [13] – in other words, too long for delay-sensitive SS7 signaling traffic.

In the SIGTRAN architecture, the unsuitability of IP for high-availability routing of SS7 signaling messages is addressed through various redundancy mechanisms at the transport and adaptation layers. As previously mentioned, one of the most important network redundancy mechanisms in SIGTRAN is the SCTP failover mechanism.

An example of how the SCTP failover mechanism works is illustrated in Figure 1. In this example, we have an SCTP connection, a so-called association, between two signaling endpoints: SEP-A and SEP-B. The association comprises two routing paths: path #1 and path #2. Since SCTP does not support load-sharing, one path in an association is always designated the primary path and is the path on which signaling traffic is normally sent. The remaining paths, if any, become backup or alternate paths. In our example, path #1 is the primary path and path #2 an alternate path.

SCTP continuously monitors reachability on the primary and alternate paths – on an active primary path SCTP probes for reachability using the transferred data packets themselves, and on idle alternate paths specific heartbeat packets are used. Furthermore, for each path (actually network destination), SCTP keeps an error counter that counts the number of consecutively missed acknowledgements to data or heartbeat packets. A path is considered unreachable when the error counter of the path exceeds the value of the SCTP parameter `Path.Max.Retrans`. In the remaining discussion, it is assumed that the SCTP stacks at SEP-A and SEP-B are configured with `Path.Max.Retrans` set to 2.

As follows from the time line in Figure 1, a failure occurs on the primary path at time t_1. At that time, the SCTP retransmission timeout (RTO) variable is assumed to be 240 ms, and it is assumed that there are outstanding traffic. Thus, at $t_2 \leq t_1 + 240\,ms$, the SCTP retransmission timer, T3-rtx, expires and a timeout occurs; an SCTP packet worth of outstanding data is retransmitted on the alternate path, and the error counter of the primary path is incremented by one. Furthermore, the RTO variable is backed off, or more precisely

$$RTO \leftarrow min\left\{max\left(2 \times RTO_{cur}, RTO_{min}\right), RTO_{max}\right\}, \tag{1}$$

where RTO_{cur} denotes the current value of the RTO variable, and RTO_{min} and RTO_{max} are SCTP parameters that limit the range of the RTO variable. Here, it is assumed that RTO_{min} is set to 80 ms and RTO_{max} to 250 ms.

At time t_3, new data is sent out on the primary path, and the T3-rtx timer is restarted with the value of the updated RTO variable. The flow of events that occurred at times t_2 and t_3 are repeated at times t_4 and t_5. When time t_6 is reached, the error counter of the primary path becomes 3, i.e., greater than `Path.Max.Retrans`, and SCTP considers the path failed and starts sending new data onto the alternate path. In other words, the failover concludes.

3 Methodology

To be able to study the impact of traffic load on the SCTP failover performance, we considered the network scenario depicted in Figure 2.

In this scenario, two M3UA users at signaling endpoints SEP1 and SEP2 were engaged in a signaling session over a SIGTRAN network with varying degrees of traffic load. The session took place over a multi-path association with one primary and one alternate path. Initially, all signaling traffic in the M3UA session was routed on the primary path. However, 30 s into the signaling session a failure occurred on the primary path. As a result, the signaling traffic was re-routed from the primary to the alternate path. The network scenario ended when 90 s had elapsed from the time of the path failure.

The network scenario in Figure 2 was modeled using the experiment setup illustrated in Figure 3. The M3UA session between SEP1 and SEP2 was modeled as a constant bit rate flow of 200 Kbps. Although it could be argued that a constant bit rate flow is not a particularly realistic model of actual SS7 traffic [14], a more realistic model would make

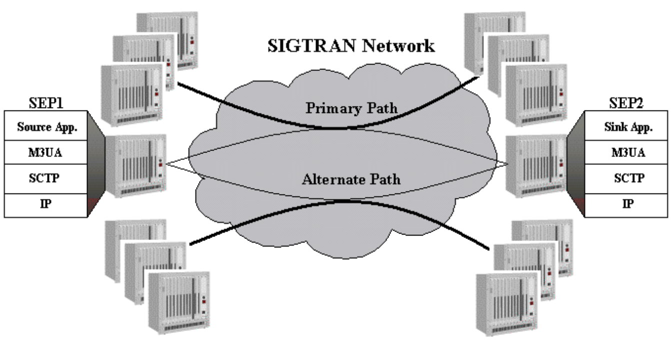


Fig. 2. Studied network scenario

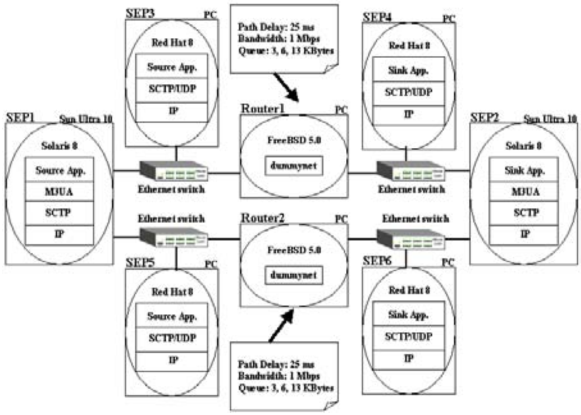


Fig. 3. Experiment setup

Table 1. Cross Traffic Characteristics

Name	Burst Size (KBytes)	Inter-Burst Gap (ms)
CT-NONE	0	0
CT-LOW	4	200
CT-MEDIUM	8	100
CT-HIGH	16	50

it much more difficult to measure the failover times. Particularly, introducing randomness in the traffic generation at SEP1 would render it difficult to establish the start times of the failovers.

The cross traffic comprised single SCTP flows between SEP3 and SEP4, and SEP5 and SEP6. Since the SS7 traffic in future dedicated SIGTRAN networks will presumably be bursty [14, 15], the cross traffic was generated as bursty flows. Tests were run for a range of cross traffic flows representing a spectrum of traffic loads with different degrees of burstiness. Specifically, tests were run with cross traffic flows having burst sizes and inter-burst gaps as listed in Table 1. It should be noted that CT-NONE denotes no cross traffic at all, and that the CT-HIGH cross traffic case actually meant that the SEP1 source application did not impose any limits on the SCTP transmission rate.

To be able to study the impact of queueing delay on the SCTP failover performance, tests were run with three different router queue sizes: 3 Kbytes (approximately half the bandwidth-delay product), 6 KBytes (approximately the same as the bandwidth-delay product), and 13 KBytes (approximately twice the bandwidth-delay product). These queue sizes were selected with the intent to model the router configurations found in both controlled, delay-sensitive, networks, and uncontrolled networks.

The SCTP stacks at SEP1 and SEP2 were configured to meet the ITU-T requirements on the MTP-L3 changeover procedure [9], i.e., according to the findings in [11, 12]. More precisely, they were configured as shown in Table 2, with the remaining parameters set as

Table 2. SCTP configuration

Parameter	Setting
RTO_{init}	250 ms
RTO_{min}	80 ms
RTO_{max}	250 ms
Path.Max.Retrans	2
SACK timer	40 ms

recommended in RFC 2960 [2]. The SCTP stacks at the remaining SEPs were configured in accordance with RFC 2960.

Tests were run for all combinations of cross traffic and router queue sizes, giving a total of 12 tests. Furthermore, to obtain statistical validity each test was repeated 40 times.

4 Results

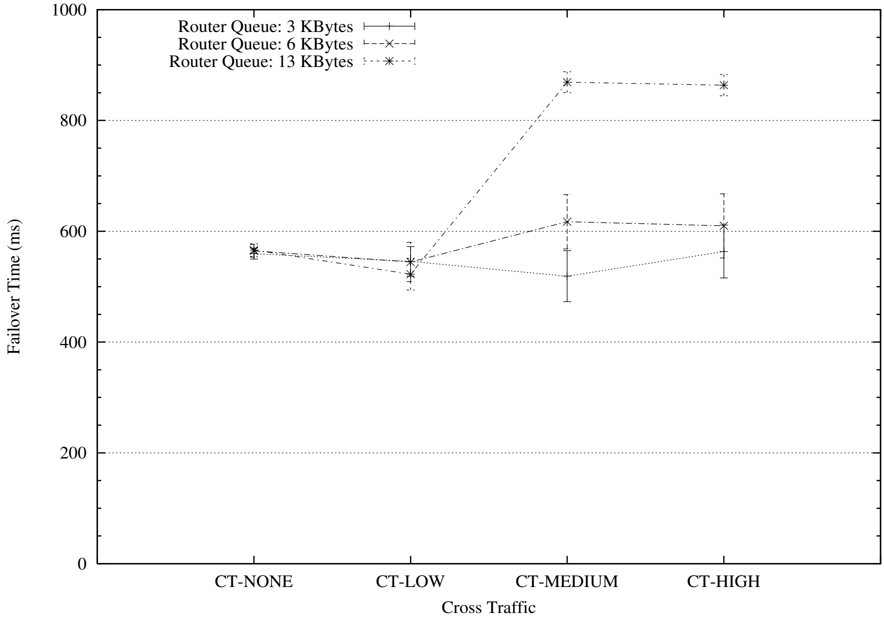
The SCTP failover performance was evaluated in terms of two metrics: the failover time experienced by the SEP1 source application, and the maximum Message Signal Unit (MSU) transfer time measured during failover in the M3UA session between SEP1 and SEP2. As estimates of the failover times and the max. MSU transfer times in the tests, the sample means were used.

Figure 4 summarizes the result of our experiment. In Figure 4 (a), it is shown how the SCTP failover time was affected by increasing traffic load at different router queue sizes, while Figure 4 (b) shows the same relationship for the max. MSU transfer time. The error bars depict the 99% confidence intervals, and the lines connecting the mean failover times and max. MSU transfer times are only supplied as a visualization aid. Specifically, these lines are only included to help visualize the trends.

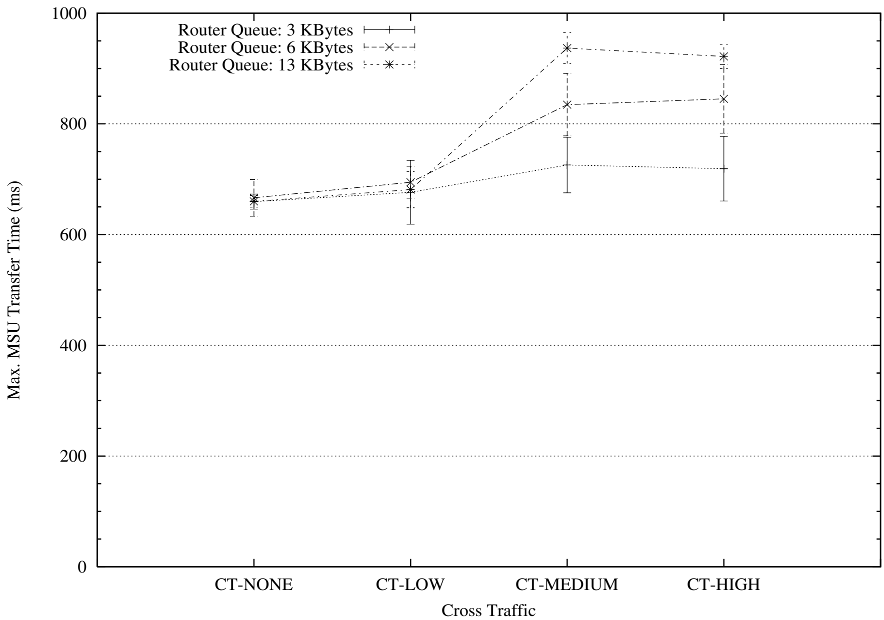
As follows from Figure 4, the deteriorating effect of the cross traffic on the failover performance increased with increased traffic load and router queue size. When the Router1 queue was only 3 KBytes, the cross traffic did not inflict significantly on the failover and max. MSU transfer times. However, as the queue size was increased, the effect of the cross traffic became more and more apparent. Thus, when the Router1 queue was 13 KBytes, the CT-HIGH cross traffic increased the failover time with more than 50% and the max. MSU transfer time with almost 40% as compared with no cross traffic at all.

The reason to the increased failover and max. MSU transfer times was the queueing delays that arose at Router1 when the router queue was fairly large, and when the cross traffic was bursty (i.e., when the short-term bandwidth requirement of the cross traffic sometimes exceeded the bandwidth capacity of the primary path). As a matter of fact, in previous tests with the same test flow, but with constant bit rate cross traffic, it was found that the traffic load had no significant impact on the failover performance provided it was less than the path capacity.

Another observation worth making concerns the SCTP failover times with regards to the requirement of ITU-T on the MTP-L3 changeover procedure [9]. To comply with



(a) Failover Time vs. Cross Traffic.



(b) Max. MSU Transfer Time vs. Cross Traffic.

Fig. 4. Impact of traffic load on SCTP failover performance

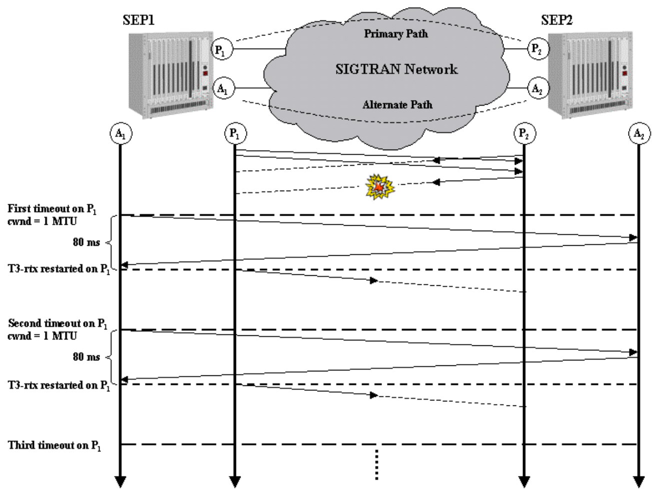


Fig. 5. Management of the T3-rtx timer during failover

this requirement, the SCTP failover times should be no more than 800 ms. However, as follows from Figure 4, this requirement was only fulfilled in those cases the Router1 queue was relatively small (3 KBytes or 6 Kbytes). In the tests with a router queue of 13 KBytes or twice the bandwidth-delay product (to our knowledge a quite common configuration [16]), the failover times averaged well above 850 ms at medium (CT-MEDIUM) and high (CT-HIGH) traffic loads.

Interestingly, in all tests, the measured failover times were significantly larger than what could be expected given the RTOs. However, the discrepancy was larger with larger traffic loads and router queues. Consider, for example, the test with a 13 KByte Router1 queue and the CT-HIGH cross traffic. When this test was re-ran with tracing on the RTO, the RTO at the time of the path failure, RTO_t, was measured to 240 ms. Only considering the timeout periods, this gives us a theoretical failover time of $240\,ms + 250\,ms + 250\,ms = 740\,ms$ (see Section 2). However, the measured failover time was 920 ms, or 180 ms larger than our estimate.

The reason to this discrepancy turned out to be substantial delays between the expiration of the T3-rtx timer and its restart during the failover (see Figure 5). When a timeout occurred, the SCTP congestion window at SEP1 was reduced to 1 Maximum Transmission Unit (MTU). As a result, no packets were sent out on the primary path, and the T3-rtx timer was not restarted, until the amount of outstanding data went below 1 MTU. This meant, as shown in Figure 5, an extra delay (apart from the timeout delay) of about 80 ms at each timeout event.

Although, an extra delay of 80 ms at each timeout during a failover has to be considered as a quite large delay in this context (SS7 signaling), even larger delays could be expected in real-world SIGTRAN networks. Specifically, it could take several transmission rounds before the T3-rtx timer of the primary path is restarted again after a timeout in cases with large amounts of outstanding data at the time of a path failure.

Finally, as an aside, we would like to mention the significant penalty in terms of failover performance that could be the result of setting RTO_{init}, the initial value of RTO, too low. Specifically, a too low value on RTO_{init} with respect to the round-trip time of the alternate path[1] could result in one extra retransmission, and thus one extra timeout period, before SCTP considers the primary path failed. To gain some appreciation of the extent to which this could in fact impede on the failover performance in a SIGTRAN network, we re-ran the test with the Router1 queue set to 13 KBytes and with no cross traffic (CT-NONE), but this time with RTO_{init} at SEP1 and SEP2 configured to 80 ms instead of 250 ms. The result of this test was that we observed an increase in failover time with about 180 ms, or 32%, compared with the original test (cf. Figure 4 (a)).

5 Conclusions

This paper studies the impact of traffic load on the SCTP failover performance in an M3UA-based SIGTRAN network. Two performance metrics are considered: the SCTP failover time, and the maximum transfer time experienced by an M3UA user during failover. The paper shows that cross traffic, especially bursty cross traffic such as SS7 signaling traffic, could indeed significantly deteriorate the SCTP failover performance. Furthermore, the paper demonstrates how important it is to configure the routers in a SIGTRAN network with relatively small queues. For example, in tests with bursty cross traffic and with router queues twice the bandwidth-delay product (to our knowledge a quite common configuration), failover times were measured which on the average were more than 50% longer than what was measured with no cross traffic at all. In fact, in these situations, our study suggests the SCTP failover performance may not even meet the requirement of ITU-T on MTP-L3 changeovers.

Two important observations are also made in the paper which concern the SCTP failover behavior. First, it is shown that the delays which occur in between the expiration of the SCTP retransmission timer (T3-rtx) and its restart during a failover could contribute significantly to the failover and max. MSU transfer times. Second, the paper comments on the extent to which a too low initial retransmission timeout (RTO) value, i.e., a too low value on the SCTP parameter RTO_{init}, could deteriorate the failover performance.

While cross traffic, T3-rtx restart delays, and low values on RTO_{init} could have a significant negative effect on the SCTP failover performance, it still remains that a major factor is the length of the timeout periods. Thus, in our future work, we intend to study ways of shortening these periods without threatening network stability. Specifically, we intend to study the effect of introducing a more relaxed RTO backoff mechanism.

[1] Note that the first transmission round on the alternate path within a timeout period only comprises a single SCTP packet. Consequently, the SACK timer delay adds to the initial round-trip time in a timeout period on the alternate path, something that is easily overlooked when RTO_{init} is configured.

References

1. Ong, L., Rytina, I., Garcia, M., Schwarzbauer, H., Coene, L., Lin, H., Juhasz, I., Holdrege, M., Sharp, C.: Framework architecture for signaling transport. RFC 2719, IETF (1999)
2. Stewart, R., Xie, Q., Morneault, K., Sharp, C., Schwarzbauer, H., Taylor, T., Rytina, I., Kalla, M., Zhang, L., Paxson, V.: Stream control transmission protocol. RFC 2960, IETF (2000)
3. George, T., Bidulock, B., Dantu, R., Schwarzbauer, H.J., Morneault, K.: Signaling system 7 (SS7) message transfer part 2 (MTP2) - user peer-to-peer adaptation layer (M2PA). Internet draft, IETF (2004) draft-ietf-sigtran-m2pa-12.txt.
4. ITU-T: Specifications of signalling system no. 7 - message transfer part: Signalling network functions and messages. ITU-T Recommendation 704, ITU (1996)
5. Sidebottom, G., Morneault, K., Pastor-Balbas, J.: Signaling system 7 (SS7) message transfer part 3 (MTP3) - user adaptation layer (M3UA). RFC 3332, IETF (2002)
6. ITU-T: Specifications of signalling system no. 7 - signalling connection control part: Signalling connection control part procedures. ITU-T Recommendation 714, ITU (1996)
7. ITU-T: Specifications of signalling system no. 7 - ISDN user part: ISDN user part signalling procedures. ITU-T Recommendation 764, ITU (1997)
8. Fu, S., Atiquzzaman, M.: SCTP: State of the art in research, products, and technical challenges. IEEE Communications Magazine **42** (2004) 64–76
9. ITU-T: Specifications of signalling system no. 7 - message transfer part signalling performance. ITU-T Recommendation 706, ITU (1993)
10. Kuhn, R.: Sources of failure in the public switched telephone network. IEEE Computer **30** (1997) 31–36
11. Jungmaier, A., Rathgeb, E.P., Tuexen, M.: On the use of SCTP in failover scenarios. In: 6th World Multiconference on Systemics, Cybernetics and Informatics (SCI), Orlando, Florida, USA (2002)
12. Grinnemo, K.J., Brunstrom, A.: Performance of SCTP-controlled failovers in M3UA-based SIGTRAN networks. In: Advanced Simulation Technologies Conference (ASTC), Applied Telecommunication Symposium (ATS), Arlington, Virginia, USA (2004) 86–91
13. Labovitz, C., Ahuja, A., Bose, A., Jahanian, F.: Delayed Internet routing convergence. IEEE/ACM Transactions on Networking **9** (2001) 293–306
14. Scholtz, F.J.: Statistical analysis of common channel signaling system no. 7 traffic. In: 15th Internet Traffic Engineering and Traffic Management (ITC) Specialist Seminar, Wurzburg, Germany (2002)
15. Andersen, A.T.: Modelling of packet traffic with matrix analytic methods. PhD thesis, Informatics and Mathematical Modelling, Technical University of Denmark, DTU (1995)
16. Allman, M.: TCP byte counting refinements. ACM Computer Communications Review **3** (1999)

A Novel Method of Network Burst Traffic Real-Time Prediction Based on Decomposition

Yang Xinyu, Shi Yi, Zeng Ming, and Zhao Rui

Dept. of Computer Science and Technology, Xi'an Jiaotong University,
710049 Xi'an, P.R.C
yxyphd@mail.xjtu.edu.cn

Abstract. Network traffic burst becomes a threat to network security. In this paper, a decomposition based method is presented for network burst traffic real-time prediction, in which, by passing smoothing filter, network traffic is decomposed into smooth low frequency traffic and high frequency traffic to make prediction respectively, and then a superposition result of the predictions is yielded. Based on LMS algorithm, an improvement of LMS predictor by adjusting prediction according to prediction errors (EaLMS, Error-adjusted LMS) is proposed to process the low frequency traffic, and a simple method of linear combination is presented to predict the high frequency traffic. The experiment results using real network traffic data shows, compared with traditional LMS, the prediction method based on decomposition obviously shorted the prediction delay and reduced the prediction error during traffic burst, while it also improves the global prediction.

1 Introduction

Network traffic burst, which is probably caused by network attack such as DDoS (Distributed Denial of Service), may result in network congestion or even collapse, and become a serious threat to network security. The improvement of real-time burst traffic prediction will accordingly contribute more to network security.

Traffic prediction is an important research field of the traffic engineering. Recent work in this area mainly includes using time series analysis model [1], artificial neural-network method [2-3], wavelet method [4], etc. Relative to long-term prediction based on periodical model, short-term real-time prediction shows much more importance in network traffic monitor, congestion control, and attack detection. For short-term real-time prediction, efficient adaptive methods are needed. Among them, least-mean-square (LMS) algorithm is of particular interest [5-7] due to its simplicity and reliability and relatively good performance in dealing with real-time signals.

As commonly considering, because of the compromise between convergence speed and tracking performance, LMS is better in dealing with stationary signals prediction. While applying LMS to traffic prediction, the problem of the compromise between prediction delay and prediction error is especially serious for sharp fluctuation of network traffic. On the one hand, a larger step size will reduce prediction delay, but

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 784–793, 2005.

bring the problem of convergence that leads to increasing prediction error; on the other hand, a smaller step size gives less prediction error but a longer prediction delay, especially the severe delay will occur during traffic burst. While, for being predictable, LPF (Low Pass Filter) in general decrease the random high frequency disturb to reduce the change of traffic, and prediction according to the filtered traffic can achieve better performance [8]. For improving the result of prediction, an idea is proposed, in which network traffic is filtered and decomposed into two parts, the low frequency traffic and the high frequency part that changed fast. An improved LMS algorithm is adopted to process the former; the latter is deal with other prediction algorithm. Decomposing is a common means towards non-stationary signal prediction [9] and we use smoothing filter for decomposition in this paper because short-term prediction needs tracing traffic real-time varying.

The low frequency traffic, obtained by smoothing filtering, preserves the main characteristic of original traffic, and is relatively more stable and more suitable for applying LMS predictor. An improved LMS algorithm by adjusting prediction according to prediction errors, which is called EaLMS (Error-adjusted LMS, proposed by ourselves in a previous paper [10]), is presented to process the low frequency traffic in this paper. EaLMS does not make any modification to LMS algorithm, but only adds a small adjustment to LMS prediction result, and this adjustment is a function of previous prediction errors. Experiment based on low frequency traffic of real network trace has proved that for short-term real-time prediction, compared with traditional LMS predictor, EaLMS significantly reduces prediction delay and prediction error at the same time. A simple linear combination method is introduced to process the high frequency traffic, because the LMS algorithm cannot meet the challenge of the prediction of high frequency traffic, that is, the fast change of traffic have bad influences on its convergence. Compared with directly prediction with traditional LMS, the result of network traffic experiment using the prediction method based on decomposition show that the traffic burst prediction delay and errors were improved obviously.

The paper is organized as follows. Section 2 introduces the main idea and the process of the decomposition prediction algorithm. Section 3 describes EaLMS as to low frequency traffic predicting and an analysis of prediction experiment, and Section 4 discusses a simple method of linear combination to predict the high frequency traffic. Section 5 compares the results of directly prediction with LMS and prediction based on decomposition, and conclusions are in section 6.

2 An Idea of Traffic Prediction Based on Decomposition

The basic idea of the prediction based on decomposition is that the traffic is divided into different ranges though filter, each range will be predicted by adopting corresponding methods, and finally all predict results are superposed. In the experiment of this paper, the filter used for decomposing is a mean smooth filter (its order is 10); low frequency traffic and high frequency part will be acquired. The

EaLMS algorithm and a method of linear combination are used to predict respectively. The flow of the decomposition prediction shows in Fig.1.

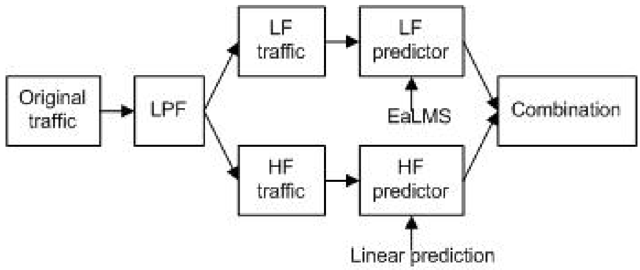


Fig. 1. The flow of the decomposition prediction

The data used by the experiment is the network traffic record trace LBL-PKT-4 in Internet Traffic Archive [11], and the length of time is 3600s. After calculating the statistic characteristics of this trace, the time serials of traffic per second (Bytes/s) is acquired, denoted as *w(t).* Then through 10-orders smoothing filters, the low frequency traffic and high frequency traffic in 1 second are acquired, and described by *z(t)* and *y(t)* respectively. See formula (1) and Fig.2 below.

$$z(t)=\begin{cases} w(t), & t\le k-1 \\ \dfrac{1}{k}\displaystyle\sum_{i=t-(k-1)}^{t} w(i), & t>k-1 \end{cases} \qquad y(t)=w(t)-z(t) \tag{1}$$

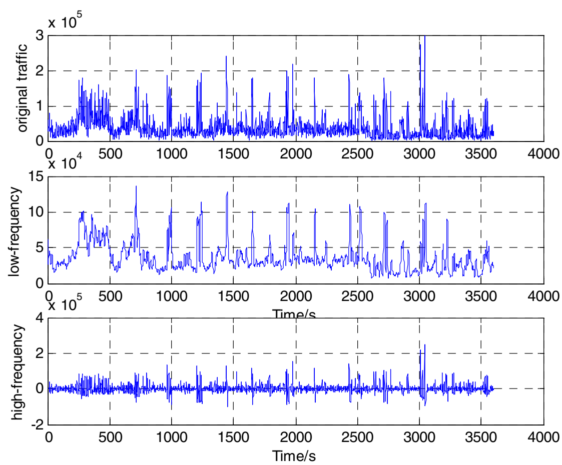


Fig. 2. The network traffic of *w(t)* in LBL-PKT-4 (top), the low frequency traffic after passing smoothness filter *z(t)* (middle) , high frequency traffic *y(t)* (bottom)

3 Low Frequency Traffic Prediction-EaLMS

The time serials in real world always have characteristics of sharp fluctuation and sudden burst. After smoothly filtering, the low frequency traffic had good stability, decreased the random disturb, minimized the variance of traffic, so can achieve better performance of prediction. Although applying LMS algorithm to the low frequency traffic can do better, the conflict between prediction delay and error stood out during traffic prediction because of the contradiction between convergence and stability maladjustment of adaptive algorithm. For this reason, we adopt an improved method EaLMS that is proposed by ourselves before [10]. In EaLMS, LMS algorithm itself is not modified and just improved by adjusting the prediction value of LMS algorithm according to prediction errors.

3.1 An Introduction to LMS

LMS is one of the most popular algorithms in adaptive signal processing, which was proposed by B.Widrow. The algorithm is of the form,

$$\hat{\mathbf{w}}(n+1) = \hat{\mathbf{w}}(n) + \frac{1}{2}\mu[-\hat{\nabla}(n)] = \hat{\mathbf{w}}(n) + \mu e(n)\mathbf{x}(n) \tag{2}$$

If LMS is applied to prediction with adaptive AR(p) model [12], the algorithm is on form,

$$e(t) = x(t) - \varphi^t(t)\mathbf{x}(t-1) \tag{3}$$

$$\varphi(t+1) = \varphi(t) + \mu\mathbf{x}(t-1)e(t) \tag{4}$$

Where $\varphi(t) = [\varphi_1, \varphi_2, ..., \varphi_p]^t$

$\mathbf{x}(t-1) = [x(t-1), x(t-2), ..., x(t-p)]^t$

Here μ is the step size. In standard LMS, μ is a constant and its value determines the speed of adaptive process. The condition of convergence is $0<\mu<\lambda_{max}$, and λ_{max} is the max eigenvalue of correlation matrix R. The initial value of the parameter matrix is 0 in general.

The tracing speed of LMS method is controlled by step size: the larger μ is, the faster the convergence speed is. However, an excessive μ can affect the convergence of algorithm and will augment steady state misjudgment. For prediction application, this drawback becomes tradeoff between prediction delay and prediction error. A larger step size will reduce prediction delay, but also brings problem of convergence that leads to increasing prediction error; otherwise, a smaller step size gives less prediction error but with a longer prediction delay. To solve the conflict between learning speed and steady state misjudgment, many improved LMS algorithms are proposed, such as varying step LMS and transform-domain LMS [13]. VSS-LMS [14], proposed by Kwong and Johston, is a typical method of first kind of improvement. VSS-LMS use a variable step size to reduce the tradeoff between maladjustment and tracking ability of the fixed step size LMS. But one inconvenience of VSS-LMS is that we have to designate the parameter values artificially.

3.2 EaLMS for Network Low Frequency Traffic

The objective of EaLMS is mainly for network low frequency traffic prediction. The advancement of EaLMS is to add a variable -- $\varepsilon(t)$, which is a function of prediction error $e(t)$, to the LMS prediction result. Thus, the key problem of EaLMS is how to compute $\varepsilon(t)$.

We estimate the traffic variation trend according to sign continuity and absolute value of $e(t)$. Adjustment $\varepsilon(t)$ is added to the LMS prediction value, so that the new predictor could follow the variation of traffic more quickly, or even forecast it in advance. The adjustment quantity is determined by two elements -- absolute value of $e(t)$ and its sign continuity, i.e. it's the product of two factors -- $sign(t)$ and $value(t)$ as

$$\varepsilon(t) = sign(t) * value(t) * e(t) \tag{5}$$

Where,

$$sign(t) = \begin{cases} 1 & n(t) = 1 \\ 2 * P\{N \geq n(t) + 1 \mid N \geq n(t)\} & n(t) > 1 \end{cases} \tag{6}$$

Definition: $n(t)$ -- the count of $e(t)$ which has the continuously same sign at t moment. $P\{N \geq n(t) + 1 \mid N \geq n(t)\}$ is conditional probability of the corresponding circumstance.

$$value(t) = \min(|e(t)| / \sigma_e(t)\ ,\ \sigma_e(t)) \tag{7}$$

For one-step prediction, correction is made by adding the product of adjustment quantity $\varepsilon(t)$ and standard deviation $\sigma(t)$, to the LMS prediction result $za1(t+1)$. The corrected value, noted as $zb1(t+1)$, is the one-step EaLMS prediction result. Here, to multiply $\sigma(t)$ corresponds to the normalization before applying LMS algorithm.

$$zb1(t+1) = za1(t+1) + \sigma(t) \cdot \varepsilon(t) \tag{8}$$

As for multi-step prediction, average of adjustment quantity ε at several previous moments, is used as an estimation of $\varepsilon(t+1)$. For example, the adjustment quantity for two-step is,

$$zb2(t+2) = za2(t+2) + \sigma(t) \cdot \hat{\varepsilon}(t+1) \tag{9}$$

Here, $\hat{\varepsilon}(t+1) = \frac{1}{4} \cdot [2 \cdot \varepsilon(t) + \varepsilon(t-1) + \varepsilon(t-2)]$

3.3 Analysis of Experiment Results

Applying LMS and EaLMS to the low frequency traffic filtered from LBL-PKT-4 by smoothing filter, the aspects to be compared include: a). Global prediction performance: the comparison of global prediction error; b). Burst prediction performance: comparison of prediction delay and prediction error when traffic bursts; and prediction error is calculated by root mean square error (RMSE).

A. Global Prediction Performance

Prediction interval of LBL-PKT-4 chooses (500, 3600). As shown in Tab.1, compared with the LMS prediction, RMSE of the EaLMS prediction is respectively reduced 27.1%, 18.9% and 15.8% as to one-step, two-step and three-step. The average value is also decreased 20.6%. That is to say, the global prediction error is decreased after adjusting the errors.

Table 1. RMSE of LBL-PKT-4 global(500, 3600) with EaLMS prediction and LMS prediction for low frequency traffic (10^3)

Step size	1	2	3
LMS prediction error	4.5780	7.3849	10.187
EaLMS prediction error	3.3341	5.9890	8.5731

B. Burst Prediction Performance

Comparison of burst prediction is to examine the response speed, i.e. prediction delay of two predictors, especially at the burst moments. The evaluation also includes prediction error as well. In Fig.3, the prediction step from up to down is one-step, two-step and three-step; z, zax and zbx represents respectively the real value, the LMS prediction value and the EaLMS prediction value.

Choose LBL-PKT-4 prediction interval (1180, 1280). There are two successive bursts and a flat period after the first burst. According to Fig.3, EaLMS method reduces prediction delay obviously; while from Tab.2, compared with the LMS prediction, RMSE of the EaLMS prediction is respectively reduced 41.3%, 32.1% and 27.6% as to one-step, two-step and three-step. The average value is also decreased 33.7%. Note that at the moments where traffic changes its variation trends (around 1203s, 1224s, etc.), EaLMS predictor has a larger error, especially in multi-step prediction, which is due to the inertial effect of adjustment.

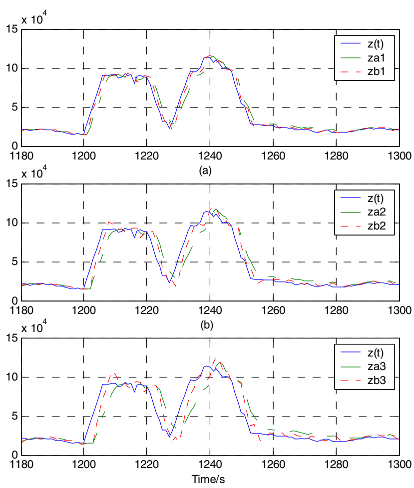


Fig. 3. Comparison of the EaLMS prediction with LMS prediction for low frequency traffic in LBL-PKT-4 burst (1180, 1280)

Table 2. RMSE of LBL-PKT-4 burst(1180, 1280) with EaLMS prediction and LMS prediction for low frequency traffic (10^3)

Step size	1	2	3
LMS prediction error	8.5963	13.511	18.312
EaLMS prediction error	5.0464	9.1726	13.258

4 High Frequency Traffic Prediction-Linear Combination

Applying LMS algorithm to the high frequency traffic acquired after passing smoothing filter will overcome the characteristic of sharp fluctuation, because the algorithm cannot converge (if step is too large) or trace the ratio of variance and the conflicts between prediction delay and error stand out (if step is too small). In Fig.2, the high frequency traffic is similar to a random serial, average of which is zero, but its energy values are still considerable, especially in traffic burst, and have an obvious influence on the global traffic change and cannot be ignored.

It is difficult to find an efficient predictor for high frequency traffic due to the occasionally of traffic burst. A simple linear combination method is adopted, that is, the average of weighted high frequency traffic in several point is used as the predicted value of the next, such as one step predicted value, $\hat{y}(t+1)=\varphi_1 y(t)+\varphi_2 y(t-1)+\varphi_3 y(t-2)$, where φ1, φ2 and φ3 are constant, equal to the parameters in AR model, and φ1=0.5, φ2=0.1 and φ3=0.01 for reducing the prediction delay.

Prediction interval (2500, 2560) is select to evaluate predict results. As showed in Fig.4, prediction result is similar to the real traffic with one delay by using linear combination method. Tab.3 describes RMSE of linear combination prediction for high frequency traffic.

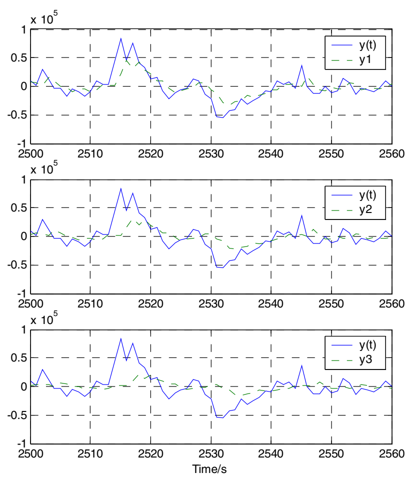


Fig. 4. The result of linear combination prediction for high frequency traffic

Table 3. RMSE of LBL-PKT-4 burst(2500,2560) with linear combination prediction for high frequency traffic (10^4)

Step size	1	2	3
Prediction error	1.6768	2.0408	2.2779

5 Prediction Results After Combination

Superposed the results of the two prediction experiments above, low frequency traffic and high frequency traffic are combined to acquire the prediction traffic. The RMSE,

Table 4. RMSE of LBL-PKT-4 global (500,3600) with two different prediction methods (10^4)

Step size	1	2	3
Directly prediction error	2.3623	2.8066	2.9716
Decomposition prediction error	2.3108	2.6980	2.9242

Table 5. RMSE of LBL-PKT-4 burst (2500,2560) with two different prediction methdos (10^4)

Step size	1	2	3
Directly prediction error	1.9298	2.5965	3.0699
Decomposition prediction error	1.7499	2.1958	2.5982

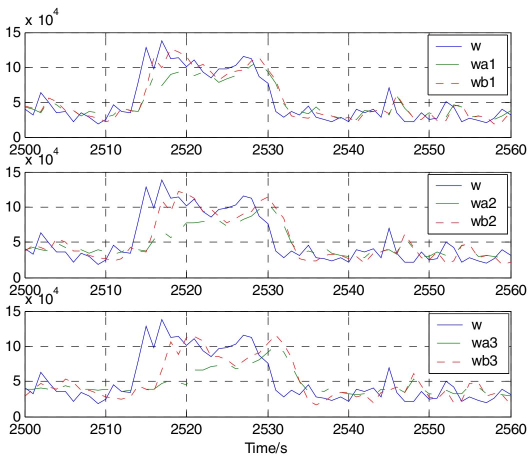


Fig. 5. Comparison of directly LMS prediction with decomposition prediction in LBL-PKT-4 burst (2500, 2560)

in the global prediction interval of LBL-PKT-4 (500, 3600), is compared between the directly prediction and the decomposition prediction, see in Tab.4. In Tab.5, there's the result in the burst prediction interval of LBL-PKT-4 (2500, 2560).

We can see from Tab.4 that the two prediction results are almost the same according to the global prediction. But as to the result of the burst prediction interval LBL-PKT-4 (2500, 2560) in Tab.5, the decomposition prediction is obviously better, because RMSE of the decomposition prediction is respectively reduced 9.3%, 15.4% and 15.4% as to one-step, two-step and three-step. The average value is also decreased 13.4%. Fig.5 shows that decomposition method obviously shorted the prediction delay during traffic burst.

6 Conclusions

A prediction method based on decomposition is proposed to solve the problem of real-time traffic burst prediction. The network traffic is filtered and decomposed into two parts, the low frequency traffic and the high frequency part that changed fast, each of which is processed respectively by using different prediction methods and lastly the prediction results is superposed. In this paper, the authors bring forward EaLMS – an improved LMS prediction method and apply it to prediction of the low frequency traffic. A simple method of linear combination is applied to the high frequency traffic. The result of experiments show that, compared with directly LMS prediction, the prediction method based on decomposition obviously shorted the prediction delay and reduced the prediction error during traffic burst, while it also improves the global prediction.

Except smoothing filter, other decomposing methods could have a try in the future. Also, for the prediction of high frequency traffic, a simple linear combination method is used here, but we will attempt to develop other means according to different needs.

References

1. N. K. Groschitz, G. C. Polyzos, A time series model of long-term NSFNET backbone traffic, In Proceedings of the IEEE International Conference on Communications (ICC'94) (May 1994), vol. 3, pp. 1400--1404.
2. .2. E. S. Yu, C. Y. R. Chen, Traffic prediction using neural networks, In Proc. IEEE Globecom '93, pp.991-995, 1993.
3. A. A. Tarraf, I. W. Habib, T. N. Saadawi, ATM multimedia traffic prediction using neural networks, In Proceedings of Global Data Networking, 1993, pp. 77-84
4. Y. Liang, E. W. Page, Multiresolution Learning Paradigm and Signal Prediction, IEEE Transactions on Signal Processing, Vol. 45, Issue 11, Nov 1997, pp. 2858-2864
5. A. Adas, Using Adaptive Linear combination prediction to Support Real-Time VBR Video Under RCBR Network Service Model, IEEE/ACM Transaction on Networking, Vol.6, No. 5, Oct. 1998, pp. 635-644.
6. S. Chong, S. Li, and J. Ghosh, Predictive Dynamic Bandwidth Allocation for Efficient Transport of Real-Time VBR Video over ATM, IEEE JSAC, Vol. 13, No. 1, Jan. 1995, pp. 12-23.

7. A.Adas, Supporting Real Time VBR Video Using Dynamic Reservation Based on Linear combination prediction, IEEE Trans. Signal Processing, Vol.44, No.5, pp. 1156-1167, May 1996
8. A. Sang, S. Li. A predictability analysis of network traffic. In Proc. IEEE INFOCOM 2000. pp. 342-351
9. XU Ke, XU Jinwu, BAN Xiaojuan, Forecasting of Some Non-Stationary Time Series Based on Wavelet Decomposition, ACTA ELECTRONICA SINICA, Vol.29, pp. 566-568, 2001
10. YANG Xinyu, ZENG Ming, ZHAO Rui, SHI Yi, A Novel LMS Method for Real-time Network Traffic Prediction, Lecture Notes in Computer Science, Springer-Verlag Heidelberg, ISSN: 0302-9743, Volume 3046 / 2004, April 2004, Pages: 127 – 136, Computational Science and Its Applications - ICCSA 2004: International Conference, Assisi, Italy, May 14-17, 2004, Proceedings, Part IV ,ISBN: 3-540-22060-7
11. The Internet Traffic Archive. http://ita.ee.lbl.gov/
12. YANG Weiqin, GU Lan, Time Series Analyzing and Dynamic Data Modeling, BEIJING INSTITUTE OF TECHNOLOGY PRESS,1988.
13. HE Zhenya, Adaptive Signal Processing, SCIENCE PRESS (CHINA). 2002
14. R. Wong, E. Johston. A variable step size LMS algorithm. IEEE Trans. on Signal Processing. Vol. 40, No.7, 1992

An EJB-Based Platform for Policy-Based QoS Management of DiffServ Enabled Next Generation Networks*

Si-Ho Cha[1], WoongChul Choi[2,†], and Kuk-Hyun Cho[2]

[1] Dept. of Computer Engineering, Sejong University, Korea
sihoc@sejong.ac.kr
[2] Dept. of Computer Science, Kwangwoon University, Korea
wchoi@daisy.kw.ac.kr, khcho@cs.kw.ac.kr

Abstract. Unlike IntServ where resource reservation and admission control is per-flow based using RSVP, DiffServ supports aggregated traffic classes to provide various QoS to different classes of traffics. However, it is possible to lead to serious QoS violations without a QoS management support. Therefore, a QoS management system that can manage differentiated QoS provisioning is required. This paper proposes and implements a policy-based QoS management platform for differentiated services networks, which specifies QoS policies to guarantee dynamic QoS requirements. High-level QoS policies are represented as valid XML documents and are mapped into EJB beans of the EJB-based policy server of the platform. The policy distribution and the QoS monitoring are processed using SNMP.

1 Introduction

The best-effort service model in current IP networks does not provide the QoS requirements of next generation network services. To solve this problem, the IETF (Internet Engineering Task Force) proposed two models of Integrated Services (IntServ) and Differentiated Services (DiffServ) [1]. IntServ model is based on per-flow resource reservation and admission control through RSVP (Resource Reservation Protocol). The main disadvantage of IntServ is that the required information of flow states and the QoS treatments in a core IP network raise severe scalability problems. DiffServ model, on the other hand, supports aggregated traffic classes rather than individual flows and provides different QoS to different classes of packets in IP networks. However, current DiffServ specifications do not have a complete QoS management framework. It is possible to lead to serious QoS violations without a QoS management support. From this reasoning, a QoS management system that can manage differentiated QoS provisioning is required. Recently, policy-based management (PBM) has been considered as a technology that can provide QoS management sup-

* The present research has been conducted by the Research Grant of Kwangwoon University in 2004.
† Corresponding author.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 794–801, 2005.

ports [2]. The objective of the PBM is to manage the behavior of a network through the business rules that are high-level policies that describe the behavior of the network in a way as independently as possible of network devices and topology. The amount of QoS management task can be reduced by using policies because one policy can be used for many policy targets that are various network nodes and the policy can accept customer's dynamic QoS requirements.

From these backgrounds, we propose and implement a policy-based QoS management platform for DiffServ networks, called PMQoSDS [3]. The implementation of the proposed PMQoSDS platform is build on EJB (Enterprise JavaBeans) framework and uses XML (Extensible Markup Language) to represent and validate high-level QoS policies. There are several advantages of using EJB framework for the design and implementation of the PMQoSDS platform. EJB framework can reduce development cost, time to market, and can improve maintainability, extensibility, and functional design for the PMQoSDS [4]. There are also several advantages of using XML in representing high-level QoS policies. Because XML offers many useful parsers and validators, the efforts needed for developing a QoS management system can be reduced. One note is that the standard protocol for policy distribution of the IETF PBM architecture uses COPS (Common Open Policy Service). While most DiffServ routers support SNMP only, few routers support COPS, therefore, current implementation of the PMQoSDS uses SNMP to distribute QoS policies.

This paper is structured as follows. Section 2 discusses the architecture and components of the PMQoSDS. Section 3 presents the implementation of the PMQoSDS and the experimental results in the video streaming system. Finally in section 4 we conclude the paper.

2 PMQoSDS

2.1 Functional Architecture

The conceptual QoS management architecture of PMQoSDS is shown in Fig. 1. To process a policy-based QoS management efficiently and correctly, the following procedures are required:

1. Topology discovery: In order to describe the QoS of a DiffServ network, the PMQoSDS should have the knowledge of the routing topology and each router's role. The topology manager (TM) accomplishes the discovery of the routing topology and router type discovery by using two SNMP MIB-II tables, ipAddrTable and ipRouteTable. The topology and router information discovered from the network are stored in the topology DB, and are represented as topology node (TN).
2. Policy definition and validation: The PMQoSDS defines high-level QoS policies (HQPs) as valid XML documents. A manager creates valid XML documents and then validates them. Once the HQPs are validated, the PMQoSDS requests a policy manager (PM) to create the instances of low-level QoS policis (LQPs): packet classification policy (PCP), traffic conditioning policy (TCP), and queuing and scheduling policy (QSP).

3. Policy translation: The translation of a QoS policy from HQPs to LQPs is done by the PM. The PM translates HQPs to LQPs by properly setting the attributes of the three LQP. The attributes of the LQP are mapped into the device configuration parameters to configure the DiffServ routers for provisioning QoS requirements.
4. Policy deployment: The deployment of LQPs is done by the three LQP. These three LQPs perform SNMP operations for deploying each LQP. The PCP and the TCP are deployed to edge routers to control the functions of the edge routers, whereas the QSP is deployed to core routers to control the functions of the core routers. The PCP classifies packet flows and the TCP performs the traffic conditioning such as metering, marking, dropping, and/or shaping packets. The QSP performs queuing, scheduling, and/or dropping packets. A set of these actions is accomplished by using the DiffServ MIB.
5. QoS monitoring: A deployed QoS policy might not behave as defined in the policy. The QoS monitoring in the PMQOSDS uses the DiffServ MIB as in the policy deployment. The QoS manager (QM) accesses the policy definition in the three LQP and compares the observed behavior of a network to the one defined in the policy. If any QoS degradation is observed, the QM notifies an administrator by alerting messages and updates the performance database.

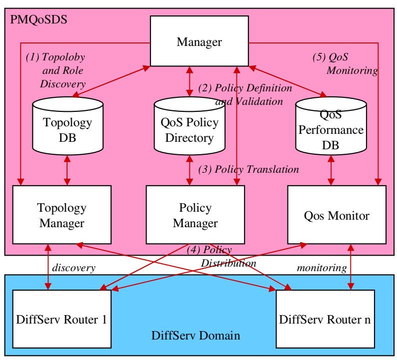


Fig. 1. Conceptual QoS management architecture of PMQoSDS

2.2 Architecture and Components

The implementation architecture of PMQoSDS is shown in Fig. 2. The PMQoSDS conforms to the Model-View-Controller (MVC) architecture. Therefore, it is highly manageable and scalable, and provides the overall strategy for the clear distribution of objects involved in managing service. There are two main components in the architecture: a Web server and an EJB-based policy server. A Web server is responsible for the presentation logic of the PMQoSDS. An EJB-based policy server is responsible for the business logic of the PMQoSDS.

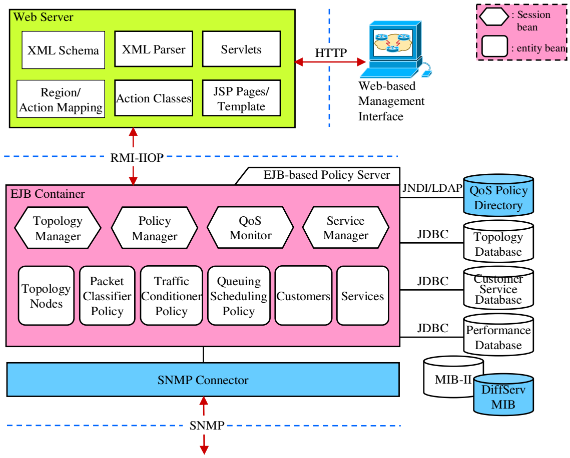


Fig. 2. Implemental QoS management architecture of PMQoSDS

As illustrated in Fig. 2, there are several functional components in the EJB-based policy server in the PMQoSDS. The PMQoSDS uses the following components to discover network topology and each router type.

– The TN bean is an entity bean containing the information of a network topology and each router type. The information is retrieved using SNMP MIB-II.
– The topology DB (TD) stores the topology information and each router type retrieved from a DiffServ network through SNMP.
– The TM bean is a session bean responsible for discovering the topology information and each router type and storing them into the TD and setting up the TN beans according to the retrieved information.

The PMQoSDS uses the following components to translate HQPs into LQPs and distribute the LQPs to the DiffServ network.

– The PCP bean is a part of LQP entity bean that classifies packet flows and assign class identifiers to them. The PCP beans are deployed to edge routers and control the inbound traffics.
– The TCP bean is a part of LQP entity bean that meters the classified packets to check whether they conform to a traffic profile and performs marking, dropping, and/or shaping packets according to the metering results. The TCP beans are deployed to edge routers and control the outbound traffics.
– The QSP bean is a part of LQP entity bean. It performs queuing, scheduling, and/or dropping packets. The QSP beans are deployed to core routers to control the outbound traffics.
– The QoS policy directory is a directory storing the LQP beans.
– The PM bean is a session bean that is responsible for translating HQPs into low-level QoS policy beans and setting the values of DiffServ MIBs of the routers. The

PM is also responsible for deploying the LQPs to relevant routers in the DiffServ network.

- The QM bean is a session bean that is responsible for monitoring the QoS resulted from a policy deployment by retrieving the values of DiffServ MIBs and comparing them to the attribute values of the three LQP beans.

2.3 QoS Policy

A QoS management policy can be represented as two views: HQP and LQP. The HQP is corresponding to a business level SLA and the LQP is corresponding to an individual device configuration. The HQPs can be populated by an administrator, while the LQPs are generated by a logic component. A HQP consists of a source/source group, a destination/destination group, a router/router group, an application/application group, a time/time group, and a service level. A service level is set to one of the class of service such as premium, gold, silver, and bronze service level. The premium service is provided using an EF PHB, whereas the gold and the silver service are provisioned to AF PHB groups of a DiffServ network. The bronze service is offered using the best-effort service of a network. In the PMQoSDS, the LQPs are specified in one of the three LQP beans such as PCP bean, TCP bean, and QSP bean. The translation of a QoS policy from HQPs to LQPs is accomplished by the PM session bean. A servlet receives data from an administrator and creates XML policy documents, and then validates them by an XML Schema. A servlet makes a request of an instance of PM session bean to create instances of the three LQP entity beans.

3 Implementation and Experiment

3.1 DiffServ Network Testbed

Linux-based routers are used for our DiffServ network testbed. Supporting differentiated services are already incorporated in the mainstream Linux kernel source code version 2.4 and later [5]. We use SNMP agent implementations for managing DiffServ routers [6]. Our SNMP agent for DiffServ MIB has been implemented by using UCD-SNMP package. We have used UCD-SNMP 4.2.2 and the DiffServ MIB has been inserted as an extension MIB in UCD-SNMP agent. For each SNMP operation on objects in DiffServ MIB, the UCD-SNMP agent invokes functions in the DiffServ agent. And then the DiffServ agent opens a netlink socket to the kernel and requests proper parameters for the SNMP operation.

3.2 PMQoSDS Platform

The PMQoSDS is implemented on a Windows 2000 server system. It consists of a Web server and an EJB-based policy server. The Web server is responsible for the presentation logic of the PMQoSDS and the EJB-based policy server is responsible for the business logic of the PMQoSDS. In the MVC architecture, the view is implemented using the JSP template mechanism and the Composite View pattern. The controller is implemented using the Front Controller and the Session Facade pattern. The model is implemented using EJB Entity Beans and Service Locator pattern.

We use Apache Tomcat 4.0.1 for the Servlet and JSP container. An EJB-based policy server within the business-tier runs an EJB server to manage EJB components. We use JBoss 2.4.10 for an EJB-based policy server and use EJB 1.1 to implement EJB beans. AdventNet SNMP APIs written in Java are used for handling SNMP operations. The Oracle 8i Enterprise Edition 8.1.6 for Windows NT is used for storing the performance and topology information derived from MIB tables.

Fig. 3 shows the input forms for the high-level policy information and shows the result of the request for the policy creation.

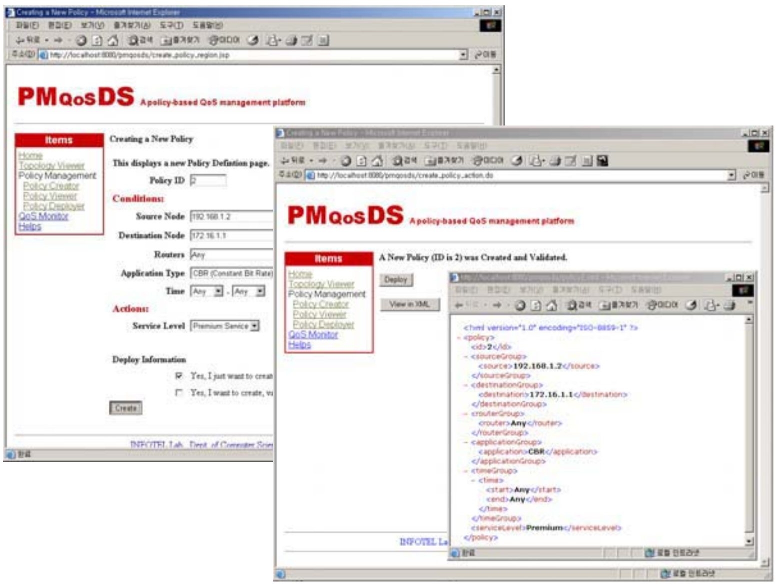


Fig. 3. Snapshot of the PMQoSDS

3.3 Experiments

To show the effectiveness of the PMQoSDS, we apply H.263-based video streaming systems [7] to our DiffServ network. To do that, we configure a testbed shown in Fig. 4, with several differences in the role of nodes. Two VOD servers are attached to the network - one server to D1 and the other to D4. They have exactly the same hardware and software system configurations. A policy server is attached to D3. The systems in the testbed are running on the following hardware configurations. The core routers are running on Pentium IV 1.8GHz with 512MB main memory, the edge routers on Pentium IV 1.5GHz with 512MB main memory, two VOD servers on Pentium IV 2.0GHz with 512MB main memory, and the other systems on Pentium III 1.0GHz with 256MB main memory. All the links in Fig. 5 are connected via FastEthernet NIC cards.

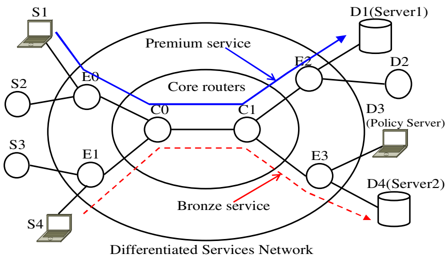


Fig. 4. Experiment Environment

Fig. 5. Snapshots of H.263 streaming system

In the configuration, there are three connections running - two for multimedia connections and the other one for cross traffic. Two connections for multimedia traffics are the connection between S1 and Server1, and the one between S4 and Server2. Those connections share a link between C0 and C1. To differentiate the services between them, the connection between S1 and Server1 is applied by Premium service, while the other multimedia connection between S4 and Server2 is applied by Bronze service. To make the sharing link congested, MGEN toolset is used to generate cross traffics on that link, and CBR traffics are used to do so. Cross traffics are generated at C0 and sinked at C1 router. By doing this, the service levels and the resulted QoS can be explicitly demonstrated.

Fig. 5 shows the snapshots of two H.263 streaming servers and two clients. Fig. 5 (a) and (b) show the snapshots at the H.263 streaming Server1 and S1 with a connection from Server1. Fig. 5 (c) and (d) show those at the H.263 streaming Server2 and S4 with a connection from Server2. As shown in Fig (b) and (d), the client S1 with Premium service level receives a video stream with bitrate (501 kbps) and video quality (29 fps), while the client S4 with Bronze service level receives a video stream with bitrate (136 kbps) and video quality (8 fps). From the experiment, we can verify that the PMQoSDS provides differentiated QoS levels to the contending connections using the management platform. Obviously, this work can be extended to a network with more complicated connections.

4 Conclusion

In this paper, we proposed and implemented a policy-based QoS management platform for DiffServ enabled IP networks, called PMQoSDS. The PMQoSDS integrated the functions of policy management and QoS monitoring by extending the original IETF PBM architecture to the policy-based QoS management. We showed the QoS management procedures as well as the structures and components of the PMQoSDS.

To show the effectiveness of our PMQoSDS, we experimented with video streaming systems in our Linux-based DiffServ testbed. In the experiment, we demonstrated that our PMQoSDS is able to manage differentiated QoS provisioning in a DiffServ network. Because this work can be obviously extended to a network with more complicated connections, we are currently extending our PMQoSDS with the various QoS policies on the testbed to study how the PMQoSDS can provide differentiated QoS guarantees with various service requirements. We expect our PMQoSDS to be successfully integrated in the service management systems used by the service providers in order to meet various dynamic QoS requirements from their customers.

References

1. S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, W. Weiss: Architecture for Differentiated Services, IETF RFC 2475, December 1998.
2. L. Lymberopoulos, E. Lupu, M. Sloman: An Adaptive Policy Based Management Framework for Differentiated Services Networks, IEEE Third International Workshop on Policies for Distributed Systems and Networks, Monterey, California June 5-7 2002.
3. S. Cha, J. Lee, D. Lee, K. Cho: A Policy-Based QoS Management Framework for Differentiated Services Networks, Springer-Verlag's Lecture Notes in Computer Science (LNCS) Vol. 2662, August 2003
4. Sun Microsystems, Inc.: Telecom Network Management With Enterprise JavaBeans™ Technology, Technical White Paper, May 2001.
5. Differentiated Services on Linux, http://diffserv.sourceforge.net.
6. Postech DiffServ MiB Implementation, http://dpnm.postech.ac.kr/research/01/ipqos/dsmib/
7. H. Cha, B. Ahn, K. Cho: A QoS-providing multicast network management system, Computer Communications 24, 2001

Determining Differentiated Services Network Pricing Through Auctions

Weilai Yang, Henry L. Owen, and Douglas M. Blough

School of Electrical and Computer Engineering,
Georgia Institute of Technology, Atlanta, GA 30332-0250 USA
owen@ece.gatech.edu
Phone: 404 894 4126

Abstract. Over the years, quality of service (QoS) has attracted attention from researchers. How to guarantee QoS to customers despite the rapid change of the network status is the main concern. Pricing provides an effective economic means of congestion control and revenue generation. We examine pricing as an effective strategy for revenue management in DiffServ networks. In this paper, we propose an auction scheme to allocate network resources efficiently so as to maximize service provider's revenue. We examine an auction mechanism that provides multiple options for customers to bid on the resources that they require as well as the price they are willing to pay. The service provider acts as an admission control unit in the sense of deciding the admission price and service provided for each class. We target the goal of maximizing a service provider's revenue through auctions.

1 Introduction

Traditionally only best effort service is employed in Internet. Under this system, all clients pay the same amount of money to get the same kind of service. When the network is congested, the service provider randomly drops packets. There is no guarantee on any specific services for the customers. As the Internet moves from only providing best effort service to a differentiated service network, a new design of pricing and resource allocation strategy is desired. Pricing has been shown to be an effective and efficient way for service improvement and revenue generation.

There are several pricing approaches, e.g. a cost based scheme, an optimization based methodology, edge pricing, auctions and so on [4, 6, 7, 8]. Despite the variety of the strategies, the basic idea is that the appropriate pricing policy will provide incentives for users to behave in ways that improve overall utilization and performance of the network. An auction is a mechanism that allows for the submission of bids that guide, rather than explicitly specify the choice of service and price to fulfill the buyer's needs. Auction is a decentralized mechanism for efficiently and fairly sharing resources inside a network [3].

We study the revenue maximization problem of a price-based resource allocation scheme for Differential Service (DiffServ) data networks [2]. Best Effort

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 802–809, 2005.

(BE) is the default per hop behaviour for best effort traffic and some minimum amount of bandwidth will always exist for BE. How to allocate the remaining bandwidth for Expedited Forwarding (EF) and Assured Forwarding (AF) is an open question. Therefore, in our model, we deal with a two-class EF and AF ratio resource allocation problem.

In this paper, we consider a scenario where customers bid and specify price and service required. There are two parts in the price bids. One is called base price, which corresponds to the minimum bandwidth requirement. The other is price sensitivity coefficient, which measures the payment for any extra resource allocation other than the minimum bandwidth requirement. The auctioneer tries to maximize the service provider's revenue by selectively accepting bids. Auctions happen at fixed time intervals. The service provider calculates a minimum bandwidth they would provide to each class based on all the bids, with the goal of maximizing the service provider's profit.

2 Related Work

"Smart Market" [5] opens the door of using an auction mechanism to solve the Internet pricing problem. The main idea of "smart market" is to find a way to deal with modeling the pricing to manage congestion, encourage network growth and guide resources to their most valuable use. The threshold price (which is calculated as a marginal cost when the network gets congested) reflects the resource costs and offers users incentives to pay more for a valuable service or release the resources to others. Even though "smart market" creates the ability to allocate Internet resources in an economic context, practically it is very hard implement in a real network. To offer a bid on each traffic packet yields too much overhead in the network and bursts are difficult to handle. Also, how fast the users react to the auction results could fluctuate the price rapidly and irregularly.

Basar and Srikant [1] assumed that the price of a per-bandwidth unit is fixed to users and the transmission rate of each user is a function of network congestion and price-per-unit bandwidth. They verified that as the number of users increases, the optimal price-per-unit-bandwidth increases.The utility function that they adopt is U=$w_i$logx_i (w_i is a sensitivity coefficient and x_i is the bandwidth). They use w_i's value as an admission criteria. Users with smaller w_i's are dropped out of the network. We consider this a valid strategy to keep admission simple but effective.

3 Problem Formulation

To make a bid in an auction, a customer needs to specify three values. First of all, customers are required to bid the minimum service that they demand (the bandwidth in our case) and the corresponding price they would like to pay. This price is called base price, to support the basic service. Besides this, if customers also need more bandwidth than the minimum requirement, they need to pay for this extra part also. This happens when customers can tolerate the minimum resource allocation, but prefer even more if possible. For example, when a video conference application is being transmitted, there is a minimum resource requirement to support it. If extra bandwidth is available, customers may be able to use it for better quality, thus they need to specify their valuation for extra bandwidth. For the sake of fairness, we assume the base price and minimum resource allocation dominate. In order to prevent the link capacity from being eaten up by those extra bandwidth requests, a logarithm function is employed here. It is described as: $W_j \log \frac{X_j}{L_j}$, where X_j is the bandwidth allocation to customer j and L_j is customer j's minimum bandwidth requirement. When $X_j = L_j$, there is no extra cost other than the minimum. If $X_j > L_j$, the amount of charge depends on value W_j, which comes from customers' bids. We call W_j the price sensitivity coefficient. Customers can bid W_j=0, which indicates that they do not want any bandwidth beyond the minimum. The general revenue function is:

$$U_{kj} = U_{0j} + W_j \log \frac{X_{kj}}{L_{kj}}$$

where U_{kj} is the revenue from client k in class j. U_{0j} is class j's base price. W_j is the sensitivity for class j, which stands for customers' willingness of paying for more bandwidth than the minimum requirement. X_{kj} is the bandwidth assigned to client k, class j. L_{kj} is the minimum bandwidth required by client k.

Customers bid for the base price, sensitivity coefficient and minimum required bandwidth. The objective is to maximize the service provider's revenue, subject to the system's available resources.
The mathematical formulation is as follows:

Decision variables:

$$Z_{ij} = \begin{cases} 1; \text{if client i is admitted to class j} \\ 0; \text{otherwise} \end{cases}$$

U_{0ij}: base price from client i in class j;
X_{ij} : bandwidth obtained by client i in class j;
L_{mj} : minimum bandwidth for class j;
W_j : price sensitivity for class j;
X_j : bandwidth assigned to each individual client in class j;
Objective function:

$$max \sum_{j=1}^{2} \sum_{i} (U_{0ij} + W_j \log \frac{X_{ij}}{L_{mj}}) * Z_{ij} \qquad (1)$$

Subject to:

$$\begin{cases} \sum_{j=1}^{2} \sum_i X_{ij} <= Q \\ X_{ij} \geq L_{mj} - (1 - Z_{ij}) * M \\ W_j \leq W_{ij} + (1 - Z_{ij}) * M \\ X_{ij} \geq V_i - (1 - Z_{ij}) * M \\ X_{ij} \geq X_j - (1 - Z_{ij}) * M \\ X_{ij} \leq 0 + Z_{ij} * M \\ X_{ij} \geq 0; L_{mj} \geq 0; W_j \geq 0 \\ X_{ij} \leq X_j \end{cases}$$

Parameters:
Q : total bandwidth
V_i : minimum bandwidth required by client i
M : a very large positive number

The scenario is that all the customers propose values: U_{0ij}, W_{ij} and L_{ij} and we have to decide which flows should be admitted for each class with the objective of maximizing the service provider's revenue. For each class j, we adopt the minimum U_{0ij}, W_{ij} as our U_{0j}, W_j and the maximum L_{ij} as our L_j. All flows in one class are assigned the same amount of bandwidth.

4 Optimal Solution

We notice that every flow in one class has the same threshold (U_{0j}, W_j, L_j) and flows within the same class will obtain the same bandwidth. Suppose that class j's assigned bandwidth is Q_j, each flow gets its own share of Q_j/m_j where m_j represents the number of flows admitted into class j, and generates the same revenue. We solve the problem using the objective function and corresponding constraints formulated as follows:

$$\max \textstyle\sum_{j\in N} \; m_j * (U_{0j} + W_j \log \frac{Q_j}{m_j L_j}) \tag{2}$$

$$\text{subject to:} \qquad \textstyle\sum_{j\in N} Q_j = Q. \tag{3}$$

The solutions are obtained by Lagrange relaxation:

$$Q_j = (m_j W_j)/(\sum_{j\in N} m_j W_j) * Q, \ \forall j \in N. \tag{4}$$

Therefore, given (m_j, U_{0j}, W_j and L_j), we can obtain Q_j as in equation (4). According to auction policy, $U_{0j} = \min\{U_{0ij}, i \in M_j\}$, $W_j = \min\{W_{ij}, i \in M_j\}$ and $L_j = \max\{L_{ij}, i \in M_j\}$. This also implies that each combination (U_{0kj}, W_{mj} and L_{nj}), where $k, m, n \in M_j$, provides a candidate value set for class j. Therefore, based on the bids in class j, we make all combinations in terms of U_{0ij}, W_{ij} and L_{ij}. For each combination, which corresponds to one predetermined set of value (U^*_{0j}, L^*_j and W^*_j) for class j, we sort out all the flows such that $U_{0ij} \geq U^*_{0j}$, $W_{ij} \geq W^*_j$ and $L_{ij} \leq L^*_j \ \ \forall i \in M_j$. Record the number as k. So each combination has a k value associated with.

Now, for each class $j \in N$, starting from m_j=1 and L_{1j}, check all the combinations of (U_{0ij}, W_{ij}). From these, the effective ones are those with $k \geq m_j$. From our previous assumption that U_{0ij} is a dominant pricing factor, which is given the highest priority to choose the combinations with largest U values. From among the largest U value set clients, choose the one with the highest W value. Until now, we obtained values of U_{0j}, W_j and L_j for each class j. Then we have all inputs ($U_{0j}, m_j, L_j, W_j \ \forall j \in N$) for the solution of equation 2, and $Q_j \ \forall j \in N$ can be calculated as in equation 4. We have specified earlier that each admitted flow in one class shares the same amount of bandwidth and this bandwidth has to be greater than or equal to their bids. We are using that property to check the validity of each possible solution. If and only if $Q_j/m_j \geq L_j \; \forall j \in N$, the solution is a qualified candidate. If so, by using those values as well as U_{0j}, the total revenue is computed. Otherwise, this set of solution values is abandoned. Following the same steps by changing the value of m_j and L_j, we can obtain all the possible feasible solutions. Finally, the solution with the highest total revenue is optimal.

So now we have the optimal solutions for calculating the best thresholds as well as the assigned bandwidth to each class and client, in terms of maximizing service provider's revenue. The next question is how should we use the thresholds to admit new flows. We know that the auction occurs with a fixed time interval. During that interval, when new customers want to join in, they present their bids. Then, if it is possible to admit them, they can get into the network. Otherwise, they have to wait for the next auction to take place. How does the service provider decide if letting them in is going to benefit him or not? We propose the following property to explain what procedure the service provider should follow in order to make a good judgment.

Property 1. If Q_j and (U_{0j}, W_j, L_j) are fixed, as long as $Q_j/m_j > L_j$, U_j is a strictly increasing function of m_j.

Proof:

The revenue function is:

$$U_j = m_j * U_{0j} + m_j * W_j \log \frac{Q_j}{m_j L_j}. \tag{5}$$

Its derivative is:

$$\frac{\partial U_j}{\partial m_j} = U_{0j} + W_j \log \frac{Q_j}{m_j L_j} - W_j. \tag{6}$$

Since $Q_j/m_j > L_j$, and U_{0j} is greater than W_j (which is our assumption), $\frac{\partial U_j}{\partial m_j}$ is always greater than 0. That guarantees that U_j is a strictly increasing function of m_j.

Using property 1, the service provider can increase the revenue by admitting any new flow i into class j as long as $L_{ij} < L_j$, $W_{ij} > W_j$ and $U_{0ij} > U_{0j}$. So, after the bidding thresholds have been decided, property 1 gives the service provider a guideline as to how to admit new flows.

5 Simulation and Analysis

As an example, we may assume that the Best Effort (BE) class is charged $35 per client per month. Divide that by days and minutes, 0.00135 cents per minute needs to be paid for BE traffic. Let a mcent (or a unit) be equal to 1/1000 cents, so the charge for BE is 1.35 mcents. Based on this, we define EF traffic's price as twice as much as BE's and AF's price as 1.5 times as much as BE's. These values

Table 1. The assumptions that are used in the simulations

	Parameters		
	FV [1]	MV [2]	SD [3]
EF	2.7 *mcents*	[2.7 - 5.4]*mcents*	[1 - 2]*mcents*
AF	2.0 *mcents*	[2.0 - 4.0]*mcents*	[1 - 2]*mcents*

are the service provider's price thresholds for each class. Any customer who bids lower than the threshold price will be rejected. The customers' valuations for EF and AF are assumed to be normally distributed.

In the flat rate scenario, a customer is admitted if and only if his valuation (which is same as the bid in the auction context) is greater or equal to the fixed price set by the service provider. The revenue is the number of customers multiplied by the price.

All the parameters given in Table 1 are used to determine the revenue generated by service provider. The customers' mean valuation (MV) and standard deviation (SD) are within a fixed range instead of a fixed number. This is because we vary the MV and SD in the simulations to show that our algorithm is not sensitive to how those parameters are chosen.

To compare the optimal resource allocation results with traditional flat rate pricing, we set the flat rate as an independent variable. We ran our algorithm with a fixed set of parameters and compared the results with the revenue generated by a fixed rate pricing scheme. The rate for each class changes within the range of (20%–220%)$\times$ MV where MV is the customers' mean valuation. We want to test and show that the algorithm's performance is not sensitive to how we choose the parameters. In other words, we want to show the auction scheme that we propose is robust. We vary the offered network load from 70%, 100% to 140% and vary the customers' mean valuations. Figure 1 shows the revenue comparison between our algorithm and flat rate pricing when the customers' valuations are normally distributed with a mean of 2.7 mcents and a standard deviation of 1 mcent for EF and a mean of 2.0 mcents and standard deviation 1 mcents for AF. The subgraphs (a), (b) and (c) show the results with network load at 70%, 100% and 140% respectively. We then vary customers' mean valuation to 4.0 mcents for EF and 3.0 mcents for AF (standard deviation remains the same as before). Figure 2 shows how the network behaves when the mean valuation changes to 5.4 mcents for EF and 4.0 mcents for AF. From all the results, we can see that the auction scheme predominantly outperforms fixed pricing. In all cases, auctions generate more revenue than the fixed rates. The revenue generated by the fixed rate pricing has similar curves because when the fixed rate is very low, even when it wins all the customers, the sum of the payments is low and when the rate is high, it loses customers which also reduces the service

[1] FV: Floor Value
[2] MV: Customers' Mean Valuation
[3] SD: Customers' Standard Deviation

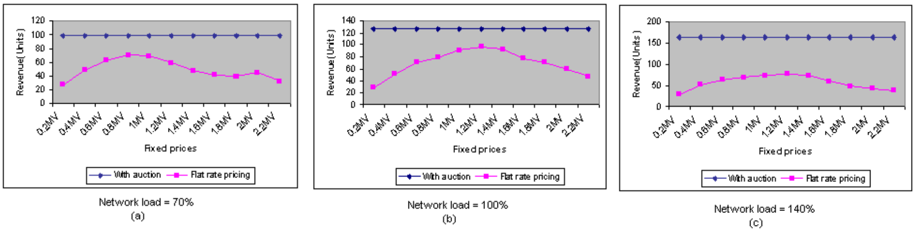


Fig. 1. The revenue comparison (customers mean valuation is 2.7 and 2.0 mcents)

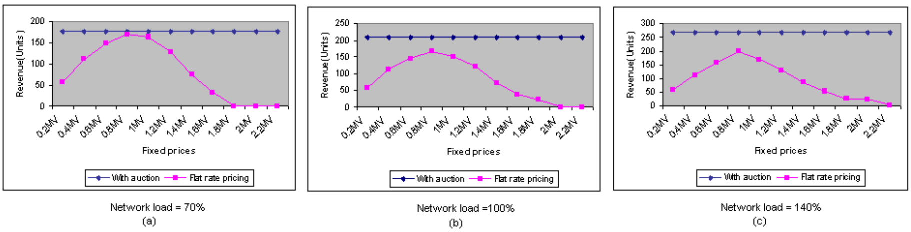


Fig. 2. The revenue comparison (customers mean valuation is 5.4 and 4.0 mcents)

provider's profit. In some cases, the curve fluctuates. This is because when the fixed rate increases, the number of admitted customers decreases. That causes the revenue function to be nonlinear. Also, it shows the same trend that as the network load increases, the gap between auctions and fixed price increases. This shows that auctions performance improves when the system gets congested. This is because auctions offer the service providers more options to choose the most valuable customers and drop others. It also causes customers to compete for bandwidth by raising their prices.

6 Conclusion

We considered a DiffServ network and studied the problem of maximizing the service provider's profit using pricing. We presented a novel pricing strategy of maximizing the service provider's revenue based on clients' bids of price as well as desired service. The scheme proposed in this paper gives customers the option to choose how much they want to pay for along with their required services. Our solution provides the thresholds for each service class according to network resource availability. The thresholds can also be used as a future reference for admitting new clients. We compared the revenue generated by auction and fixed pricing. Our auction scheme generates the best result even when varying the parameters. Our results show that the auction strategy beats the commonly used fixed rate pricing scheme.

References

1. T. Basar and R. Srikant, Revenue-Maximizing Pricing and Capacity Expansion in a Many-Users Regime, *IEEE Infocom*, 2002.
2. Y. Bernet, J. Binder, S. Blake, M. Carlson, B. E. Carpenter, S. Keshav, E. Davies, B. Ohlman, D. Verma, Z. Wang, And W. Weiss, A Framework for Differentiated Services. IETF Internet Draft, February 1999.
3. A. A. Lazar and N. Semret, Auctions for Network Resource Sharing CTR Technical Report CU/CTR/TR 468-97-02, Columbia University February 11, 1997.
4. P. Marbach, Pricing Priority Classes in a Differentiated Services Network. In *Proceeding of the 37th Annual Allerton Conference on Communications, Control, and Computing,* Monticello, IL, 1999.
5. J. F. MacKie-Mason and H. Varian, Pricing the Internet. In B. Kahin and J. Keller, editors, *Public Access to the Internet*, Prentice-Hall, Englewood Cliffs, NJ, 1994.
6. J. Mackie-Mason and H. Varian, Pricing Congestible Network Resources, *IEEE JSAC*, vol. 13, no. 7,pp 1141-48, Sept 1995.
7. I.Ch.Paschalidis and J.N.Tsitsiklis, Congestion-Dependent Pricing of Network Services. *IEEE/ACM Transactions on Networking*, 8(2): 171-184, April 2000.
8. P. Reichl, S. Leinen, and B. Stiller, A Practical Review of Pricing and Cost Recovery for Internet Services *Proc. of the 2nd Internet Economics Workshop Berlin (IEW'99)*, May 1999.

A Congestion Control Scheme for Supporting Differentiated Service in Mobile Ad Hoc Networks

Jin-Nyun Kim, Kyung-Jun Kim, and Ki-Jun Han*

Department of Computer Engineering, Kyungpook National University,
1370 Sankyuk-dong, Book-gu, Daegu, 702-701, Korea
{duritz, kjkim}@netopia.knu.ac.kr
kjhan@bh.knu.ac.kr

Abstract. There is a growing need to support quality-of-service (QoS) in mobile ad hoc networks. Supporting differentiated services (DiffServ) in mobile ad hoc networks, however, is very difficult because of the dynamic nature of mobile ad hoc networks, which causes network congestion. We propose DiffServ module to support differentiated service in mobile ad hoc networks through congestion control. Our DiffServ module uses the periodical rate control for real time traffic and also uses the best effort bandwidth concession when network congestion occurs. We evaluate our mechanism via a simulation study. Simulation results show our mechanism may offer a low and stable delay and a stable throughput for real time traffic in mobile ad hoc networks.

1 Introduction

Differentiated services (DiffServ) [1] has been widely accepted as the service model to adopt for providing quality-of-service (QoS) over the next-generation IP networks. DiffServ uses the concept of Per Hop Behaviors (PHBs), which provide different levels of QoS to aggregated flows. This is done by classifying individual traffic flows into various service levels desired before entering the DiffServ network domain. Within the DiffServ domain, flows of the same class are aggregated and treated as one flow. Each aggregated flow is given a different treatment, in terms of network resources assigned, as described by the PHB for that class.

There are three PHBs such as Expedited Forwarding (EF) [2], [3] service, Assured Forwarding (AF) service and Best Effort service. Service level agreements (SLAs) contain delay and throughput requirements among others like reliability requirements [4]. The EF PHBs provide low loss, low delay, and low jitter services for real time traffic that represents traffic like video or voice. We will use EF traffic as the same term with real time traffic within this paper.

* Correspondent author.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 810–817, 2005.

A mobile ad hoc network is formed by a group of wireless stations without infrastructure. There is a growing need to support real time traffic in mobile ad hoc networks. This, However, is very challenging because mobile ad hoc networks represent dynamic nature, which causes unexpected network congestion [6]. In this figure, we can see that network congestion is induced when a mobile station moves, which may consequently cause high delay and low throughput. Consequently, the QoS guarantee of real time flows is violated.

In this paper, we propose a DiffServ module to support differentiated service in mobile ad hoc networks through congestion control. Our DiffServ module uses the periodical rate control for real time traffic and the best effort bandwidth concession when network congestion occurs.

The organization of this paper is as follows. In Section 2, we describe our DiffServ module and congestion detection and congestion control mechanism. In Section 3, we represent simulation model, simulation parameters and some results. Finally, conclusions are offered in Section 4.

2 Proposed DiffServ Module for Mobile Ad Hoc Networks

The most dominant factor of packet transfer delay in networks is queuing delay. So, delay and jitter are minimized when queuing delays are minimized. The intent of the EF PHB is to provide a PHB in which EF marked packets usually encounter short or empty queues. EF Service should provide minimum delay and jitter [2].

According to RFC 3246 [2] which discusses the departure time of EF traffic, a node that supports EF on an interface I at some configured rate R must satisfy the following condition for the j-th packet:

$$d(j) \leq F(j) + E \tag{1}$$

where d(j) is an actual departure time, F(j) is a target departure time, and E is a tolerance that depends on the particular node characteristics. E provides an upper bound on (d(j)-F(j)) for all j.

F(j) is defined iteratively by

$$F(0) = 0, \qquad d(0) = 0 \qquad \text{for all } j > 0: \tag{2}$$

$$F(j) = max[a(j), min(d(j-1), F(j-1) + \frac{L(j)}{R}. \tag{3}$$

where a(j), L(j), and R denote an arrival time, the packet length, and the EF configured rate, respectively.

The rate at which EF traffic is served at a given output interface should be at least the arrival rate, independent of the offered load of non-EF traffic to that interface [2]. The relationships between EF traffic's input rate and output rate in each node are represented in the following three cases:

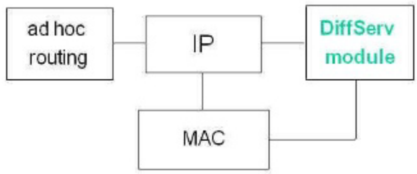


Fig. 1. DiffServ module

1. input rate > output rate
2. input rate < output rate
3. input rate = output rate

In case 1, it is difficult to support EF service because of the higher queuing delay. In case 2, the queuing delay is minimal so that high quality is provided to EF traffic. Non-EF traffic, however, is starved. Also, the delay and throughput of EF traffic can fluctuate because the output rate of EF traffic is disturbed. Finally, case 3 is considered as an ideal case for EF traffic. We assert that the input and output rate of EF traffic should be the same.

The proposed DiffServ module exists in the IP layer together with routing protocol as illustrated in Fig. 1. There is an interface between the DiffServ module and MAC for their interoperation. The DiffServ module controls the output rate of traffic using a MAC delay or bandwidth usage provided through the interface. The objective of our DiffServ module is guaranteeing the QoS requirement of already established real time traffic. The DiffServ module has two main roles:

1. It periodically regulates the output rate of real time traffic to meet bandwidth requirements. In other words, the output rate is maintained the same as the input rate. This rate maintenance provides not only the ideal EF service as previously described but also a stable throughput and delay of real time traffic. The rate adjustment can be implemented by using the token (leaky) bucket [9].
2. When congestion occurs, the bandwidth of best effort traffic is conceded to real time traffic in order to prevent the QoS requirement penalty. Fig. 2 illustrates the conceding of best effort bandwidth to real time traffic.

Fig. 2. The concession of best effort bandwidth

If queues remain short and empty relative to the buffer space available, packet loss and queuing delay is kept to a minimum. Since EF traffic usually encounters short or empty queues, and node mobility induces obscurity of queue utilization (i.e., the queue length of node after moving reflects the queue length of at position right before moving), the conventional congestion detection method (e.g., drop tail, RED (Random Early Detection)) using a queue overflow or queue threshold value is not appropriate for mobile ad hoc networks. For these reasons, in our scheme, congestions are detected by monitoring when the delay and bandwidth utilization of real time packets exceed a given threshold. Packet delay and bandwidth utilization can be simply measured at the congestion node by using the timestamp in a packet and counting amount of packet received per second, respectively. Also, the recent research, BLUE [8] shows that RED congestion avoidance algorithm using a queue length estimate to detect congestion has inherent weakness. Queue lengths do not directly relate to the true level of congestion in the real packet networks. BLUE use the packet loss and link utilization history for congestion detection.

When a node detects congestion, it sends out congestion notification messages in the direction of the source node of the real time flow as illustrated in Fig. 3. The notification messages are broadcast because of wireless medium characteristics. First, one-hop upstream nodes receiving the notification messages concede the bandwidth allocated for their best effort flows to their real time flows to relieve a congested situation. At this time, if the congestion is resolved, all congestion control processes end and congestion notification messages are no longer relayed in the direction of source nodes. Usually, the congestion can be solved at one-hop upstream nodes of congestion node because many one-hop upstream nodes (three nodes in Fig. 3) simultaneously reduce the rate of their best effort traffic. If the congestion is not relieved, however, the congestion notification messages are continuously relayed in the direction of source nodes. So, two-hop, three-hop, . . . , upstream nodes receiving the notification messages reduce their output rates of their best effort flows. Through this process, if the congestion is solved all processes successfully end, and if the congestion is not solved the notification messages are continuously relayed in direction of source nodes until

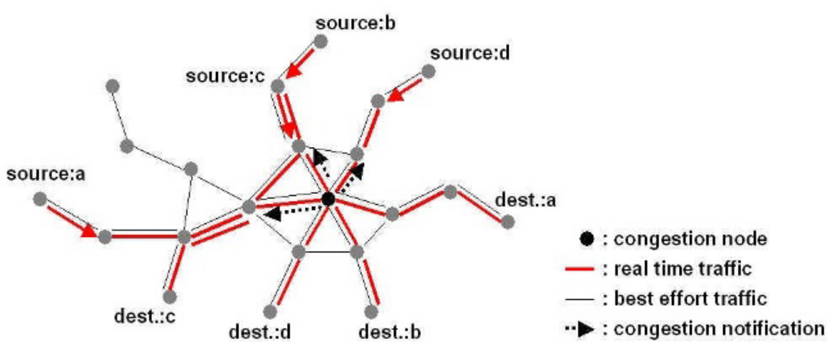


Fig. 3. Congestion control in mobile ad hoc networks

congestion is solved. So, light congestion is simply relieved at one-hop upstream nodes, but the heavy congestion is relieved after many upstream nodes reduce their best effort output rates.

3 Simulation

We evaluated our mechanism via simulation. Fig. 4 illustrates the network model used in the simulation. We have modeled only one congestion node and it's three upstream nodes from network of Fig. 3 because only three one-hop upstream nodes of congestion node can completely relieve the congestion in our simulation. Also, the rate reduction amount of the best effort bandwidth at each node can determine more relay of congestion notification message. This is network design choice.

In Fig. 4, three nodes simultaneously access a channel in order to communicate with a congested node. It is assumed that a contention-based service by the IEEE 802.11 Distributed Coordination Function (DCF) [5] channel access mode, based on the carrier sense multiple access with collision avoidance (CAMA/CA), is used to contend for the medium for each packet transmission. When a packet arrives at the MAC layer, the MAC listens to the channel and defers access to the channel according to CSMA/CA algorithm. When the MAC acquires access to the channel, then packets are exchanged.

We have simulated the 802.11 DCF as time slot based. The length of data packet is assumed as 80 bytes which is equivalent to 26 μs at the channel bit rate of 24 Mbps. The DCF and simulation parameters are reported in Table. 1. Each

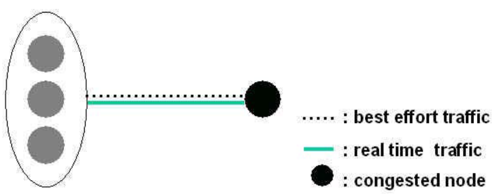
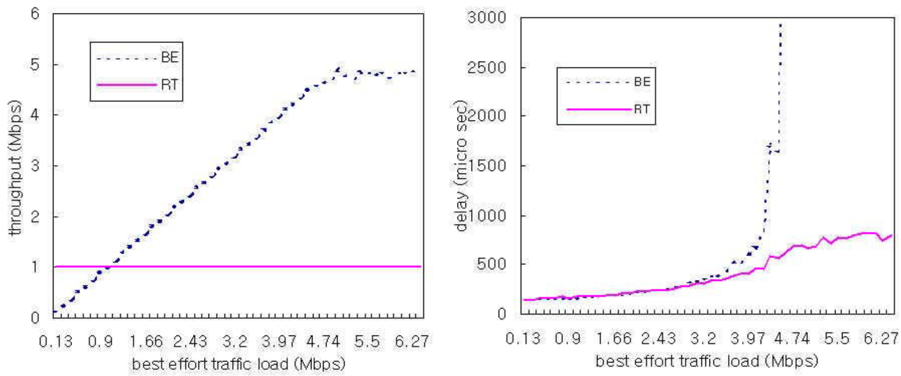


Fig. 4. Network model

Table 1. DCF and simulation parameters

Channel bit rate	24 Mbps
Slot time	9 μs
SIFS	16 μs
DIFS	34 μs
Length (size) of contention window	0~63 μs (8)
ACK transmission time	5 μs
Data packet transmission time	27 μs (80bytes)



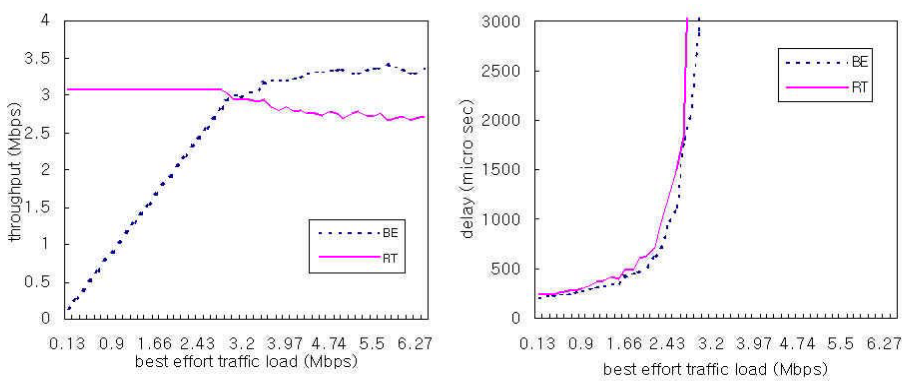
(a)Throughput (b)Delay

Fig. 5. The throughput and delay when the real time bandwidth requirement is 1 Mbps

node is modeled as a perfect output buffered device, that is, one which delivers packets immediately to the appropriate output queue.

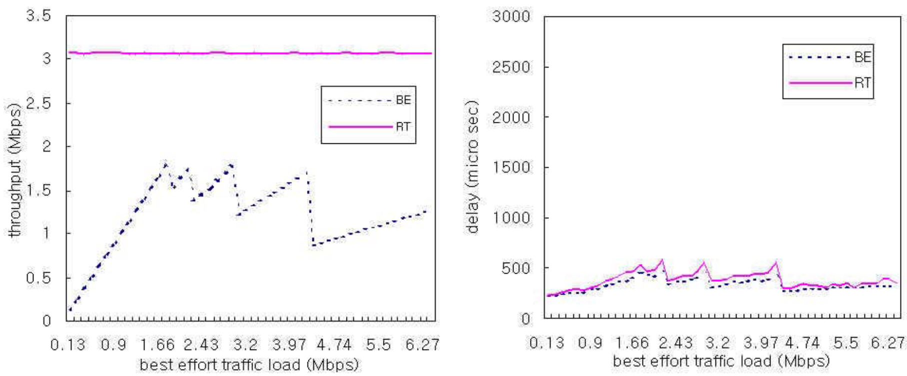
The focus of the experiments is whether our DiffServ module guarantees the QoS requirements of real time traffics as the best effort traffic load is increased.

Fig. 5 shows throughput and delay when the output rate of real time traffic is maintained the same as the input rate. The input rate of real time traffic is 1Mbps that represents a bandwidth requirement. In the experiment, the best effort traffic load continuously increased while the rate of real time traffic is maintained at 1 Mbps. As a result of experiment, the throughput of the best effort traffic increases up to some point and after that point, is saturated to almost 5 Mbps. Furthermore, the delay of the best effort traffic suddenly increases after the saturation point. We can see that the throughput of real time traffic is maintained



(a)Throughput (b)Delay

Fig. 6. The throughput and delay with no congestion control when the real time bandwidth requirement is 3.027 Mbps



(a)Throughput (b)Delay

Fig. 7. The throughput and delay with congestion control when the real time bandwidth requirement is 3.027 Mbps

successfully meeting the bandwidth requirement. The delay performance of real time traffic also shows a relatively stable pattern. Assuming that the bandwidth requirement is 1 Mbps and the node-to-node delay requirement is 10 ms, then the network is not congested. A congestion control mechanism for real time traffic is not needed.

If the bandwidth requirement of real time traffic is changed to 3.027Mbps, however, we can observe congestion as shown in Fig. 6. In this case, the bandwidth requirement of real time is satisfied when the offered loads of best effort are relatively low. As the best effort traffic load increases, however, the throughput of real time traffic decreases and the delay also terribly increases, which induces the QoS violation of real time traffic.

Fig. 7 shows the throughput and delay performances with our congestion control mechanism. When the bandwidth requirement of real time traffic is 3.027Mbps, the throughput of real time traffic is stably maintained at 3.027Mbps and the delay is also maintained at stably low values as a result of the best effort bandwidth concession. In Fig. 7, the crooked points represent that congestion control is performed. Whenever congestion is detected, the bandwidth of best effort traffic is conceded to real time traffic through its rate reduction. As previously described, congestion is detected by a delay threshold value of real time packet.

4 Conclusion

In this paper, we proposed DiffServ module, supporting service differentiation in mobile ad hoc networks through rate regulation and congestion control. In our scheme, for real time traffic, we regulated its output rate the same as the input rate. This regulation produced stable throughput and delays. The congestion was detected by measuring the delay or bandwidth utilization of real time traffic and comparing it with some threshold values. The congestion was controlled by conceding the best effort bandwidth to real time traffic. We verified our DiffServ mechanism through simulation. The experiment results showed that our mechanism could offer stable throughput and stably low delays for real time traffic.

Acknowledgement. University Fundamental Research Program supported by Ministry of Information and Communication in Republic of Korea.

References

1. S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss.: An architecture for differentiated services. IETF, RFC 2475, Dec (1998)
2. B. Davie, A. Charny, J.C.R. Bennett, K. Benson, J.Y. Le Boudec, W. Courtney, S. Davari, V. Firoiu, and D. Stiliadis.: An expedited forwarding PHB. IETF, RFC 3246, March (2002)
3. A. Charny, J.D.R. Bennett, K. Benson, J.Y. Le Boudec, A. Chiu, W. Courtney, S. Davari, V. Firoiu, C. Kalmanek, and K.K. Ramakrishnan.: Supplemental information for the new definition of the EF PHB. IETF, RFC 3247, March (2002)
4. T. C-K. Hui and C.-K. Tham.: Adaptive provisioning of differentiated services networks based on reinforcement learning. IEEE Trans. Syst. Man. Cybern. C, vol. 33, no. 4, pp. 492-501, Nov. (2003)
5. IEEE 802.11 Working Group.: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications. IEEE standard 802.11, June (1999)
6. Gahng-Seop Ahn, Andrew T. Campbell, Andras Veres, and Li-Hsiang Sun.: Supporting service differentiation for real-time and best-effort traffic in stateless wireless ad hoc networks. IEEE Trans. Mobile computing, vol. 1, July-Sept. (2002)
7. S. Chen and K. Nahrstedt.: Distributed quality-of-service routing in ad hoc networks. IEEE JSAC, vol. 17, no. 8, Aug. (1999)
8. Wu-chang Feng, Kang G. Shin, Dilip D. Kandlur and Debanjan Saha.: The BLUE Active Queue Management Algorithms. IEEE/ACM Trans. Networking, Aug. (2002)
9. Mohamed A. El-Gency, Abhijit Bose, Kang G. Shin.: Evolution of the internet QoS and support for soft real-time applications. Proc. IEEE, vol. 91, no. 7, Jury (2003)

Models and Analysis of TCC/AQM Schemes over DiffServ Networks

Jahwan Koo[1], Jitae Shin[1], Seongjin Ahn[2], and Jinwook Chung[1]

[1] School of Information and Communications Engineering,
Sungkyunkwan Univ. Chunchun-dong 300,
Jangan-gu, Suwon, Kyounggi-do, Korea
{jhkoo, jwchung}@songgang.skku.ac.kr
jtshin@ece.skku.ac.kr

[2] Department of Computer Education,
Sungkyunkwan Univ. Myeongnyun-dong 3-ga 53,
Jongno-gu, Seoul, Korea
sjahn@comedu.skku.ac.kr

Abstract. QoS-enabled networks consist of major functions: one is a TCP congestion control (TCC) mechanism at a source/sink node and the other is an active queue management (AQM) scheme at an intermediate node. In this paper, we introduce the major TCC mechanisms and AQM schemes, and provide analytical models and simulation-based comparisons of these TCC/AQM schemes for the purpose of identifying the reciprocal relationship between TCC mechanisms and AQM schemes in QoS-enabled networks. The results show that the equilibrium and dynamics of the underlying network depends on the harmony between the TCC/AQM pairs. The NewReno/PI pair, a feedback-based mechanism encompassing both network and end-systems, can enhance the performance of packet loss and delay sensitive applications. In our opinion, an appropriate combination of active queue management from the network and TCP source reaction would provide an effective solution to the instantaneous network fluctuation which occurs on the Internet.

1 Introduction

Millions of users have started to use wired and/or wireless networks and the amount of traffic has increased considerably. In addition, new peer-to-peer applications such as Napster, Kazza, and e-donkey have led to an increase in the amount of traffic, and there has also been a significant rise in the amount of multimedia traffic such as audio and video. One significant technological breakthrough which facilitated this growth was the introduction of congestion control [1], which allowed many users to share the network without causing congestion collapse. Although the best-effort service, which was used in the early days of the Internet, was adequate as long as the applications using the network were not sensitive to variations in losses and delays, it is no longer adequate, due to the explosion in the number of different applications. To solve this problem,

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 818–825, 2005.

in the last few years, there has been a wave of interest in providing network services with performance guarantees and in developing algorithms supporting different levels of services. The various solutions that have been proposed to solve these problems can be summarized under the general heading of Quality-of-Service (QoS).

QoS-enabled networks consist of major functions: 1) a TCP Congestion Control (TCC) mechanism at a source/sink node that dynamically adjusts the rate (or window size) in response to congestion in its path, and 2) an Active Queue Management (AQM) scheme at an intermediate node that updates, implicitly or explicitly, a congestion measure, drops (or marks) some packets in order to avoid network congestion, and sends these packets back, implicitly or explicitly, to the source or sink.

In this paper, we introduce the major TCC mechanisms and AQM schemes and describe the basic approaches that have been proposed. We also present analytical models of these TCC/AQM schemes for the purpose of identifying the reciprocal relationship between the TCC mechanisms and AQM schemes in QoS-enabled networks.

The rest of the paper is organized as follows. In section 2, we briefly review the major TCC mechanisms, such as TCP Tahoe [1], TCP Reno, and TCP Vegas [2], and the major AQM schemes, such as DropTail, random early detection (RED) [5], and Proportional Integral (PI) [6], and present analytical models of these TCC/AQM schemes. In section 3, we describe how network simulations are performed using *NS-2* [7] simulator. In section 4, we presents simulation-based comparisons of the TCC/AQM pairs. In the final section, we offer our concluding remarks.

2 Current TCC Mechanisms and AQM Schemes

2.1 TCC Mechanisms

TCC has three important features. The first is the "window" flow control feature. A source node maintains a variable called the window size that determines the maximum number of outstanding packets that have been transmitted, but not yet acknowledged. When the window size is exhausted, the source must wait for an acknowledgment before sending any new packets. In two features are important. The second is the "self-clocking" feature that automatically slows down the source when the network becomes congested and acknowledgments are delayed. The third is that the window size controls the source rate: roughly one window of packets is sent during each round-trip. We will briefly review the major TCC mechanisms and analytically model the average source rate of these mechanisms.

To model the average behavior of the additive increase, multiplicative decrease (AIMD) mechanism, we assume the following expressions. Let $w_i(t)$ be the window size. Let τ_i be the equilibrium round trip time (propagation plus equilibrium queueing delay), which is constant. Let $x_i(t)$ defined by $x_i(t) =$

$w_i(t)/\tau_i$ be the source rate at time t. Let $q_i(t)$ be the end-to-end marking probability to which source algorithm reacts. In period t, it transmits at rate $x_i(t)$ packets per unit time, and receives (positive and negative) acknowledgments at approximately the same rate, assuming every packets is acknowledged. Hence, on the average, source i receives $x_i(t)(1-q_i(t))$ number of positive acknowledgments per unit time and each positive acknowledgment increases the window $w_i(t)$ by $1/w_i(t)$. It receives, on the average, $x_i(t)q_i(t)$ negative acknowledgments (losses) per unit time and each halves the window. Hence, in period t, the net change to the window is roughly

$$x_i(t)(1-q_i(t))/w_i(t) - x_i(t)q_i(t)w_i(t)/2 \tag{1}$$

Whereas, the Vegas source determines the queueing delay by monitoring its round-trip time (the time between the sending of a packet and the receipt of its acknowledgment) and subtracting from this the round-trip propagation delay. Therefore, TCP Vegas is modelled as the following expression. Let d_i be the round trip propagation delay for source i and assume $\alpha_i = \beta_i$ for all i. Then the source rate is adjusted according to:

$$\frac{1}{(d_i + q_i(t))^2}\operatorname{sgn}\left(1 - \frac{x_i(t)q_i(t)}{\alpha_i d_i}\right) \tag{2}$$

where $\operatorname{sgn}(z)$ is -1 if $z < 0$, 0 if $z = 0$, and 1 if $z > 0$. Here, $q_i(t)$ is the sum of link queueing delays in the path of i at time t, $d_i + q_i(t)$ is the round trip time of i at time t, and $x_i(t)q_i(t)$ is the number of packets that are buffered in the queues in i's path. Hence (3) says that the window (rate $\times$ round trip time) is incremented or decremented by 1 packet per round trip time, according as the number $x_i(t)q_i(t)$ of packets buffered in the path is smaller or greater than the target $\alpha_i d_i$. In equilibrium, each source i maintains $\alpha_i d_i$ packets in its path.

The analytical models for each TCC mechanism are derived in [3] , as summarized in Table 1.

Table 1. Analytical Models of TCC mechanisms

TCC Mechanism	Analytical Model
Reno	$\bar{x} = \frac{1-q_i(t)}{\tau_i^2} - \frac{1}{2}q_i(t)x_i^2(t)$
Vegas	$\bar{x} = \frac{1}{(d_i+q_i(t))^2}\operatorname{sgn}\left(1 - \frac{x_i(t)q_i(t)}{\alpha_i d_i}\right)$
Parameters	τ_i: equilibrium round trip time for source i $x_i(t)$: source rate for source i at time t $q_i(t)$: end-to-end marking probability for source i at time t d_i: round trip propagation delay for source i α: control gain $\bar{x}$: average source rate for source i at time t

2.2 AQM Schemes

We provide a description of the basic schemes for IP network such as DropTail, RED [5], and PI [6] and present analytic models of their dropping (or marking) probability.

- **DropTail.** DropTail maintains exactly simple FIFO queues. There is no methods, configuration parameter, or state variables that are specific to drop tail queues.
- **RED.** RED [5] was presented with the objective to minimize packet loss and queueing delay, avoid global synchronization of sources, maintain high link utilization, and remove biases against bursty sources. To achieve these goals, RED utilizes two thresholds, min_{th} and max_{th}, and a exponentially-weighted moving average (EWMA) formula to estimate the average queue length, $Q_{avg} = (1-W_q)*Q_{avg}+W_q*Q$, where Q is the current queue length and W_q is a weight parameter, $0 \leq W_q \leq 1$. The two thresholds are used to establish three zones. If the average queue length is below the lower threshold (min_{th}), RED is in the normal operation zone and all packets are accepted. On the other hand, if it is above the higher threshold (max_{th}), RED is in the congestion control region and all incoming packets are dropped. If the average queue length is between both thresholds, RED is in the congestion avoidance region and the packets are discarded with a certain probability P_a:

$$P_a = \frac{P_b}{(1 - count \cdot P_b)} \tag{3}$$

  This probability is increased by two factors. A counter is incremented every time a packet arrives at the router and is queued, and reset whenever a packet is dropped. As the counter increases, the dropping probability also increases. In addition, the dropping probability also increases as the average queue length approaches the higher threshold. In implementing this, RED computes an intermediate probability P_b,

$$P_b = \frac{max_p}{max_{th} - min_{th}} \times (Q_{avg} - min_{th}) \tag{4}$$

  whose maximal value given by max_p is reached when the average queue length is equal to max_{th}. For a constant average queue length, all incoming packets have the same probability to get dropped. As a result, RED drops packets in proportion to the connections' share of the bandwidth.
- **PI.** PI [6] uses a feedback-based model for TCP arrival rates to let the queue occupancy converge to a target value, but assumes a priori knowledge of the round-trip times and of the number of flows traversing the router. It improves responsiveness of the TCP flow mechanisms by means of proportional control, stabilizes the queue length around target value q_{ref} by means of integral control, and marks each packet with a probability, p,

$$p(t+1) = p(t) + a(q(t+1) - q_{ref}) - b(q(t) - q_{ref}) \tag{5}$$

Two main functions are used in the PI algorithm: one is the congestion indicator (to detect congestion) and the other is the congestion control function (to avoid and control congestion). The PI-controller has been designed based on (1) not only to improve responsiveness of the TCP flow dynamics but also to stabilize the router queue length around Q_{ref}. The latter can be achieved by means of integral (I)-control, while the former can be achieved by means of proportional (P)-control using the instantaneous queue length rather than using the exponentially weighted moving average (EWMA) queue length.

3 Simulation Method

In the previous section, we showed that Internet congestion control is an independent but inter-related algorithm between TCC mechanisms and AQM schemes. To compare the current TCC/AQM pairs via simulation, in this section, we explain how we simulated the different schemes discussed in the previous section.

We perform three simulation experiments. In the first experiment, we compare the performance provided by the DropTail, RED [5], and PI [6] schemes, at a single node. The first experiment focuses on the performance issues from the point of view of the queueing information at a bottleneck link. In the second experiment, we compare different TCC/AQM pairs, in order to determine which pair of schemes provides the "best" performance under the same conditions.

Setting Up the Single-node Topology for Experiment 1 - In this experiment, we consider a bottleneck link with a bandwidth of 10 Mbps, a propagation delay of 10 ms, and a queue size of 150,000 bytes. The remaining links (edge links) all have a bandwidth of 100 Mbps and a propagation delay of 1 ms. Each source node is connected to the corresponding sink node at the other side of the network, i.e., source node S_i is connected to sink node Ri, as shown in Figure 1. Since numerous tutorials and manuals are available concerning the nodes and link objects in *NS-2*, we will not provide any further discussion on this subject. There are 3 TCP source/sinks and one UDP source/sink connected to each edge node. Each TCP source is an FTP application on top of NewReno TCP. The FTP packet size is 500 bytes. Each UDP source is a Pareto On-Off source with a peak rate of 5,000 Kbps, a burst time of 10 ms, and an idle time of 10 ms. The experiment lasts for 70 seconds of simulated time, and ECN is available in the entire network.

- **Drop-Tail**. We use DropTail to have an estimate of the performance measure encountered without AQM scheme. With DropTail queue, incoming packets are discarded only when the queue is full.
- **RED**. RED utilizes two thresholds, min_{th} and max_{th}, and an EWMA formula to estimate the average queue length. RED is configured with a minimum threshold min_{th} = 30,000 bytes, and a maximum threshold max_{th} = 120,000 bytes. Also, the parameter max_p is set to 1, and the weight used in the computation of the average queue size is set to $W_q = 0.002$.

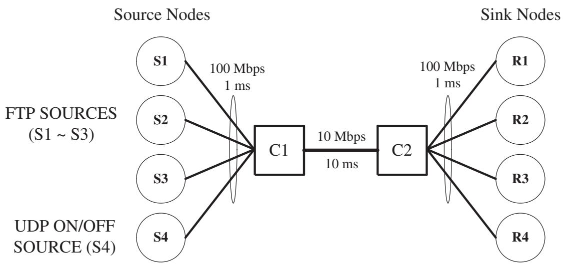


Fig. 1. Network topology with single node for per-node queueing behavior

- **PI**. We configure the PI algorithm with approximate RTTs and a tight upper bound on the round-trip times $R+ = 180$ ms, with a sampling frequency of 160 Hz, and get $a = 1.643\text{e} - 4$ and $b = 1.628\text{e} - 4$. The target queue length Q_{ref} is set to 70,000 bytes. Note that such a crude parameter tuning is to account for the uncertainty on estimates of the RTTs and of the number of flows at router configuration time.

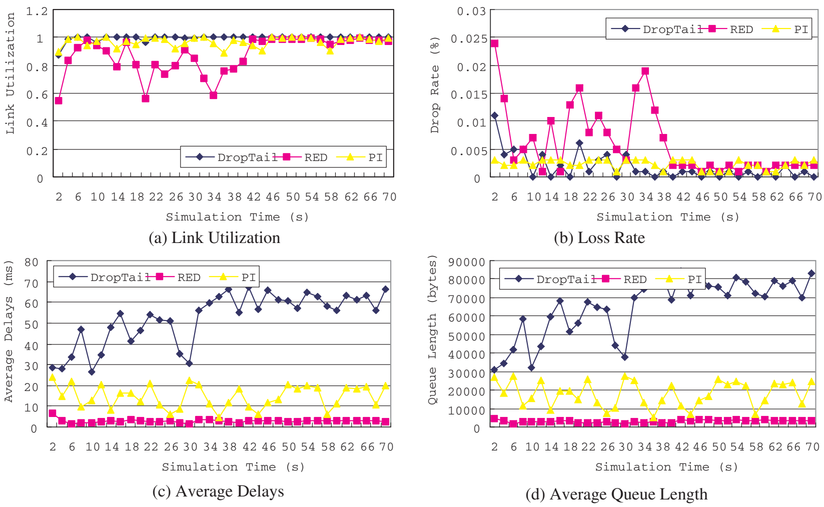
Setting Up the TCC/AQM pair for Experiment 2 - In this experiment, we use the same network topology and traffic pattern as those used in experiment 1. In addition, we implement the TCC mechanisms (i.e. TCP Tahoe, TCP Reno, TCP SACK, and NewReno) at the source/sink nodes and AQM schemes (i.e. DropTail, RED, and PI) at the core nodes.

4 Simulation Results

In this section, we present the simulation results for each queue discipline in terms of the per-node queueing behavior.

For each AQM scheme, we monitor the link utilization, loss rate, average delay, and average queue length at the bottleneck core link, as shown Figure 1, and present our results in Figure 2. It was found that the DropTail and RED schemes could achieve better link utilization than the PI scheme.

For each TCC/AQM pair, we monitor the end-to-end average loss rate and average delay, and present our results in Table 2. Considering the average loss rate, the AQM schemes under the NewReno mechanism provide relatively better performance than the other schemes. Note that the PI scheme can achieve lower-delay performance than the other schemes. Furthermore, the PI scheme was almost unaffected by the TCC mechanisms, as shown in the standard deviation of Table 2. This means that PI, which provides a suitable feedback mechanism, can help delay sensitive applications to adapt themselves dynamically to the underlying network and to stabilize the end-to-end QoS within an acceptable limit. The simulations show that the equilibrium and dynamics of the network depend on the harmony between the TCC/AQM pairs. In our opinion, an appropriate combination of active queue management from the network and source reaction



(a) Link Utilization (b) Loss Rate

(c) Average Delays (d) Average Queue Length

Fig. 2. Per-node queueing behavior with single node

Table 2. End-to-end performance results of TCP/AQM schemes

TCP/AQM Schemes	Average		Standard Deviation	
	Loss Rate (%)	Delay (ms)	Loss Rate (%)	Delay (ms)
Tahoe/DT	0.369	48.39	0.418	7.791
Tahoe/RED	0.343	50.27	0.393	8.917
Tahoe/PI	0.400	14.69	0.091	6.499
Reno/DT	0.283	44.46	0.384	14.355
Reno/RED	0.237	48.16	0.325	13.822
Reno/PI	0.329	15.77	0.141	7.213
SACK/DT	0.409	48.95	0.450	5.624
SACK/RED	0.397	49.82	0.427	6.545
SACK/PI	0.383	14.98	0.158	6.650
NewReno/DT	0.174	52.27	0.238	12.458
NewReno/RED	0.171	51.07	0.230	11.454
NewReno/PI	0.226	14.67	0.082	5.487

in needed to provide an effective solution to the instantaneous network fluctuations which occur on the Internet. The NewReno/PI pair, a feedback mechanism encompassing both network and end-systems, can enhance the performance of packet loss and delay sensitive applications.

In summary, the existing TCC mechanisms and AQM schemes have focused on seven main issues: 1) avoid congestion, 2) reduce the packet transfer delay, while keeping the queue lengths at low levels, 3) avoid the TCP global synchronization problem, 4) achieve fairness among different traffic types, 5) deliver service guarantees (guaranteed or differentiated), 6) reduce the program complexity, and 7) increase the scalability. These issues, however, are all inter-related.

5 Conclusion

We presented an analysis of the reciprocal relationship between TCC mechanisms and AQM schemes, by considering the average packet loss rate and average delay. The analysis provided herein has two objectives. First, we describe each algorithm's design goals and performance issues. Second, we compare the performance of the surveyed TCC/AQM pairs in QoS-enabled networks. To understand which pair of schemes in the most harmonious, we briefly reviewed the major TCC/AQM schemes, described their analytical models, and provided simulation-based comparisons of the TCC/AQM pairs under the same conditions. In addition, the method of analysis presented in this paper could be used as a basic means of identifying the behavior of network entities which are more complicated and diversified in terms of their per-node queueing information and per-flow end-to-end behavior.

References

1. V. Jacobson. "Congestion avoidance and control," ACM Computer Communication Review, 18:314-329, 1988.
2. L.S. Brakmo, L.L. Peterson. "TCP Vegas: end-to-end congestion avoidance on a global Internet," IEEE Journal on Selected Areas in Communications, 13(8):1465-80, 1995.
3. S.H. Low. "A duality model of TCP and queue management algorithms," Proceedings of ITC Specialist Seminar on IP Traffic Measurement, Modeling and Management, 2000.
4. S.H. Low, F. Paganini, and J.C. Doyle. "Internet congestion control," IEEE Control Systems Magazine, 22:28-43, 2002.
5. S. Floyd and V. Jacobson. "Random early detection for congestion avoidance," IEEE/ACM Transactions on Networking, 1(4):397-413, July 1993.
6. C.V. Hollot, V. Misra, D. Towsley and W. Gong. "On designing improved controllers for AQM routers supporting TCP flows," In proceedings of IEEE INFOCOM 2001, volume 3, pp 1726-1734, Anchorage, AK, April 2001.
7. ns-2 network simulator. http://www.isi.edu/nsnam/ns/.

Choice of Inner Switching Mechanisms in Terabit Router

Huaxi Gu[1], Zhiliang Qiu[1], Zengji Liu[1], Guochang Kang[1],
Kun Wang[2], and Feng Hong[3]

[1] State key lab of ISN, Xidian University, Xi'an, China 710071
{hxgu, zjliu}@xidian.edu.cn,
zlqiu@mail.xidian.edu.cn, gckang@163.com
[2] School of Computer Science, Xidian University, Xi'an, China 710071
kwang@mail.xidian.edu.cn
[3] HUAWEI TECHNOLOGIES CO.LTD., Shenzhen, China 518129
hongf@huawei.com

Abstract. More and more attention is focused on direct interconnection networks when designing the switching fabrics in the terabit routers. Various switching mechanisms are proposed for multi-computer systems, which also rely on direct interconnection networks between processors to support the messages passing mechanism. But it remains unknown which one is more suitable for fabrics in the terabit routers. Based on the requirements of terabit class routers we made analysis and simulations on various switching mechanisms, such as store and forward, wormhole switching, virtual cut through switching and pipelined circuit switching. The results show that virtual cut through exhibits superior performance characteristics over other switching mechanisms under various conditions. Simulations of the performances of virtual cut through shows that larger buffer, longer flit and more virtual channels help to sustain higher throughput at the cost of increasing latency.

1 Introduction

Historically, routers have used backplane bus and crossbar switches as their switching fabrics. However, bus architecture is not sufficient for more than very few Gbps speed ports. Crossbars cannot economically scale to large number of nodes since the cost grows as the square of the number of nodes. Direct interconnection network (DIN) [1] has drawn much interest recently as a promising candidate for high-speed and high-performance fabrics in terabit class routers. For example, Avici Systems uses 3-D torus as switching fabrics in their terabit router AVICI TSR [2], while Pluris makes use of hypercube in TeraPlex20 [3].

Switching mechanism defines how messages propagate through the switching fabrics. A variety of switching mechanisms have been proposed [4], among which are: circuit switching (CS), store and forward (S&F), wormhole switching (WS), virtual cut through switching (VCT), pipelined circuit switching (PCS) and adaptive cut through switching (ACTS). These switching mechanisms are mainly designed for multi-computer systems while the core router presents a somewhat different set of requirements. For example, most commercial multi-computer systems implement WS. But it is not well suited for fabrics in terabit routers for its relatively low throughput.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 826–833, 2005.

In this paper, we make analyses of the five popular switching mechanisms based on the requirements of terabit class routers. The results show that VCT outperforms other switching mechanisms in many ways. The rest of the paper is organized as follows: In Section 2, comparisons of the switching mechanisms are presented. Attention is focused on advantages and disadvantages of the five type of switching mechanisms: CS, WS, VCT, PS and PCS. In section 3 we introduce the evaluation methodology, including the routing algorithm used in the simulations. Section 4 describes the different simulations performed, as well as the results obtained. In section 5 we conclude this paper and provide an outlook to future research.

2 Comparisons of the Switching Mechanisms

2.1 Various Switching Mechanisms

In this section, various switching mechanisms are compared including circuit switching, store and forward, wormhole switching, virtual cut through switching and pipelined circuit switching.

The oldest technique, which is not popular anymore in parallel computers, is circuit switching. In CS, messages are not divided into parts. Before a message is transmitted, a complete path will be established from source to destination by sending a probe. Once the transmission completes, the path will be torn down. The most notable feature of CS is its provision of guaranteed latency once the connection is set up. But link utilization is relatively low in CS, which lead to less use in modern multi-compute systems.

In S&F, a message is divided into packets that are independently routed towards their destinations. Each packet contains the destination address and alternative paths can be selected upon encountering network congestion or faulty nodes. Before it is forwarded to the next node, the entire packet has to be stored in the current node. Therefore, the time to transfer a packet from source to destination is directly proportional to the number of hops in the path.

In WS, packets are sub-divided into a sequence of smaller units called flits. The first flit is used to determine the route and the remaining data flits follow in a pipeline fashion (the last flit releases the reserved connections). The network latency for WS is $(L_f/B)D+L/B$, where B is channel bandwidth, D is the number of hops and L, L_f is the length of the packet and the flit respectively. Thus, the latency of WS is insensitive to the distance D for $L_f \ll L$.

VCT behaves in the same way as WS except when the requested outgoing channel is busy. VCT buffers the whole packet at the local node while WS stalls the packet at each node along the path up to the current node. Therefore, VCT can achieve higher throughput than WS for this reason. This effect becomes particularly evident with heavy network traffic.

PCS is a combination of CS and WS. In PCS, the data flits are waiting in the source before a path is established from source to destination by the header flit. PCS is a reliable switching mechanism, since fault tolerant routing algorithm can be easily designed. The most notable advantage of PCS is its ability to provide messages with an agreed upon service, e.g. guaranteed latency, once the connection is established. But the advantages of PCS are obtained at the expense of longer path-set-up time. For short messages, higher latency and low throughput will be the penalties from PCS.

2.2 Choice of Switching Mechanisms in Terabit Router

In several core routers, the switching fabrics internally operate on fixed-size data units. Examples of such routers and switches can be found in both commercial products and laboratory prototypes, such as Cisco GSR [5], the Tiny-Tera [6] and so on. Using fixed-size data units in the switch has many advantages. For example it can make the implementation much easier compared to the variable-length packets. Therefore, VCT, WS and PCS are preferred since they cut packets into fixed size data units.

On the other hand, the terabit class routers have to handle a large number of high-speed ports. So fabrics may be expanded to large scale. But the delay of S&F is proportional to the distance. Hence, higher delay is obtained for large-scale fabrics if S&F is used. The distance does not heavily affect the latency of WS and VCT, so delay-sensitive real time application will greatly benefit from VCT and WS because of their shorter latency. Therefore, they are suitable for the scalable fabrics in terabit class routers. However, considering the heavy traffic faced by the terabit router, VCT is more suitable than WS and PCS.

Since VCT propagates a packet all the way to its destination, if a packet is corrupted, it may not be able to fully be removed from the network until it reaches the destination, thus wasting bandwidth. This is a disadvantage when cut through is used in Internet environment since the error rate is relatively high. But in switching fabric of terabit routers, the error rate for a data packet is very low. Hence, this disadvantage is no longer a big problem.

From discussions above, we can see VCT performs efficiently while imposing less constraint. It is better suited for the terabit routers. To enhance our analysis, we have made different simulations as follows.

3 Evaluation Methodology

To evaluate the various switching mechanisms we use one of the most powerful software simulation package-OPNET [7]. OPNET provides a comprehensive development environment for the specification, simulation and performance analysis of communication networks.

The simulations are carried on a 4×4×4 3D torus network due to the popularity of this topology in many systems [2, 4, 8]. The routing algorithm used in the simulations is proposed by Duato in [9]. It has been accepted by many real systems such as the Cray T3E [8], Reliable Router [10] and so on. In the case of 3D torus, the algorithm requires at least three virtual channels (VC), which are divided into two classes *a* and *b*. Class *b* contains two virtual channels, in which deterministic routing is applied. The rest virtual channels belong to class *a*, where fully adaptive routing is used. The messages can adaptively choose any virtual channels available from class *a*. If all the virtual channels of class *a* are busy, the message enter channels that belong to class *b*.

Each node operates asynchronously. They generate packets at time interval chosen from a negative exponential distribution. Unlike the traditional use of fixed-length packets [4, 8], we use two kinds of packet length distributions. One is uniform distribution, which ranges from 64 to 1500 bytes. The other is a specific distribution SP (Size and Percent) that is based on the IP (Internet Protocol) packet size and percentages sampled over a two-week period [11]: 40 bytes (56 % of all traffic), 1500 bytes (23 %), 576 bytes (16.5 %) and 52 bytes (4.5 %). Recent studies have revealed that traffic in Internet can exhibit a high degree of burst, but the traditional Poisson arrival process is unable to model traffic burst. So we also use an on-off source in the simulation. To the best of our knowledge, ours is the first attempt to incorporate such source and packet length distributions into evaluating performance of direct interconnection networks. Such configurations of simulation environments are more close to reality, which makes the results more convincing.

The performance of the switching mechanisms is measured in terms of ETE (End to End) delay and throughput. Loss rate is used as the metrics of the throughput. In all the figures presented below, the horizontal axis represents the injected traffic into the network while the vertical axis shows the ETE delay or loss rate.

4 Simulation Results

In this section, we show by analysis and simulations that, in the case of terabit class routers, VCT may provide performance advantages over other switching mechanisms. We first compare the performance of different switching mechanisms. Then we evaluate the performance of VCT under various working conditions.

4.1 Comparisons of WS, VCT, PS and PCS

Figure 1 depicts latency results of the four popular switching mechanisms under various conditions. The figures reveal that when the offered traffic is between 0.1 and 0.2 T bit/s, WS and VCT have almost the same ETE delays. This is due to their similar behavior under light traffic load. The latency of PCS is greater than that of WS or VCT because of the path set up time. The reason for highest latency of S&F is that the packets are stored node by node. When the traffic increases, the performance merits of VCT become apparent since less contention occurs in VCT than those in WS.

On the other hand, the four figures reveal that VCT can sustain higher traffic load than the other three. This is due to the fact that the blocked packets remained in the network and kept resources previously reserved. Therefore, VCT achieves lower ETE delay and sustains higher traffic load under different simulation environments among the four candidates. The feature of low latency of VCT meets the requirement of some real time traffic. Hence, VCT is a cost effective method to support QoS in the fabrics of the terabit class routers. On the other hand, since backbone routers are faced with heavy traffic, high throughput of VCT satisfies another requirement of the core routers.

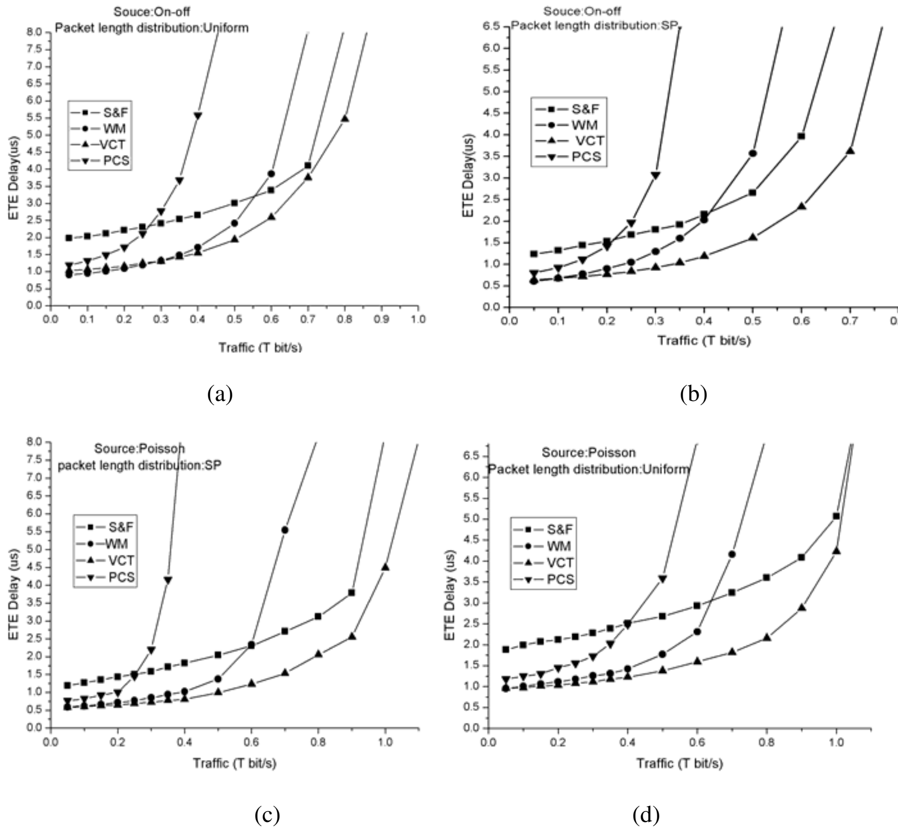
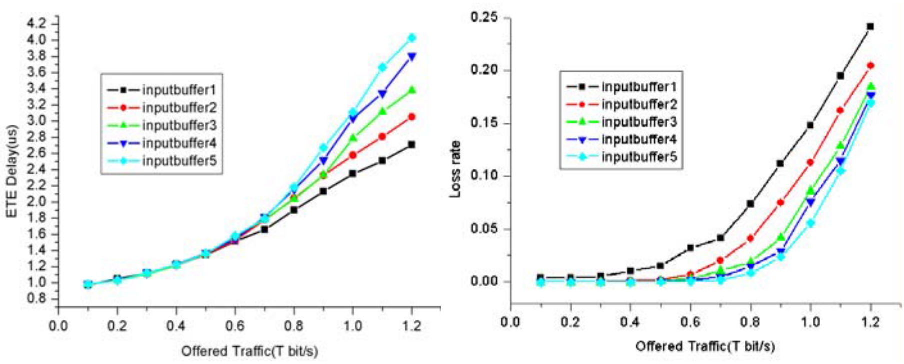


Fig. 1. Comparisons of the four switching mechanisms under different environments

4.2 Performance of VCT Under Various Working Conditions

Figure 2-Figure 4 show the results of the simulations with 3D torus topology by using VCT. We evaluate the effect of input buffer size, flit length and number of virtual channels on the network performance. The buffer size means the number of maximum-size packets (1500 bytes) that can be stored.

Figure 2 shows the effect of input buffer size on the network performance. As the buffer size increases, the loss rate becomes lower. For example, when the offered traffic is equal to 0.8 T bit/s, the loss rate has been improved by 8% with the buffer size increasing from 1 to 5. On the other hand, the improvement of the throughput is at the cost of increase of the latency. In Figure 2(a), as the buffer size increases, the ETE delay increase too. Lager buffer store more packets and more contention occur, which leads to larger ETE delay. When the buffer is large enough, the performance cannot be improved much by adding more buffers.

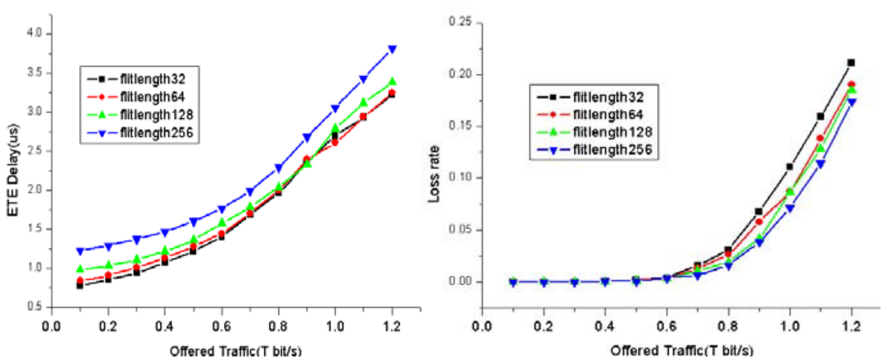


(a) ETE delay (b) Throughput

Fig. 2. Effect of input buffer size on the network performance

Figure 3 reveals the effect of flit length on the network performance. The flit length varies from 32, 64,128 to 256 bytes. Both the number of VC and the buffer size are 3. Observation from Figure 3 (a) is that the latency performance has been improved with shorter flits. For example, the latency performance has been improved by 24% at offered traffic 0.5T bit/s (moderate traffic load) when the flit length decreases from 256 to 32 bytes. For the offered traffic of 0.8 T bit/s (heavy traffic load), the latency has been improved by14.5%. The reason is that sending a longer flit takes more time and thus increases the waiting time of other flits. What' more, long flits are easily blocked by each other under heavy traffic load.

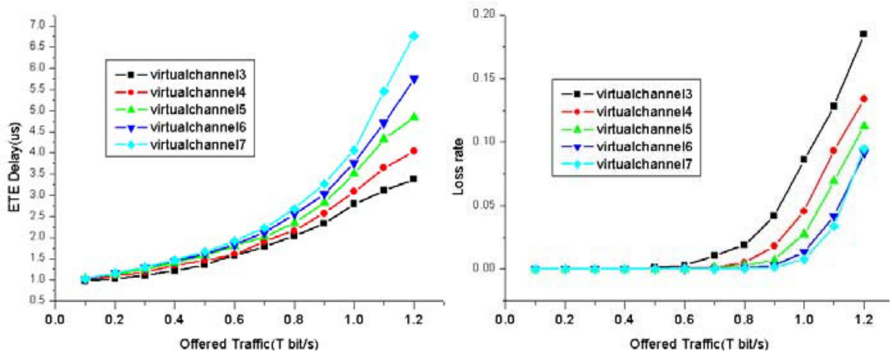
In Figure 3 (b), the four curves almost overlap at moderate traffic load (0.6 T bit/s or less). As the traffic increases, longer flits can help to sustain higher throughput. In the simulation, the average packet size is 782 bytes. If the flit length is 32 bytes, the average number of flits will be about 24. If 256 bytes, there are just about 3 flits. The more flits, the more contention occurs, thus increasing the loss rate.



(a) ETE delay (b)Throughput

Fig. 3. Effect of flit length on the network performance

Figure 4 shows the ETE delay and throughput when the number of the virtual channels is varied from 3 to 7. The input buffer is 3 and flit size is 128byte. With low traffic loads, the variation of the number of virtual channels has little influence upon the ETE delay. But as the traffic load increases, the latency gets higher with more virtual channels. In addition, Figure 4(b) reveals that increasing the number of VC decreases the loss rate. The latency increase from 3.1 to 5.5 and the loss rate drops from 12.8 % to 3.4% when the number of VC varies from 3 to 7(The offered traffic is 1.1 Tbit/s).



(a) ETE delay (b) Throughput

Fig. 4. Effect of the number of the virtual channels on the network performance

Virtual channels allow new messages to bypass the blocked message, leading to better utilization of link bandwidth and thus increase throughput. But virtual channels arbitration and multiplexing introduce additional delays, so more virtual channels cause higher latency.

5 Conclusions and Future Work

Various switching mechanisms are proposed in literature such as circuit switching, store and forward, wormhole switching, virtual cut through and pipelined circuit switching. In this paper, we compare their advantages and disadvantages when used in fabrics of terabit router. The comparison results suggest that, contrary to the current designs in multi-computer system, VCT is better suited than other switching mechanisms for the terabit router.

Our future study is to develop an analytical model of the four switching mechanisms. It will provide cost-effective and efficient tools that requires less computation time than simulation. On the other hand, the explosive increase in multimedia applications implies new requirements on the core routers. The routers must therefore provide different QoS requirements to offer efficient, predictable services to multimedia flows. Thus, hybrid switching is another topic in future research, i.e., using connection-oriented switching like circuit switching to support delay sensitive traffic while use virtual cut through to support best-effort traffic.

Acknowledgment

This work was supported by the National High-tech Research and Development Plan of China under grant No.2002AA103062 and No.2003AA103520.

References

1. L. Ni, M. and P. McKinley, K., A Survey of Wormhole Routing Techniques in Directed Networks, Computer vol. 26, 1993, 62-76,
2. William J. Dally, Scalable Switching Fabrics for Internet Routers, White paper, Avici Systems Inc. 2001
3. Pluris, Inc., Avoiding Future Internet Backbone Bottlenecks, white papers, On the Web at http://www.pluris.com, 2000.
4. J. Duato, S. Yalamanchili and L. Ni., Interconnection Networks, an Engineering Approach, Morgan-Kaufmann Press.2002.
5. Cisco 12000 Gigabit Switch Router, Product Overview, www.cisco.com, Apr. 2000
6. McKeown N., et al., The Tiny Tera: a packet switch core, IEEE Micro Magazine, vol. 17, Feb. 1997, 27-40
7. OPNET Modeler, OPNET Modeler manuals, MIL 3, Inc.3400 International Drive NW, Washington DC 20008 USA,1989-2004.
8. Ed Anderson, Jeff Brooks, Charles Grassl, and Steve Scott. Performance of the Cray T3E multiprocessor. In Supercomputing 97, San Jose, California, November 1997,1-17
9. J. Duato, A New Theory of Deadlock-Free Adaptive Routing in Wormhole Networks, IEEE Trans. on Parallel and Distributed Systems vol. 4, 1993,1320-1331
10. W. J. Dally, et al., The Reliable Router: A reliable and high-performance communication substrate for parallel computers, Proceedings of the Workshop on Parallel Computer Routing and Communication, May 1994, 241--255
11. David Newman, Internet Core Router Test, www.lightreading.com. March 6, 2001

Effect of Unbalanced Bursty Traffic on Memory-Sharing Schemes for Internet Switching Architecture[1]

Alvaro Munoz and Sanjeev Kumar

Senior Member, IEEE, Department of Electrical Engineering,
The University of Texas – Pan American, Edinburg, Texas-78539, USA
{amvargas1, sanjeevk}@utpa.edu
Ph: 956-381-2401

Abstract. Shared-memory based packet switches are increasingly being used for high-performance Internet switches and routers. The shared-memory switches are known to provide better throughput and packet-loss performance for bursty data traffic in high-speed networks and Internets compared with other buffering strategies under conditions of identical memory resource deployed in the shared-memory switch. The scheme to share the common memory resource among various broadband lines has direct impact on the throughput and packet-loss performance of the switch. In this paper, we compare the effect of unbalanced bursty traffic on commonly used memory-sharing schemes, namely the individual-static threshold based, global-static threshold based, dynamic threshold based and SMDA based memory-sharing schemes. *Index terms*—Shared Memory, Packet Switch, Unbalanced bursty Traffic, Memory-Sharing Schemes.

1 Introduction

Switching systems employing shared memory have been known to provide highest throughput and incur the lowest packet-loss compared to that of packet switches employing input or output buffering strategies under conditions of identical memory size and bursty traffic. The memory-sharing schemes have direct impact on the throughput performance and utilization of its output ports [1]–[4]. This paper presents a performance comparison of commonly used memory-sharing schemes for the class of shared-memory packet switches under conditions of unbalanced bursty traffic.

2 Background

A shared memory switch allows multiple broadband lines to share a common memory space for queuing packets bound for various output-ports of the switch. It is common to allow some kind of control on sharing of the common memory space among the

---

[1] Authors are with the Networking Research Lab (NRL) at UTPA. The research work of Dr. Kumar is supported in part by fundings from CITeC, FRC, FDC, and OBRR.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 834–841, 2005.

packets for different output ports of the switch. In the case of complete memory sharing, it is possible for packets of a given output port or a group of output ports (monopolizing ports) to completely occupy the common memory space and in effect block the passage of packets belonging to non-monopolizing ports of the switch. Furthermore, an unbalanced distribution of packets to the output ports could make the problem worse as packets of an output port arrive in bursts. In order to alleviate this problem of unfairness, it is common to restrict the occupancy of the common-memory space in order to always allow passage to packets of all input-output pairs. In this paper, we compare the impact of various memory-sharing schemes on the throughput and packet loss performance of a shared-memory switch under conditions of unbalanced bursty traffic.

3 Individual-Static Threshold Based Sharing Scheme

This is a straightforward scheme used to control, on an individual basis, the output-queue build-up inside the shared-memory switch. Under this scheme, a restriction is placed on the maximum length of the output queues [1] to a pre-determined value which is defined as the individual-static threshold value (ST). This ST value is set to a multiple α of the total buffer space (B).

$$ST = \alpha \cdot B \textit{ packets (where } 0 < \alpha \leq 1) \tag{1}$$

An individual output queue (Q_i) inside the common memory-space is not allowed to exceed the ST value. The packets of the output-queues that exceed the ST value are dropped. This scheme prevents any individual output-queue from completely occupying the common memory space and hence attempts to improve fairness and switch throughput. This method of restricting the maximum length of individual queue works well in preventing a single output queue from completely occupying the common memory space. However, at higher loads, it is still possible for a group of output queues to completely occupy the common memory space and unfairly deny (drop) the packets belonging to other source-destination pairs to access the common memory space for switching purposes.

4 Global-Static Threshold Based Sharing Scheme

According to this scheme, a restriction is put on the occupancy status of the entire global memory space. In this scheme, a predetermined limit, called the global-static threshold (GT) is imposed on the occupancy of the global memory space (B).

$$GT = (1-\alpha) \cdot B \textit{ packets (where } 0 < \alpha \leq 1) \tag{2}$$

If the occupancy of the global memory space reaches that threshold (GT) then the packets only from qualifying output ports are admitted to the remaining memory space = ($\alpha \cdot B$) packets. A predetermined admittance policy is used to qualify the output ports whose packets will be admitted in the remaining memory space. An example of an admittance policy is given in the section below.

A. Admittance Policy for Qualifying Output-Ports

Once the global-static threshold (GT) is reached on the occupancy of the entire global memory space then the admittance policy accepts packets for only those output ports whose output-queue length is less than ($\alpha \cdot B$) packets. Where B is the total shared memory space and α is a proportionality constant (where, $0 < \alpha \leq 1$) imposed on the occupancy of global memory space (B).

5 Dynamic Threshold Based Sharing Scheme

Dynamic threshold based memory-sharing scheme is described in detail in [3]. According to this scheme, the occupancy of the buffer that dynamically changes with the traffic conditions impose dynamically changing restrictions on the active output ports from entering the remaining memory space at any given time. Each queue length (Qi) inside common memory space is limited to a predetermined value called the dynamic threshold (DT) value. This DT value is function of the remaining buffer space and it could increase or decrease depending on the traffic conditions at time t. Let B be the total buffer space and ΣQi the sum of all queue lengths, (i.e., the total memory occupied by packets) then the dynamic threshold DT value at time t is calculated:

$$DT(t) = \alpha \cdot (B - \Sigma Qi) \textit{ packets } (\textit{where } \alpha > 0) \tag{3}$$

Where α is proportionality constant of the available memory ($B - \Sigma Q_i$) space at time t. Packets belonging to output queue *i* whose queue-length (Q_i) is less than DT are allowed to be stored in the remaining buffer space; otherwise packets are dropped. Dynamic threshold scheme is inherently adaptive and dynamically respond in time according to the unused memory space. If there is sufficient buffer space it allows active output ports to increase their output queues as much as necessary. Contrary, if the buffer nearly overflows it imposes very restrictive conditions in a way that only packet for less active ports are accepted. DT scheme reduces queue lengths by blocking new arrivals for the active ports, and waits for the queues of active ports to reduce naturally by the work of the switching system.

6 SMDA Based Memory Sharing Scheme

Another memory-sharing scheme, namely the shared-memory with dedicated access (SMDA) is similar to scheme called sharing with minimum allocation (SMA) scheme mentioned in [2]. SMDA or SMA based memory-sharing scheme aims to guarantee full utilization of the output ports first, and then attempts to maximize the throughput for a given bursty traffic. Under this scheme, a packet switch uses both the shared memory and dedicated memory for its output ports. A small percentage of total memory is dedicated to each output port and the remaining memory is shared among all the ports. For a given output-port, the dedicated memory is first used to store the packets and when the dedicated portion of the memory is full then only the packets can access shared memory space of the switch. Dedicated memory space for the SMDA scheme represents the minimum number of packet locations within memory space allocated to each output port for its individual use and is calculated as following.

$$\textit{Dedicated memory per port} = \alpha \cdot B \,/\, N \textit{ packets} \tag{4}$$

A portion of the total memory space B is divided equally among all the N ports for its dedicated use. The amount of remaining memory space is shared among all the ports and is calculated as following.

$$\textit{Shared memory space per port} = (1\text{-}\alpha)\cdot B \textit{ packets} \tag{5}$$

Here, B is the total memory space, and N is the number of I/O ports for NxN packet switch. Under this scheme, when the shared memory space is occupied due to traffic backlog then the inactive ports still have a dedicated memory to allow its packets a passage through the memory space. Unlike other memory-sharing schemes, the SMDA or SMA scheme guarantee full output-utilization even under the conditions of backlog that is common with bursty Internet traffic.

7 Performance Evaluation

For performance evaluation, the shared-memory switch is considered of size NxN = 32x32 ports and the input and output ports are operated at same speed. The total memory-size in switch is considered 1024 packets. The memory-sharing schemes presented in this paper regulate the length of the logical queues inside the common memory space. A bursty source with an average burst length (ABL) = 16 packets is used to generate traffic for each input ports. The bursty traffic is generated using a two state ON-OFF model i.e. by alternating a geometrically distributed period during which no arrivals occur (idle period), by a geometrically distributed period during which arrivals occur (active period) in a Bernoulli fashion and vice versa. Because of the uneven distribution of bursts in the simulated unbalanced bursty traffic, some ports have a greater chance to receiving bursts than the others. This unbalanced bursty traffic scenario produces two classes of output ports: very active output ports and lightly active output ports. The 50% of the output ports, though a high number, are considered very active ports for this simulation. The probability that a burst of packets is designated to a very active port is four times greater than that of a lightly active port in this simulations study.

Throughput versus α parameter is shown in Fig. 2 for all memory-sharing schemes under at 90% load. A high load (90%) intensifies the difference among the various memory-sharing schemes. The interval of α parameter extends in (0, 1) for all memory-sharing schemes except for the dynamic threshold scheme (where α could be any value greater than zero). Individual-static threshold and global-static threshold based sharing schemes have a higher throughput at small α value. Individual-static threshold scheme exhibit an acceptable performance for smaller values of $\alpha < 0.1$. This interval is very small for the possible α values and indicates that a good throughput performance is obtained at high loads when the queue lengths inside memory space are limited to short values. Similarly global-static threshold scheme shows an adequate performance for smaller values of $\alpha < 0.1$. For the applied load of 90% of unbalanced bursty traffic, both schemes namely the individual-static and global-static schemes experience a rapid degradation in performance when the α value is increased. SMDA and dynamic threshold based sharing schemes are very stable in variations of α parameter. At high loads (90%) throughput increases slightly in SMDA scheme with greater α values, nevertheless this means that a large percentage of memory is dedicated to each output port, which reduces the advantages of the sharing effect. Dynamic threshold scheme shows a better performance with variations in α parameter, where α value could be greater than one. Fig. 3 shows the throughput versus α parameter, for all memory-sharing schemes at 60% load. Decreasing the switch-load slow down the throughput variations with different α values compared to Fig. 2.

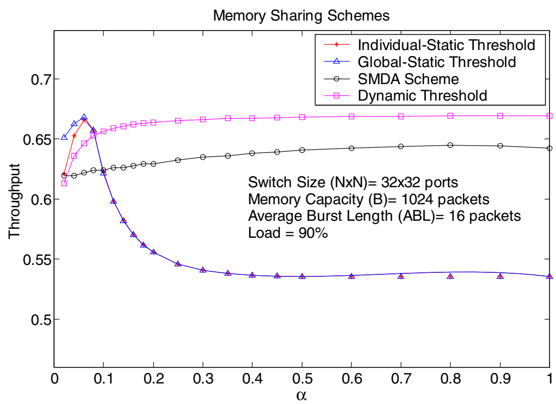


Fig. 1. Throughput vs. α for different memory sharing schemes at 90% load

Throughput performance for individual-static threshold and global-static threshold schemes for 60% of applied load (Fig. 3) suffer a notable variation within the range they perform well at high load of 90% (Fig. 2). SMDA scheme for 60% of applied load (Fig.3) presents a decrease in throughput as α is incremented. This is in contrast to the throughput value at higher load of 90% (Fig.2). It is apparent that SMDA scheme performs well under overload conditions. However, for smaller loads of 60%, the SMDA throughput somewhat decreases with increase in α value. This phenomenon occurs at low load because when α is incremented more memory space is reserved for each output port (dedicated buffer), decreasing the advantages of shared memory and hence the chances are greater that some ports have idle buffer while other ports are discarding packets due to a lack of space to store incoming packets. Dynamic threshold scheme performs similarly at high and low loads. It adapts to the changing traffic conditions, while there is a high occupancy of the memory space only packets from underrepresented output ports are accepted to the remaining buffer space. Figures 4-6 in this paper present packet lost ratio (PLR) for each memory-sharing scheme. Because of the unbalanced distribution of traffic to the output ports there are two classes of output ports, and packet-loss is evaluated individually for each class. As expected very active output ports incur a higher packet-loss compared to lightly active output ports.

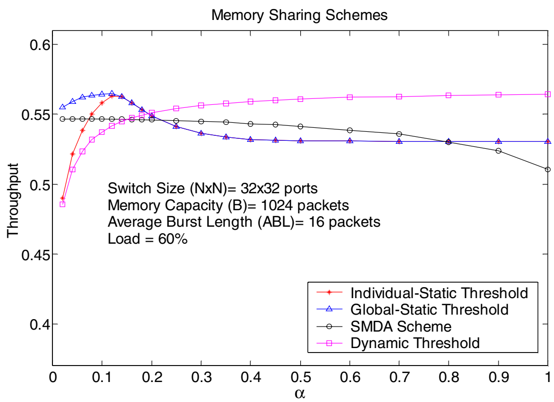


Fig. 2. Throughput vs. α parameter for different memory sharing schemes at 60% load

A good sharing policy should allow packets for less active ports to have access to the memory resources despite the overload conditions in the switch, and hence increase the fairness and utilization of the switching system. Fig. 4 presents PLR versus load using individual-static threshold scheme to control the sharing of the memory space.

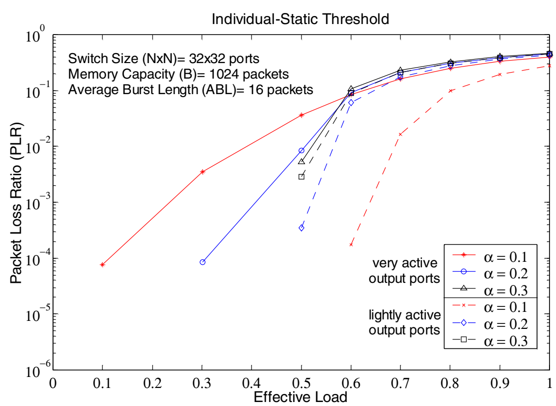


Fig. 3. PLR versus load for individual-static threshold based sharing scheme

Three different values of α parameter (α = 0.1, 0.2, and 0.3) are evaluated for both groups of ports. PLR at α = 0.1 shows a marked difference for very active and lightly active output ports. However there is packet-loss at low loads (10%) for very active port due to the fact that queue lengths are very restricted in size. Greater α values (α =

0.1, and 0.2) causes more packets for lightly active ports to be dropped and the PLR increases to levels similar to that present in very active ports. Compared to individual-static threshold scheme, Global-static threshold scheme doesn't suffer from packet-loss at low loads (Fig. 5). Both sharing schemes present similar levels of packet-loss at higher loads. Fig. 6 shows PLR for SMDA based sharing scheme.

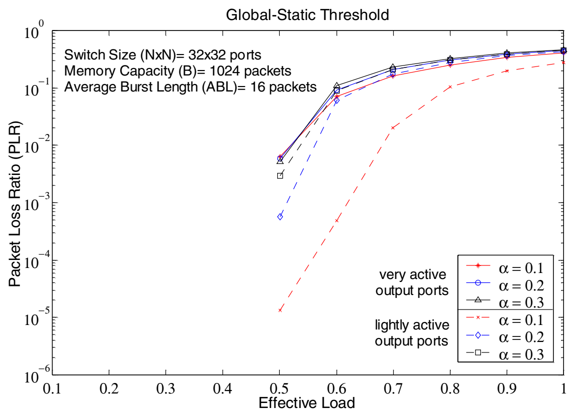


Fig. 4. PLR versus load for global-static threshold based sharing scheme

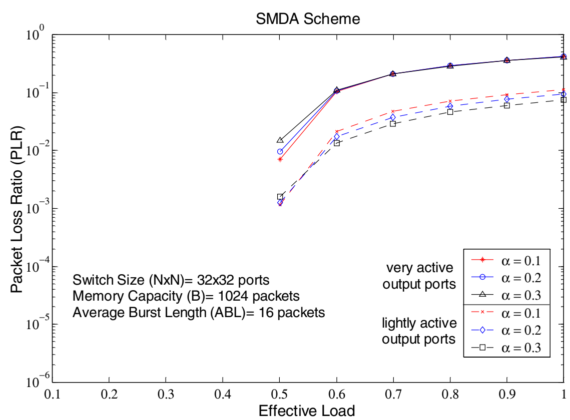


Fig. 5. PLR versus load for the SMDA based sharing scheme

Packets for lightly loaded ports always have access to the switch due to its pre-reserved memory per port. The levels of packet-loss for lightly active ports will depend on the amount of memory reserved and the applied load. For unbalanced bursty traffic, the dynamic threshold scheme has a superior performance compared to other memory-sharing schemes. Packet-loss for lightly active ports is very low. This scheme provides the best access for packets of underrepresented ports. The dynamic nature of this scheme provides high performance at both, high and low loads.

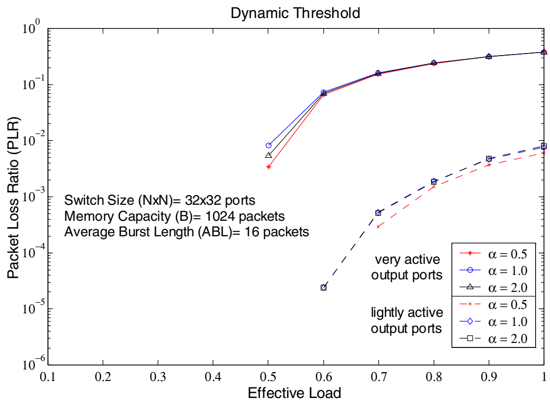


Fig. 6. Packet-loss ratio (PLR) versus load for dynamic threshold based sharing scheme

8 Conclusion

In this paper, performance of commonly used memory-sharing schemes, namely the individual-static threshold based, global-static threshold based, dynamic threshold based and SMDA based memory-sharing schemes have been compared under conditions of unbalanced bursty traffic and identical memory resources deployed in a shared-memory based switch. In individual-static threshold based and global-static threshold based memory-sharing schemes, it is difficult to find a fixed threshold value that works well both at high and low loads. SMDA based memory-sharing schemes provide a fair access to packets belonging to underrepresented output ports. Where dynamic threshold based scheme provides the lowest packet-loss for less active output ports and the best throughput performance under unbalanced bursty traffic.

References

[1] Irland, M.I., "Buffer management in a packet switch," *IEEE Transactions of Communications*, vol.26, pp. 328-337, 1978.

[2] Kamoun, F. and Kleinrock, L., "Analysis of shared finite storage in a computer node environment under general traffic conditions," *IEEE Transactions of Communications*, vol.28, pp.992-1003, 1980.

[3] Choudhury, A.K. and Hahne, E.L., "Dynamic queue length thresholds for shared-memory packet switches," *IEEE/ACM Transactions of Networking*, vol.6, no.2, pp.130-140, 1998.

[4] Kumar S., "The Sliding-Window Packet Switch: A new class of packet switch architecture with plural memory modules and decentralized control," *IEEE Journal on Selected Areas in Communications*, vol. 21, no.4, pp. 656-673, May 2003.

New Layouts for Multi-stage Interconnection Networks

Ibrahim Cahit[1] and Ahmet Adalier[2]

[1] Department of Computer Engineering, Near East University, Nicosia, Cyprus
icahit@neu.edu.tr
[2] Department of Computer Information Systems,
Cyprus International University, Nicosia, Cyprus
aadalier@ciu.edu.tr

Abstract. In this paper, we present new layouts for the multi-stage interconnection networks such as shuffle, baseline and banyan networks that are suitable for photonic switching. In these new layouts, we decrease the number of crossovers of the stage links and crossovers between inlet-outlet of stages, which are known as the main bottleneck for the increase in switch capacity when it is realized for integrated photonic switching fabric.

1 Introduction

Advances in the photonic switching systems have been reported in the literature and several new switch architectures are introduced to cope with the need of terabits/s volume of the future switches [1,2,3,4]. For example, Nishio et al. [5] has considered a photonic ATM switch using vertical to surface transmission electro-photonic devices (VSTEPs) to handle optical cell rates up to 1.6 Gbps in the optical buffer memory and self routing with priority controlled switches. Sawano et al. [6] has considered polarization independent $LiNbO_i$ matrix switches in their design with a maximum capacity of 128 lines photonic (circuit) switching systems. In both designs, the main bottleneck is the increase in the capacity, which is prevented by weakened optical signals from any inlet to any outlet in the switch fabric. Optical amplifiers have been used between the stages to compensate for the optical signal losses. Even this couldn't completely solve the capacity problem of the photonic switch. Very recently, Yanik et al. have proposed an easy and practical way of storing optical signals [7],[8].

This paper presents new layouts for multi-stage interconnection networks by investigating the following two characteristics of the multi-stage shuffle, baseline and banyan interconnection networks that are shown in Figure 1.

1. Minimization of the total number of crossovers in a switching network, which is related to the overall complexity of the fabrication process.
2. Minimization of the maximum number of crossovers between an inlet-outlet pair, which is related to the worst case attenuation which then determines the required number of optical amplifiers.

The outline of the paper is as follows. In section 2, we introduced crossover minimization via topological embedding and cyclical drawing of shuffle, baseline and banyan graphs. Finally, in section 3, some conclusions will be drawn.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 842–848, 2005.

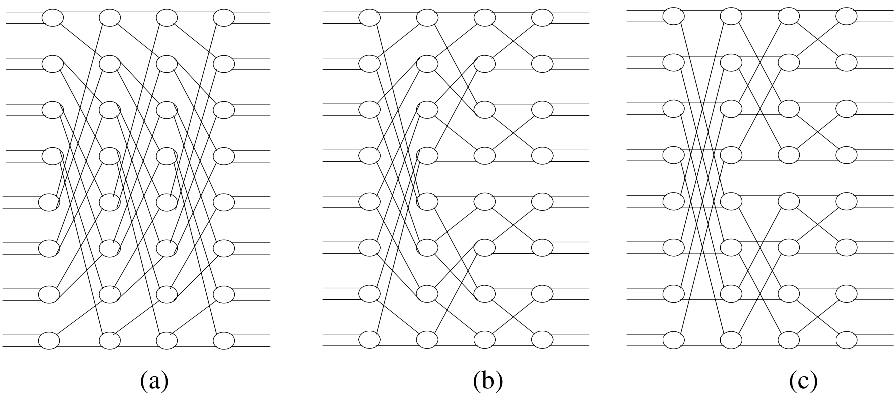


Fig. 1. Conventional layouts for (a) Shuffle (b) Baseline (c) Banyan networks

2 Crossover Minimization via Topological Embedding

In [9], a modular construction scheme was given to design directional-coupler-based switching networks with minimum number of crossovers, which is based on the permutation of stage node numberings without changing the conventional structure of the interconnection networks, such as shuffle, baseline and banyan. By the conventional drawing of an interconnection network, we comprehend that all input nodes in the plane are placed vertically on the left-side while all output nodes are placed also vertically on the right-side and links connecting internal stage nodes are drawn serially. In this paper, we adjust locations of the nodes in the interconnection network, so that all nodes can be placed in the plane provided that the adjacency relation of the links will remain the same as, in the shuffle, baseline or banyan interconnection networks. A free embedding of the interconnection network is called a topological embedding. If the resulting network is a multi-stage interconnection network, depending on the structure of the original network, it is called a banyan graph, a shuffle graph or a baseline graph. Our aim is to find suitable topologies in terms of crossover minimization without imposing restrictions on the location of input and output nodes. In order to find new layouts, we place the input nodes, denoted by the set $\{I_i\}$ vertically starting from the top to the bottom while we place the output nodes, denoted by the set $\{O_i\}$ horizontally starting from the left to the right, where $i=1,2,...,2^k$. We call cyclical representation of the network for such an embedded multi-stage interconnection network.

We note that, the exact minimum crossing number not only for the class of multipartite graphs but even for the complete bipartite graph $K_{m,n}$ includes several open problems [10],[11]. For example, R. Guy [12] showed the following theorem:

Theorem 1: The crossing number of $K_{m,n}$ satisfies the inequality:

$$cr(K_{m,n}) \le \left\lfloor \frac{m}{2} \right\rfloor \left\lfloor \frac{(m-1)}{2} \right\rfloor \left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{(n-1)}{2} \right\rfloor \quad (1)$$

where m+n is the number of nodes of $K_{m,n}$

In the above inequality, upper bound has only been proved when if m≤6 and *n* is arbitrary and then it is conjectured that inequality holds for all *m* and *n*.

We use the notation $N=2^k$ to denote the size of the network, where *k* is the number of node stages. Each input node has two inlets and each output node has two outlets. Any node in the network consists of a 2-by-2 switching element.

2.1 Cyclical Drawing of Shuffle Graphs

Shuffle networks are widely used in the sorting and in the interconnection of multi-processor computer systems. The topology of each link stage is the same, but it has more link crossovers than the other interconnection topologies. It can be verified that the shuffle graph shown in Figure 2 corresponds exactly to the conventional multi-stage shuffle interconnection network.

This can be realized by using the following input and output node numberings:

$$I_{2i-1} = \begin{cases} 2i-1 & i = 1,2,\ldots,2^{k-2} \\ 2i-2^{k-1} & i = 1+2^{k-2}, 2+2^{k-2}, \ldots, 2^{k-1} \end{cases} \tag{2}$$

$$I_{2i} = \begin{cases} 2^{(k-1)}+2i-1 & i = 1,2,\ldots,2^{k-2} \\ 2i & i = 2^{k-2}+1, 2^{k-2}+2, \ldots, 2^{k-1} \end{cases}$$

and $O_i = i$, $i = 1, 2, \ldots, 2^k$

Property 1: Consider the bipartite graph $G_{(2^{k-1})}$ shown in the Figure 3 which consists of node disjoint union of twisted 2^{k-1} cycles of length 4. Then the number of crossovers of $G_{(2^{k-1})}$ is given by

$$X\left(G_{(2^{k-1})}\right) = 2^{k-1}\left(2^{k+1}-3\right) \tag{3}$$

Property 2: Let G(N) is the cyclical embedding of the N-by-N multi-stage shuffle network. Then the total number of crossovers is given by

$$X(N) = 4\left(\sum_{i=0}^{k-4} 2^{k-4-i} X\left(G_{(2^i)}\right)\right) \tag{4}$$

$$\text{where} \quad X\left(G_{(2^i)}\right) = 2^{i-1}\left(2^{i+1}-3\right)$$

Property 3: The maximum number of crossovers between an inlet *s* and an outlet *d* in a k-stage cyclical shuffle graph G(N) is given by

$$X^{(k)}(s,d) = 2^k - 3k + 2 \tag{5}$$

where $1 \le s \le 2^k$ and $1 \le d \le 2^k$

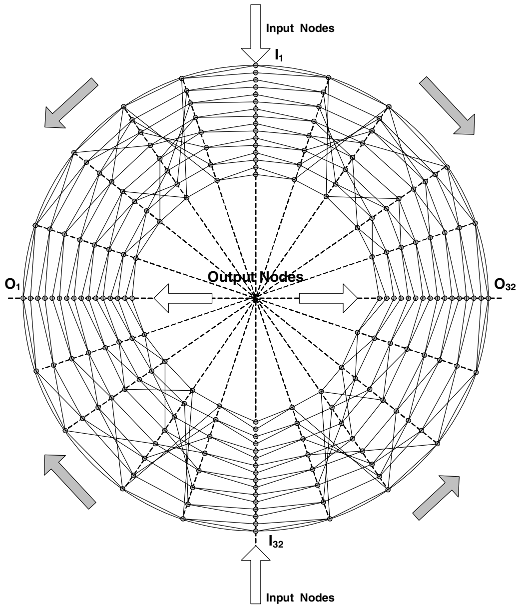


Fig. 2. A 6-stage 64-by-64 Cyclical Shuffle Graph

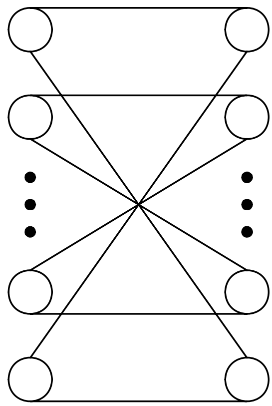


Fig. 3. The Bipartite Graph G_{2^k}

2.2 Cyclical Drawing of Baseline Graphs

Baseline networks have also applications in sorting and in many switching architectures. Cyclical embedding of baseline network is illustrated in Figure 4 for 5-stage, 32-by-32 baseline interconnection network. As it can be seen from the graph, it is decomposed into four identical sub-graphs where each sub-graph is the baseline network of size 8-by-8. Node numberings for general N, for input and output nodes are given by

$$I_i = i,\ i = 1,2,\ldots,2^k,$$

$$O_i = \begin{cases} \dfrac{N}{2} - i + 1 & i = 1,2,\ldots,2^{k-1} \\ \dfrac{3N}{2} - i + 1 & i = 2^{k-1}+1, 2^{k-1}+2,\ldots,2^k \end{cases} \tag{6}$$

Property 4: The total number of the crossovers in a k-stage cyclical baseline graph is given by

$$X(N) = 2^{2k-4} - (k-1)2^{k-2} \tag{7}$$

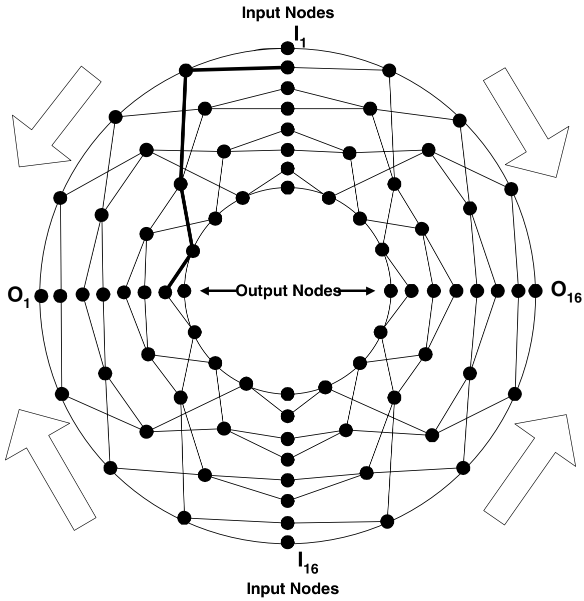


Fig. 4. A 5-stage 32-by-32 Cyclical Baseline Graph

Property 5: The maximum number of crossovers between an inlet *s* and an outlet *d* in a k-stage cyclical baseline graph is given by

$$X^{(k)}(s,d) = 2^{k-2} - k + 1 \tag{8}$$

where $1 \le s \le 2^k$ and $1 \le d \le 2^k$

2.3 Cyclical Drawing of Banyan Graphs

Banyan networks are widely used in the Fast Fourier transform in digital signal processing. Cyclical drawing of banyan network, illustrated in Figure 5, considerably reduces the number of crossovers. Input and output node numberings for these graphs are exactly the same as the mappings of the cyclical shuffle graphs (see Section 2.1).

Property 6: The total number of crossovers in a k-stage cyclical banyan graph is given by

$$X(N) = (3/8)2^{2k-2} - (2k-3)2^{k-2} \tag{9}$$

Property 7: The maximum number of crossovers between an inlet *s* and an outlet *d* in a k-stage cyclical banyan graph is given by

$$X^{(k)}(s,d) = 2^{k-2} - k + 1 \tag{10}$$

where $1 \le s \le 2^k$ and $1 \le d \le 2^k$

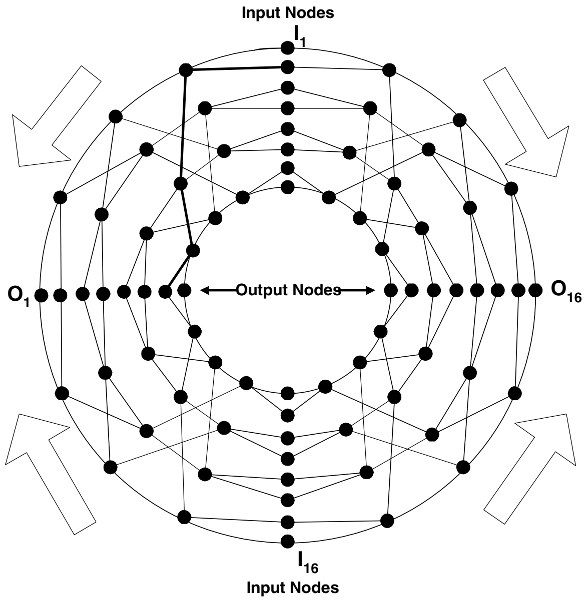


Fig. 5. A 5-stage 32-by-32 Cyclical Banyan Graph

3 Conclusion

The number of crossovers between the stage-links in the interconnection networks has an impact on the integrated optical realization, particularly when they are realized with the directional-coupler-based devices. In this paper, we embedded the conventional multi-stage interconnection network in the plane in such a way that the crossovers are minimized. We have summarized in Table 1, the total number of crossovers and the maximum number of crossovers between the inlet-outlet pairs for the conventional multi-stage shuffle, baseline and banyan networks and for the new

Table 1. Number of Crossovers in Conventional Drawing and Cyclical Drawing of Multi-stage Interconnection Networks

Number of stages	Conventional Drawing X(N), X(s,d)			Cyclical Drawing X(N), X(s,d)		
	Shuffle	Baseline	Banyan	Shuffle	Baseline	Banyan
2	1, (1)	1, (1)	1, (1)	0, (0)	0, (0)	0, (0)
3	12, (5)	8, (4)	10, (4)	0, (0)	0, (0)	0, (0)
4	84, (16)	44, (11)	60, (11)	4, (1)	4, (1)	4, (1)
5	480, (44)	208, (26)	296, (26)	48, (6)	32, (4)	40, (4)
6	2480, (111)	912, (57)	1328, (57)	304, (19)	176, (11)	240, (27)

layouts of the corresponding cyclical interconnection graphs. From Table 1, it can be seen that the cyclical baseline graphs have lower crossover numbers than the others. Moreover, the reduction of the number of crossovers with respect to the conventional drawings is in the order of four. Although we have not attempted to show whether the proposed interconnection layouts result in the minimum number of crossovers, for small values of *k* (the number of stages) the layouts suggest that the crossover numbers are the minimum possible.

Since many networks are based on shuffle, baseline and banyan networks' topologies, the results of this paper can be applied extensively to the study of crossover minimization for many other switching networks.

Arranging alternating fixed-size optical planes and electronic planes in a sandwich fashion can accomplish a package of the new interconnection network layouts to increase the capacity. Similar physical structures have already been implemented by using the three-dimensional optical interconnection concept [13-14].

References

1. J. Giglmayr: Planar Realization of All Optical Multiplayer Switching Fabrics. Proc. SPIE, Vol. 3288, (1998) 242-255
2. I. Cahit and J. Giglmayr: Recirculating Interconnection Networks: Directed Graph Representations, Routing, And Crossover Minimization. 1996 International Topical Meeting on Photonics in Switching, Vol. 1, Sendai, PWC8, Japan (1996)
3. C. Dhas, V. K. Konangl and M. Streetharan: Broadband Switching: Architectures, Protocols, Design and Analysis. IEEE Computer Society Press, (1991)
4. Lars Thylén, Gunnar Karlsson and Olle Nilsson: Switching Technologies for Future Guided Wave Optical Networks: Potentials and Limitations of Photonics and Electronics. IEEE Communications Magazine, Vol. 34, No. 2, (1996) 106-113
5. M. Nishio et al.: Photonic ATM Switch Using Vertical to Surface Transmission Electro-Photonic Devices (Vsteps). ISS XIV, B10.4, Yokohama, Japan (1992)
6. T. Sawano et al.: High Capacity Photonic Switching System. ISS XIV, B9.4, Yokohama, Japan (1992)
7. M. F. Yanik, et al.: High Contrast All-Optical Bistable Switching in Photonic Crystal Microcavities. Applied Physics Letter. Vol 83(14) (2003) 2739-2741
8. M. F. Yanik, et al.: All-Optical Transistor Action in Photonic Crystal Cross Waveguide Geometry, Optics Letters. (2003) 2506-2508
9. C.-T. Lea: Crossover Minimization in Directional-Coupler-Based Photonic Switching Systems. IEEE Trans. On Communications, 36(3), (1988) 355-363
10. F. Harary, Graph Theory, Addison-Wesley, Reading, Mass.(1972)
11. N. Hartsfield and G. Ringel: Pearls in Graph Theory. Academic Press, Boston (1990)
12. R. Guy: Crossing Number of Graphs. Graph Theory and Applications, Springer, New York (1972) 111-124
13. Guoqiang, L., Yuceturk, E., Dawei, H., and Esener, S.C.: Analysis of Free-Space Optical Interconnects for the Three-Dimensional Optoelectronic Stacked Processor. Optics Communications 202(4-6) (2002) 319-29
14. Guoqiang, L., Dawei, H., Yuceturk, E., Marand, P. J., Ozguz, V. H., Yue, L., Esener, S.C.: Three-Dimensional Optoelectronic Stacked Processor by use of Free-Space Optical Interconnection and Three-Dimensional VLSI Chip Stacks. Applied Optics 41(2) (2002) 348-360

Packet Scheduling Across Networks of Switches

Kevin Ross[1] and Nicholas Bambos[2]

[1] UCSC School of Engineering
kross@soe.ucsc.edu
[2] Stanford University
bambos@stanford.edu

Abstract. Recent developments in computer and communication networks require scheduling decisions to be made under increasingly complex system dynamics. We model and analyze the problem of packet transmissions through an arbitrary network of buffered queues, and provide a framework for describing routing and migration. This paper introduces an intuitive geometric description of stability for these networks and describes some simple algorithms which lead to maximal throughput. We show how coordination over sequential timeslots by algorithms such as those based on a round robin can provide considerable advantages over a randomized scheme.

1 Introduction

We consider the scheduling of service over generalized switch networks. In this paper we develop methodology to analyze networks of queues where service resources must be distributed over a network, and each queue may forward processed requests to another queue. Besides theoretical interest, this work has immediate applied impact in the design of multi-stage/multi-fabric switches (due to limited scalability of switching cores) as well as controlling interconnection networks.

Consider a general network of queues, with arbitrary interrelations between the queues. Packets, jobs or requests enter some queue in the network and remain there until they are served. Upon completion, packets are either forwarded to other queues or they depart the network. This model is a significant generalization to that presented in [8] where there is no forwarding or feedback allowed, and packets served in any queue immediately depart the network.

Several important results have been shown [6, 7, 4] on the stability of switches which can be modeled as interacting queues competing for service. For networks of switches, the potential for localized switching algorithms to lead to instability was shown in [2]. An early overview of queueing network theory is given in [9] and some recent work has included the analysis of greedy algorithms in [1] and using an adversarial fluid model approach in [5].

This paper proceeds as follows. In section 2, we describe in detail the model under consideration. In section 3 we discuss system stability and throughput, and in section 4 we introduce throughput maximizing algorithms with examples of their performance. Conclusions are outlined in section 5. Due to space limitations we have restricted the content to model formulation and simple algorithms.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 849–856, 2005.

2 The Network Model and Its Dynamics

In this section we develop the network model, using a sequence of definitions explained via carefully chosen examples and figures.

We consider a processing system which is a network comprised of Q first-in-first-out (FIFO) queues of infinite buffer capacity, indexed by $q \in \mathcal{Q} = \{1, 2, ..., Q\}$. Time is slotted and indexed by $t \in \{0, 1, 2, 3, ...\}$. Packets (jobs/tasks) may arrive at each queue in each time slot. Upon receiving service and departing from that queue, they may be routed to another queue, and then another, visiting several queues before eventually exiting the network.

We use the term *cell* to denote a unit of packet backlog in each queue. For simplicity, we assume that each packet can be 'broken' arbitrarily into cells or segments of cells, and in each time slot a number of cells can be processed at each queue then forwarded to another queue (or exit the network).

Vectors are used to encode the network backlog state, arrivals, and service in each time slot. Specifically, $X(t) = (X_1(t), X_2(t), ..., X_q(t), ..., X_Q(t))$ is the backlog state, where $X_q(t)$ is the integer number of cells in queue $q \in \mathcal{Q}$ at time t. The vector of external arrivals to the network is $A(t) = (A_1(t), A_2(t), ..., A_q(t), ..., A_Q(t))$ where $A_q(t)$ is the number of cells arriving to queue q at time t from outside the network (as opposed to being forwarded from other queues). The following is assumed for each $q \in \mathcal{Q}$

$$\lim_{t\to\infty} \frac{\sum_{s=0}^{t} A_q(s)}{t} = \rho_q \in [0, \infty) \tag{1}$$

that is, the long-term average external arrival load to each queue is well-defined, non-negative and finite; there is at least one queue with strictly positive external load $\rho_q > 0$, while several queues may have zero external load $\rho_{q'} = 0$. The long-term average load vector is $\rho = (\rho_1, \rho_2, ..., \rho_q, ..., \rho_Q)$. We do not assume any particular statistics that may generate the traffic traces, allowing for very general traffic loads to be applied.

At each time slot, the network may be set to one transfer mode. This is represented by a matrix $\mathbf{T}^m$ and a corresponding vector S^m chosen from the set of available modes $m \in \{1, 2, ..., M\}$.

Each $\mathbf{T}^m$ is a $Q \times (Q+1)$ matrix of transfer rates under mode m. It represents all of the cell transfers in that mode. In particular, for $q \neq Q+1$, $\mathbf{T}^m_{pq}$ is the number of cells sent from queue p to queue q in one timeslot when configuration mode m is used (with $\mathbf{T}^m_{pp} = 0$ for all p). For $q = Q+1$, $\mathbf{T}^m_{pq}$ is the number of cells served in queue p and then departing the system immediately under m.

For example if $Q = 3$ and the matrix $\mathbf{T}^* = \begin{bmatrix} 0 & 2 & 0 & 3 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ is used then two packets are forwarded from queue 1 to queue 2, three packets are served in queue 1 and then exit, and one cell exits from queue 3.

Corresponding to each matrix $\mathbf{T}^m$ are three service *vectors* S^m, S^{m+} and S^{m-}. These vectors reflect the total change in queue lengths for each queue in the system when mode m is selected. In particular, $S^{m+}_q = \sum_{p=1}^{Q+1} \mathbf{T}^m_{qp}$ is the total number of departures from queue q under mode m, $S^{m-}_q = \sum_{p=1}^{Q} \mathbf{T}^m_{pq}$ is the total number of arrivals to queue q generated by mode m, and $S^m = S^{m+} - S^{m-}$ is the vector of total change in workload (service) to the system under mode m. According to our example $\mathbf{T}^*$ above we have $S^{*+} = (5, 0, 1), S^{*-} = (0, 2, 0), S^* = (5, -2, 1)$.

At each timeslot, a mode m is selected from the available modes. If $S_q^{m+} > X_q$ for some q then more cells are scheduled to be served in queue q than are actually waiting.

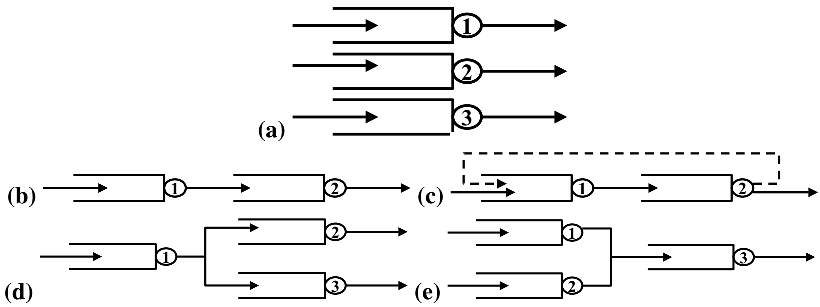


Fig. 1. Service modes under various queuing structures

(a) Parallel queues. This is the simple case of a parallel queue network topology with no cell routing interaction between queues; For example, the possible service transfer matrix $\mathbf{T} = \begin{bmatrix} 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ would serve two cells from queue 1 and one cell from queue 3 when applied in a slot.

(b) Tandem queues. The transfer matrix $\mathbf{T} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ would correspond to one cell being served in queue 1 and forwarded to queue 2.

(c) Queues with feedback. Cells served in one queue may be routed back to an upstream queue even if they have previously been processed there. On return to the upstream queue, the cell is either routed to the exact same queue or stored in a separate virtual tandem logical queue. Separate queues must be utilized when cells need to be distinguished according to the number of times they have already been processed there.

(d) Routing or splitting. There are two main scenarios covered by the model. In the first one, the mode selects which downstream queue to send each cell to. The configurations $\mathbf{T}^1 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$ and $\mathbf{T}^2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$ represent forwarding a cell from queue 1 to either queue 2 or 3 respectively. In the other scenario, queue 1 produces/spawns several cells and forwards to both queue 2 and queue 3. For example, $\mathbf{T}^3 = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$ would correspond to two cells served in queue 1 and then sending one to queue 2 and the other to queue 3 (similar to cell multicasting).

(e) Merging. In this network topology, cells may be forwarded from different queues to the same queue. For example, the configuration $\mathbf{T} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$ would allow both queues 1 and 2 to forward to queue 3 simultaneously

In this case the matrix $\mathbf{T}^m$ and vectors S^m must be adjusted to correspond to actual transitions. This is done through a careful notational change, differentiating between the selected mode at time t, labeled $m(t)$, and the actual transition and service levels $\mathbf{T}(t)$ and $S(t)$ (which are based on $\mathbf{T}^{m(t)}$ and $S^{m(t)}$ respectively). The updating of $\mathbf{T}(t)$ can be by some rule reflecting the priorities of waiting cells and maintains the property that the total workload forwarded under m is at most the number of cells waiting.

Assumption 1. At timeslot t, for a workload vector $X(t)$ and a selected service mode $m(t)$, the matrix $\mathbf{T}(t)$ of actual workload transfer at time t is found by some function $\mathbf{T}(t) = f(X(t), m(t))$ which satisfies $\mathbf{T}(t) \leq \mathbf{T}^{m(t)}$. Corresponding actual service vectors are $S_q^+(t) = \sum_p \mathbf{T}_{qp}(t) \leq X_q(t)$ and $S_q^-(t) = \sum_p \mathbf{T}_{pq}(t)$ for each $q \in \mathcal{Q}$.

One example of such a function would be $\mathbf{T}_{pq}(t) = \frac{X_q(t)}{S_q^{m(t)}} \mathbf{T}_{pq}^{m(t)}$, which sends cells in proportion to the scheduled transition matrix. Another example would be to reduce $\mathbf{T}_{pq}(t)$ for each q in order of priority.

Using our example matrix $\mathbf{T}$ from earlier, if the workload vector is $X(t) = (3, 5, 8)$ then five cells are scheduled to depart from queue 1 but only 3 are waiting. The function f may choose an alternative transfer matrix $\mathbf{T}(t) = \begin{bmatrix} 0 & 1 & 0 & 2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \leq \mathbf{T}^m = \begin{bmatrix} 0 & 2 & 0 & 3 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

Having defined carefully the terms in the workload evolution, the vectors representing workload and workload change follow the simple evolution equation:

$$X(t+1) = X(t) - S(t) + A(t) \tag{2}$$

Fig. 1 shows various network topology features, and the way that this model would describe each case. A general network topology would include multiple queues entangled via various tandem and feedback cell routing paths.

By extension of (2), in the long term

$$X(t+1) = X(0) + \sum_{s=0}^{t} A(s) - \sum_{s=0}^{t} S(s) \tag{3}$$

where $X(0)$ is the vector of initial backlog levels. The objective of this analysis is to develop algorithms for these systems to select $m(t)$ in each timeslot in a way that ensures that all cells are served and no backlog queue will grow uncontrollably.

For simplicity, all queues are considered to be *store-and-forward*. Current cell arrivals are registered *at the end* of the slot while cell service and departures *during* the slot. Therefore, it is not allowed for any cells to both arrive and depart in the same slot. Moreover, we assume *per-flow queueing* (or per-class) in the sense that if packets/cells are differentiated by class/flow they are queued up in separate (logical) queues in the system. Such class/flow differentiation may reflect distinct paths/routes of nodes that various packets/cells need to follow through the network or diverse service requirements they might have at the nodes.

3 Stability and Throughput

The vector backlog framework described here leads to an intuitive geometric understanding of stability. We say that an arrival rate is *stable* if there exists a sequence of configurations to match the arrival rate, and an algorithm is throughput-maximizing if it finds such a sequence for *any* such stable arrival rate.

We utilize the concept of **rate stability** in our throughput analysis of the system. In particular, we seek algorithms which ensure that the long-term cell departure rate from each queue is equal to the long-term arrival rate. Such algorithms must satisfy

$$\lim_{t\to\infty} \frac{\sum_{s=0}^{t} S_q(s)}{t} = \lim_{t\to\infty} \frac{\sum_{s=0}^{t} A_q(s)}{t} = \rho_q \tag{4}$$

for each $q \in \mathcal{Q}$, that is, there is cell *flow conservation* through the system.

In section 2 we described the transfer matrix $\mathbf{T}^m$ (or $\mathbf{T}(t)$) and the service vector S^m (or $S(t)$). For any set of modes available there is a finite set of possible vectors $S(t)$ which could be realized. We call this set $\mathcal{S}$. Note that the set $\{S^m\}_{m=1}^{M}$ is itself a subset of $\mathcal{S}$.

Definition 1. The **stability region** $\mathcal{R}$ of the switching system described is the set of all load vectors ρ for which rate stability in (4) is maintained under at least one feasible scheduling algorithm. The stability region can be expressed [3, 7] as

$$\mathcal{R} = \left\{ \rho \in \Re_+^Q : \rho \le \sum_{S\in\mathcal{S}} \phi_S S, \text{ for some } \phi_S \ge 0 \text{ with } \sum_{S\in\mathcal{S}} \phi_S = 1 \right\} \tag{5}$$

where $\mathcal{S}$ is the set of possible vectors that $S(t)$ can take on.

Intuitively speaking, a load vector ρ is in the stability region $\mathcal{R}$ if it is dominated (covered) by a convex combination of the service vectors $S \in \mathcal{S}$. This is illustrated in Fig. 2. Notice that some service vectors are themselves outside of the stability region due to their negative components. The stability region turns out to be the intersection of the convex hull of available configurations with the positive quadrant.

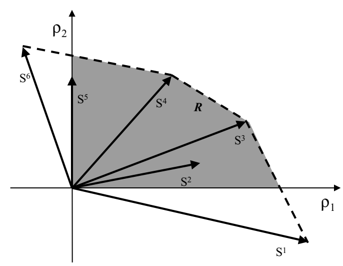


Fig. 2. The stability region of allowable rate vectors ρ

If $\rho \notin \mathcal{R}$ it is impossible to maintain rate stability and flow conservation in all queues no matter what feasible schedule we use; hence, at least one queue will suffer an outflow deficit compared to the cell inflow. That is shown by the following proposition.

Proposition 1 (Instability outside of $\mathcal{R}$). If $\rho \notin \mathcal{R}$ then it is always true that $\lim_{t\to\infty} \frac{\sum_{s=0}^{t} S(s)}{t} \neq \lim_{t\to\infty} \frac{\sum_{s=0}^{t} A(s)}{t} = \rho$ and the system is unstable for any scheduling algorithm.

Proof:
Proceeding by contradiction, assume that $\lim_{t\to\infty} \frac{\sum_{s=0}^{t} S(s)}{t} = \lim_{t\to\infty} \frac{\sum_{s=0}^{t} A(s)}{t} = \rho$. Now from (3),

$$X(t) = X(0) + \sum_{s=0}^{t} A(s) - \sum_{s=0}^{t} S(s) = \sum_{s=0}^{t} A(s) - \sum_{S\in\mathcal{S}} \sum_{s=0}^{t} S\mathbf{1}_{\{S(s)=S\}} \qquad (6)$$

Rearranging, and taking the limit as $t\to\infty$, this implies the relationship giving $\sum_{S\in\mathcal{S}} \lim_{t\to\infty} \frac{1}{t} \sum_{s=0}^{t} S\mathbf{1}_{\{S(s)=S\}} = \rho$. Setting $\phi_S = \lim_{t\to\infty} \frac{1}{t} \sum_{s=0}^{t} \mathbf{1}_{\{S(s)=S^{m'}\}}$ it follows that $\sum_{S\in\mathcal{S}} S\phi_S = \rho$ with $\sum_{S\in\mathcal{S}} \phi_S = 1$ which contradicts stability from (5). ∎

4 Throughput Maximizing Algorithms and Their Performance

We are interested in scheduling algorithms which maintain rate-stability as in (4) for all $\rho \in \mathcal{R}$. Here we present two such classes of algorithms and compare their structure and performance.

Randomized and round robin algorithms are simple algorithms which can achieve maximum throughput by using each configuration S the fraction ϕ_S of the total time. Consider the fraction of time corresponding to each particular mode. Let $\phi_m = \sum_{S|m} \phi_S$ be the fraction of time in mode m (under these stabilizing schemes), where $S|m$ is the set of possible $S(t)$ values that derive $S(t)$ from S^m.

Definition 2. Randomized algorithms use the policy in every timeslot t to

$$\boxed{\text{Select mode } m \text{ with probability } \phi_m}$$

Randomized algorithms are very simple to implement, requiring only a 'coin-flip' operation at each timeslot. Round robin algorithms, described below, use the same principle, but with a deterministic ordering of configurations instead of a randomized selection.

Definition 3. Round robin algorithms use the following for some fixed batch size T.

$$\boxed{\text{For each } m \text{, use mode } m \text{ for } \phi_m T \text{ timeslots}}$$

If $\phi_m T$ is not an integer number of timeslots, then rounding should be done in such a way to ensure that the long term average fraction of time spent using configuration S^m is ϕ_m.

From (5), it is easily seen that both randomized and round robin algorithms guarantee rate stability for the known arrival rate vector ρ.

We compare the performance of randomized algorithms and round robin algorithms in Fig. 3. Both algorithms were applied to a simple network with four queues and both tandem and parallel features. The backlog trace under each algorithm is shown when applied to the same randomly generated sequence of arrivals.

The round robin algorithm is seen to perform significantly better over time. This is due to the coordination of service, meaning it is less likely that the round robin algorithm

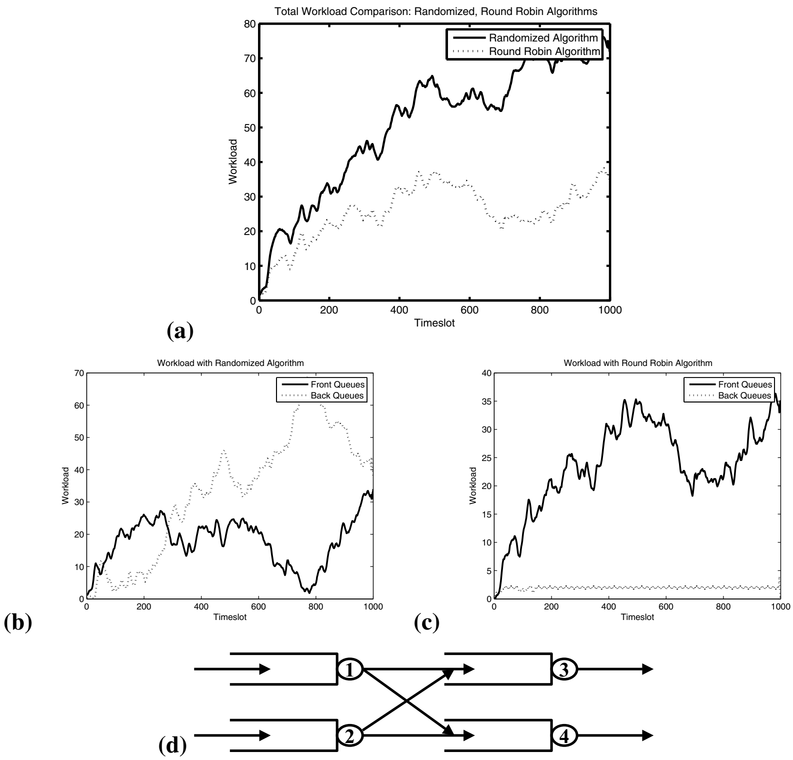


Fig. 3. Performance Comparison We compare the workload performance of randomized and round robin algorithms in the network illustrated in (d) above. At each timelot, the scheduler determines which one of the four queues to serve in each timeslot. Each of the 'front' queues can forward to either of the 'back' queues, and the back queues send the cells out of the network. This network allows six different service configurations on the four queues,

$$\mathbf{T}^1 = \begin{bmatrix} 0\,0\,1\,0\,0 \\ 0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0 \end{bmatrix}, \mathbf{T}^2 = \begin{bmatrix} 0\,0\,0\,1\,0 \\ 0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0 \end{bmatrix}, \mathbf{T}^3 = \begin{bmatrix} 0\,0\,0\,0\,0 \\ 0\,0\,1\,0\,0 \\ 0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0 \end{bmatrix}, \mathbf{T}^4 = \begin{bmatrix} 0\,0\,0\,0\,0 \\ 0\,0\,0\,1\,0 \\ 0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0 \end{bmatrix},$$

$$\mathbf{T}^5 = \begin{bmatrix} 0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,1 \\ 0\,0\,0\,0\,0 \end{bmatrix}, \text{and } \mathbf{T}^6 = \begin{bmatrix} 0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,1 \end{bmatrix}.$$

(a) The performance of randomized and round robin algorithms is compared. The total backlog over 1000 timeslots is recorded. The round robin performs better than the randomized algorithm due to its periodic sequence of forwarding to then serving the back queues. Both algorithms are known to be stable in the long term and apply the same proportion of service to each queue. However, coordinating that service more effectively gives the round robin algorithms an advantage.
(b) and **(c)**: These figures show the performance of each algorithm over the front and back queues. Observe that the round robin algorithm keeps tight control on the back queues. This is due to the logical ordering of service which coordinates arrivals in one slot with service in the next

will choose to serve an empty queue. For example, in each batch service to downstream queues follows forwarding from an upstream queue. The middle two plots in Fig. 3 show the performance of each algorithm separated into the front and back queues. The round robin algorithm is very efficient at serving the back queues since each cycle involves forwarding cells to the back and then serving them.

5 Conclusions and Further Research

We have introduced a general methodology for modeling networks of queues with distributed service. This methodology allows arbitrary combinations of queues to be connected and service applied to any combination of queues. Cells may be forwarded from one queue to others in the network, and feedback is also incorporated into this model.

In comparing randomized and round robin algorithms it is clear that great benefit can be gained by coordinating service over sequential timeslots. This is particularly useful intuition for developing more complex algorithms.

Both of the algorithm classes described here rely on prior knowledge of the long term arrival rates to each queue. We conjecture that throughput maximizing algorithms which do not rely on this information will also be found, and ongoing research in this area will be presented in the future.

References

1. Andrews, M., Awerbuch, B. Fernandez, A, Kleinberg, J., Leighton, T., and Liu, Z., (1996) Universal stability results for greedy contention-resolution progocols, Proc. IEEE Conf. Foundations Computer Science, 1996, pp380 - 389, 1996.
2. Andrews, M. and Zhang, L. (2003) Achiving stability in networks of input-queued switches. ACM/IEEE Trans. on Networking, Vol 11, No. 5, pp. 848-357, 2003.
3. Armony, M. and Bambos, N. (2003) Queueing Dynamics and Maximal Throughput Scheduling in Switched Processing Systems. Queueing Systems Vol. 44, No. 3, pp209, 2003.
4. Dai, J. G. and Prabhakar, P. (2000) The throughput of data switches with and without speedup. IEEE INFOCOM'00, pp. 556-564, 2000.
5. Gamarnik,D. (2000) Using fluid models to prove stability of adversarial queueing networks. IEEE Trans. on Automatic Control, Vol 45:4, pp 741-746, 2000.
6. McKeown, N., Mekkittikul, A., Anantharam, V., Walrand, J. (1999) Achieving 100% throughput in an input-queued switch. IEEE Transactions on Communications, 47(8):1260-1267, 1999.
7. Ross, K. and Bambos, N. (2004) Local Search Scheduling Algorithms for Maximal Throughput in Packet Switches, IEEE INFOCOM 2004.
8. Ross, K. and Bambos, N. (2004) Optimizing Quality of Service in Packet Switch Scheduling. Conference Proceedings, IEEE ICC, 2004.
9. Walrand, J. (1988) An Introduction to Queueing Networks. Prentice Hall, 1988.

New Round-Robin Scheduling Algorithm for Combined Input-Crosspoint Buffered Switch

Igor Radusinovic and Zoran Veljovic

Department of Electrical Engineering, University of Montenegro,
Cetinjski put bb, 81000 Podgorica, Montenegro
igorr@cg.ac.yu

Abstract. In this paper a high performance and simple scheduling algorithm for combined input-crosspoint crossbar switches, called exhaustive round-robin (ERR), is presented and analyzed. We propose using of this scheduling system for arbitration at inputs and crosspoints. If the virtual output queue (crosspoint buffer) becomes empty, the input (crosspoint) arbiter updates its pointer to the next location in a fixed order. Otherwise, the pointer remains at the current virtual output queue (crosspoint buffer). It is shown that this new solution achieves 100% throughput for several admissible traffic patterns, including uniform and unbalanced traffic, using only one-cell crosspoint buffers. ERR-ERR ensures service to the queues with high load using the exhaustive service and to the queues with low load using RR selection. Also, the performance of proposed CICQ under unbalanced traffic pattern increases and converges to output buffered switch performance as the crosspoint buffer increases. This scheduling algorithm is based only on the information about cell existing in virtual output queue (crosspoint buffer). Therefore, it requires much less hardware than the proposed algorithms. These results show the advantage of the ERR-ERR CICQ switch as a competitor for the next generation of high-performance packet switches.

1 Introduction

The amount of traffic carried over the Internet has been dramatically increasing with the tremendous popularity of World Wide Web (WWW). Because of that, high speed switches and routers have to be designed in a way to enable high throughputs (more than 1Tb/s) in a cost-effective manner. An attractive cell switching fabric is a non-blocking switch with input queuing due to easy hardware implementation. On the other side, it is well known that this switch is throughput limited [1], due to the head-of-line (HOL) blocking effect.

There are many techniques that have been suggested for HOL blocking reduction [2]. One of them is based on a simple buffering strategy where each input port maintains multiple queues (m) for a selected set of outputs. This queuing discipline is known as multi-input queuing (MIQ) [3]. If there is a separate queue for each of N outputs (m=N), MIQ becomes Virtual Output Queuing (VOQ). VOQ is technique where in each time slot, the iterative matching algorithm chooses a matching of input and output ports to schedule the switch matrix. Each input port is connected to at most

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 857–864, 2005.

one output port and vice versa. Such a VOQ solution enables complete elimination of HOL blocking. Various maximum weight/size matching, randomized (with linear complexity) or derandomized algorithms have been proposed for the VOQ architecture. In order to achieve not only the elimination of HOL blocking, but also high throughput in a cost effective manner, a NxN buffered crossbar switching fabric can be combined with VOQs as shown in Fig. 1.

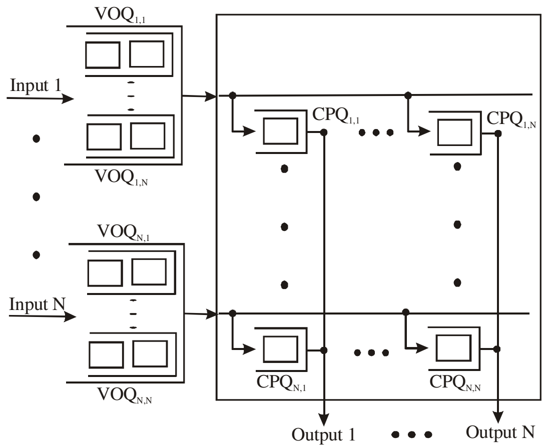


Fig. 1. A combined input-crosspoint buffered cell switch

This switch is known as the combined input-crosspoint queued (CICQ) switch [4]. The implementation of small Crosspoint Queues (CPQ) (with one or few cell length) allows multiple input ports to match with the same output port simultaneously, thus enabling all buffers (input and crosspoint) to operate at only twice of the input/output port rate. Using a credit flow control [4], the input and output (crosspoint) schedulers operate independently based on the states of the crosspoint buffer. It has be shown that these switches, with only one-cell crosspoint buffer and Round-Robin (RR) scheduling algorithm for arbitration at input and output ports, provide 100% throughput under uniform traffic [5]. But, it was shown that it is not true under admissible traffic patterns with nonuniform distributions [6].

Different scheduling algorithms as possible solutions for the input and crosspoint schedulers have been proposed. The oldest cell first (OCF) scheduling algorithm for each input and output scheduler was proposed in [4]. The longest queue First (LQF) scheduling algorithm for input scheduler and round-robin (RR) scheduling algorithm for output schedulers were implemented, as it is shown in [6]. In [7] we focused our research on these scheduling algorithms impact (input and crosspoint) on combined input-crosspoint buffered switch performance. Thus, our analyses enabled the extension of the previous results from [4] and [6], with novel performance evaluations under different traffic conditions. The most critical buffer first (MCBF) have been proposed in [8]. This solution has been proven to outperform solutions [5] and [6] for crosspoint buffer size of eight cells. Weight-based algorithms (LQF, OCF and MCBF) need to perform comparisons among all contenting queues (LQF,OCF) or internal buffers (MCBF), which number can be large. As the queueing structures tend to be flow-based, the number of comparisons is expected to increase. These algorithms may starve some queues to provide more service to the congested ones [9]. Many RR algorithms have been shown to provide fairness and implementation simplicity as no comparisons are needed among queues [9]. Recently, the round robin selection with adaptable-size frame (RR-AF) have been proposed in [10]. This RR-AF scheme and one-cell crosspoint buffers provides nearly 100% throughput under uniform and unbalanced traffic models. This RR based scheme does not need to compare status of other queues or weights. Each time that VOQ (or crosspoint buffer) is selected by the arbiter, the VOQ gets the right to forward a frame, where a frame is formed by one or more cells. The frame is adaptively determined (without intervention for the frame size selection) by the serviced and unserviced traffic.

Also, great effort was done in investigation of hardware design and burst stabilization protocol for the RR–RR combined input-crosspoint buffered switch in [11][12].

All previous switches suppose internally fixed cell switching. On the contrary, architecture, a chip layout and cost analysis, and a performance evaluation of a 300Gbps RR CICQ switch operating on variable-size have been presented in [13]. This architecture, using no speedup, has been shown to perform very close to ideal output queueing system and outperform practical unbuffered crossbar architectures with speedup less than 2x. Analytically description of the speed-up value needed for a packet-to-cell segmentation and new method of segmentation have been proposed in [14].

In this paper, we propose a new scheduling algorithm for CICQ that uses RR selection with exhaustive service. In each time slot, if VOQ (or crosspoint buffer) is selected corresponding cell will be transferred. After that, if the VOQ (crosspoint buffer) becomes empty, the input (crosspoint) arbiter updates its pointer to the next location in a fixed order. Otherwise, the pointer remains at the current VOQ (or crosspoint buffer). This is called the exhaustive service policy [15]. It is more feasible solution than RR-AF. We suppose internally fixed cell switching, but it is very easy to extend this concept on variable length packet switches. We show that this scheduling algorithm achieves nearly 100% throughput under a nonuniform traffic pattern, the unbalanced traffic model, with only one-cell crosspoint buffers. We prove, through simulations, that this scheduling algorithm offers a very high performance.

The paper is organized as follows. In Section 2, we present ERR scheduling algorithm. Section 3 contains a simulation study of the delay performance and stability of ERR-ERR CICQ switch under uniform and nonuniform traffic patterns. Finally, the conclusions and directions of future work are given in Section 4.

2 ERR-ERR CICQ Switch Operation

This section presents CICQ switch operation, as well as ERR scheduling algorithm.

Regarding Fig.1, we consider NxN buffered crossbar switch with a small buffer at each crosspoint. Every input port buffer has N VOQs, each of infinity length. The virtual output queue $VOQ_{i,j}$ holds cells arriving at input i addressed for output j

($i=1,2,\ldots N$, $j=1,2,\ldots N$). Particular crosspoint queue (CPQ), $CPQ_{i,j}$, is associated with one $VOQ_{i,j}$ in a way that cells stored in $VOQ_{i,j}$ will be sent to crosspoint queue $CPQ_{i,j}$,, each of c_p length. We assume that the time is slotted and the cells arrive at the switch at the beginning of a time slot. If $VOQ_{i,j}$ is selected, incoming cell will be stored in corresponding $CPQ_{i,j}$ immediately without waiting.

In every time slot the scheduling operation consists of independent crosspoint and input scheduling phases. A form of credit flow control is used between input and crosspoint schedulers. We choused credit-based flow control because the popular start/stop flow control requires an additional RTT window (plus a hysteresis safety margin) of buffer space per crosspoint [13]. Each CPQ and corresponding VOQ has an associated credit, used as a flag for the state of CPQ (1=not full, 0=full). It indicates to input i whether $CPQ_{i,j}$ has available place for a cell or not, as described in [5] and [10].

During the crosspoint (or input) scheduling phase, RR crosspoint (or input) schedulers at each output j (or input i), select one nonempty $CPQ_{i,j}$ (or nonempty $VOQ_{i,j}$ whose credit state is 1) based on the ring arbitration. The cell from selected $CPQ_{i,j}$ (or $VOQ_{i,j}$) departs switch (or $VOQ_{i,j}$ and enters $CPQ_{i,j}$). The selected $CPQ_{i,j}$ (or $VOQ_{i,j}$) gets right to forward until it becomes empty or exhausted. When any $CPQ_{i,j}$ becomes full crosspoint schedulers set its credit state to 0. It is simpler than RR scheduling algorithm with adaptable-size frame [10] because there is no need for a frame-size counter and a current service counter. Cell transmission from buffers (input or crosspoint) occurs at the end of a time slot.

3 Performance Analysis

In this section, we present a number of properties of the ERR-ERR architecture. We show that 100% throughput is achieved with a simple round-robin arbitration, with exhaustive selection policy for independent uniform traffic. A stability and delay performance were carried out. We do not take into account packet to cell segmentation and reassembly delay. The performance evaluation is done through two traffic models: bursty uniform and Bernoulli nonuniform (unbalanced).

In the bursty uniform traffic model, the traffic at each input is modeled as Interrupted Bernoulli Process. Output port addresses are uniformly distributed. Each of the inputs is described by the same ON-OFF model where both busy and idle periods are geometrically distributed. Cells of the same burst are destined for the same output (model of fragmented packet). We suppose that the average burst size equals b_s for the considered 32x32 switches. We simulate interval of 1000000 cell slots.

In the unbalance model, the cells arriving at each input at each time slot follow the same Bernoulli process with the same probability p (input load) of having a new cell. The incoming cells are distributed not-uniformly to all output ports [8], [10]. When w=0, the offered traffic is uniform. Otherwise, when w=1, traffic is completely unbalanced. This means that all the traffic of input ports is destined for output port j only, where $i=j$. We simulate interval of 100000 cell slots.

Despite one-cell CICQ with ERR scheduling algorithm for input scheduler and crosspoint schedulers, we studied the performance of five combined 32x32 input-crosspoint buffered crossbar switches: RR-RR (a CICQ fabric with a simple RR input/crosspoint arbitration and one-cell crosspoint), LQF-RR (a one-cell crosspoint CICQ switch using LQF scheduling algorithm for input scheduler and RR scheduling algorithm for crosspoint schedulers), OCF-OCF (a CICQ fabric using OCF scheduling algorithm for input scheduler and crosspoint schedulers), MCBF (a ten-cell crosspoint (due to stability reasons under unbalanced traffic [8]) CICQ fabric using the shortest internal buffer first at the input side and the longest internal buffer first at the output side) and OB (output buffered switch).

Similar to [6] we vary the average rates for a 2x2 switch with unbalanced loading, $\lambda_{i,j}$ of the connections and measure the maximum queue of each VOQ and its HOL cell delay for 10 consecutive intervals of 10000 cell slots. If the maximum value for a VOQ increases every interval or HOL cell delay reaches 1000, the switch is considered unstable.

Fig. 2 shows the instability regions for five scheduling algorithm under Bernoulli arrivals. Fig.4 illustrates that the RR-RR algorithm produces an instability region for admissible loads, but it doesn't intersect the $\lambda_{1,2} \leq ½$ region. LQF-RR, OCF-OCF, MBCF and ERR-ERR produce instability for inadmissible loads. We had to introduce additional criteria for HOL cell delay, because we obtained that ERR-ERR was stable in wide range of inadmissible region in sense of criteria with increase of maximum queue of each VOQ.

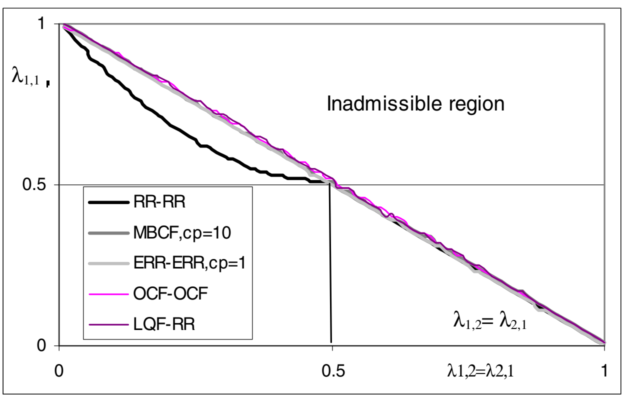


Fig. 2. Instability regions for unbalanced traffic on a 2x2 switch

Similar to [6] and [8], the input queues occupancies can serve to prove the stability of the scheduling algorithm. That is, if under a service policy X, we can show that $E(\|L(n)\|) < \infty$, then we can conclude that X is stable. $\|L(n)\|$ is defined as *l*-two norm vector representing the occupancy of the VOQs at time *n* and is defined as follows:

$$\|L(n)\| = \sqrt{VOQ_{11}(n)^2 + \ldots + VOQ_{1N}(n)^2 + \ldots + VOQ_{N1}(n)^2 + \ldots + VOQ_{NN}(n)^2} \tag{1}$$

Fig.3 shows simulation results of CICQ switches with ERR-ERR, LQF-RR, OCF-OCF and MBCF (with ten-cell crosspoint queue) under uniform traffic with Bernoulli arrivals (b_s=1) and bursts with average lengths of 8 (b_s=8) and 32 (b_s=32) cells. The simulation shows that the ERR-ERR scheduling algorithm has similar stability performance as other algorithms, despite the fact that it is the simplest one.

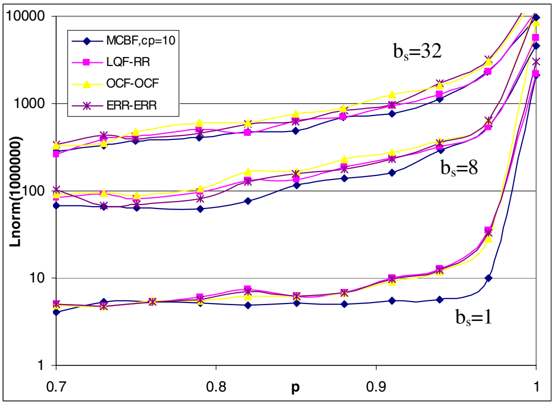


Fig. 3. The *l*-two norm vector under bursty uniform traffic for different average burst size b_s

Fig 4. illustrates that ERR-ERR with one-cell crosspoint buffer provides near 100% throughput irrespective of the unbalanced coefficient. We can see that ERR-ERR has throughput always higher than RR-RR (CIXB-1 [5]) and very close to OQ, LQF-RR, OCF-OCF and MBCF (c_p=10). This results in a feasible implementation of ERR-ERR CICQ switch. ERR-ERR ensures service to the queues with high load using the exhaustive service and to the queues with low load using RR selection.

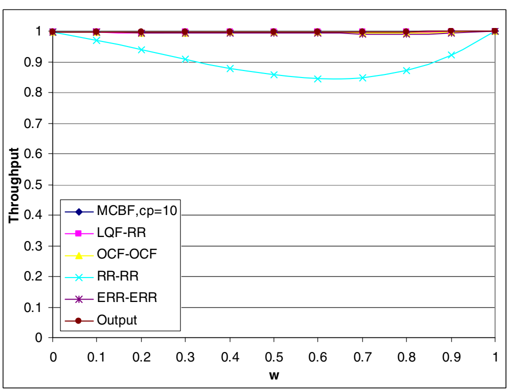


Fig. 4. Stability under nonuniform traffic

Fig.5 depicts the average delay performance under bursty uniform traffic with burst lengths equal 1 (b_s=1), 8 (b_s=8) and 32 (b_s=32). ERR-ERR exhibits the average delay performance between LQF-RR and OCF-OCF. MCBF with ten-cell crosspoint buffers has almost always (except in uniform case bs=1) greater average delay than other switches. For all considered CICQ switches increase of the average burst length means growth of the average cell delay.

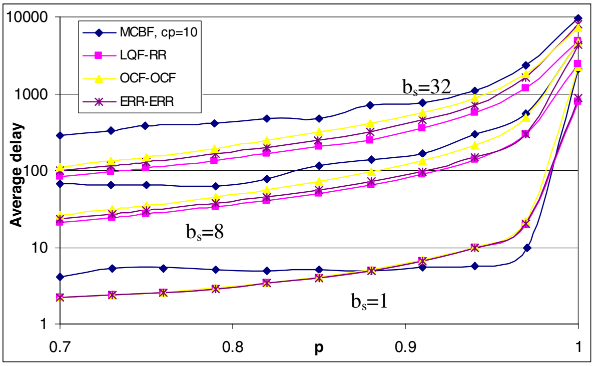


Fig. 5. Performance under bursty uniform traffic

4 Conclusions

We have proposed and investigated a new RR scheduling algorithm with exhaustive service to make RR based CICQ feasible and stable for the unstable region of ordinary RR-RR CICQ identified in [7]. We prove through simulations that this scheduling algorithm offers a high performance. We show that this scheduling algorithm achieves nearly 100% throughput under a nonuniform traffic pattern, the unbalanced traffic model, with only one-cell crosspoint buffers. The queues with large occupancy will have a higher opportunity to send cells. The queues with small occupancy will not be starved because of RR selection. The performance of ERR-ERR increases and converges to output buffered switch performance as the crosspoint buffer increases. Furthermore, our algorithm does not need to compare status or weights of other input/crosspoint queues as well as to take some counters (service and frame) into account. ERR-ERR exhibits the delay performance very close to LQF-RR and OCF-OCF and much better than MCBF with ten-cell crosspoint buffers under bursty uniform traffic no matter on burst lengths. In addition to high throughput and excellent delay performance, the switch provides timing relaxation that allows high-speed scheduling and scalability.

References

1. Karol, M., Hluchyj, M., Morgan, S.: Input versus output queuing on a space division switch, IEEE Trans. on Commun., Vol.35, (1987) 1347- 1356

2. Karol, M., Eng, K., Obara, H.: Improving the performance of input-queued ATM packet switches, Proceedings of IEEE INFOCOM ’92, (1992) 110-115
3. Tamir, Y., Frazier, G.: High performance multi-queue buffers for VLSI communications switches, Proceedings of 15th Ann. symp. on Comp. Arch., (1988) 342-354
4. Nebeshima, M.: Performance evaluation of a combined input- and crosspoint-queued switch, IEICE Trans. Commun., Vol. E83-B, No.3, (2000)
5. Rojas-Cessa, R., Oki, E., Jing, Z., Chao, H.J.: CIXB-1: Combined input one-cell-crosspoint buffered switch, Proceedings of IEEE WHPSR 2001, pp. 324-329.
6. Javidi, T., Magill, R., Hrabik, T.: A High-Throughput Scheduling Algorithm for a Buffered Crossbar Switch Fabric, Proceedings of IEEE ICC ’2001, 2001.
7. Radusinovic, I., Pejanovic, M., Petrovic, Z.: Impact of Scheduling Algorithms on Performances of Buffered Crossbar Switch fabrics, Proceedings of IEEE ICC ’2002, 2002.
8. Mhamdi L., Hamdi, M.: MCBF: A High-Performance Scheduling Algorithm for Buffered Crossbar Switches, IEEE Communications Letters, Vol.7, No.9, (2003) 451-453
9. McKeown, N.: Scheduling Algorithm for Input-queued cell switches, Ph.D. dissertation, Dept. EECS., Univ.California at Berkeley, Berkeley, CA, (1995)
10. Rojas-Cessa R., Oki, E.: Round-Robin Selection with Adaptable-Size Frame in a Combined Input-Crosspoint Buffered Switch, IEEE Communications Letters, Vol.7, No.11, (2003) 555-557.
11. Gunther, N.J., Christensen, K.J., Yoshiqoe, K.: Characterization of the Burst Stabilization Protocol for the RR/RR CICQ Switch, Proceedings of IEEE Conference on Local Computer networks, (2003) 260-269
12. Yoshiqoe, K., Christensen, K.J., Jacob, A.: The RR/RR CICQ Switch: Hardware design for 10-Gbps Link speed, Proceedings of IEEE Performance, Computing and Communications Conference, pp.481-485, April 2003.
13. Katevenis, M., Passas, G., Simos, D., Papaefstathiou, I. and Chrysos, N.: Variable Packet Size Buffered Crossbar (CICQ) switches, Proceedings of IEEE ICC’2004, 2004.
14. 14.Christensen, K.J., Yoshiqoe, K., Roginsky, A., Gunther, N.: Performance of Packet-to-Cell Segmentation Schemes in Input Buffered Packet Switches, Proceedings of IEEE ICC’2004, 2004.
15. Li, Y., Panwar, S., Chao, H.J.: Performance analysis of an exhaustive service dual round-robin scheduling algorithm, Proceedings of IEEE HPSR 2002, 2002.

Scheduling Algorithms for Input Queued Switches Using Local Search Technique

Yanfeng Zheng[1], Simin He[1], Shutao Sun[2], and Wen Gao[1]

[1] Institute of Computing Technology,
Chinese Academy of Sciences, Beijing 100080, China
{yfzheng, stsun, wgao}@jdl.ac.cn
[2] Graduate School of Chinese Academy of Sciences, Beijing 100039, China
stsun@jdl.ac.cn

Abstract. Input Queued switches have been very well studied in the recent past. The Maximum Weight Matching (MWM) algorithm is known to deliver 100% throughput under any admissible traffic. However, MWM is not practical for its high computational complexity $O(N^3)$. In this paper, we study a class of approximations to MWM from the point of view of local search. Firstly, we propose a greedy scheduling algorithm called *GSA*. It has the following features: (a) It is very simple to compute the weight of a neighbor matching. *GSA* only needs to compute the weight of two swapped edges instead of the weight of all the edges. (b) The computational complexity of *GSA* is $O(c\_max)$, where *c_max* denotes the maximum number of iterations. Hence we can adjust the value of *c_max* to achieve low computational complexity. Secondly, we observe that: (a) Local search is well suitable for parallel computing. (b) Each line card of high performance router has at least one processor. Based on the two important observations, we develop the second algorithm *PGSA*. Compared with *GSA*, *PGSA* significantly reduce the number of iterations. Simulation results show that *PGSA* with three iterations outperforms algorithms in [1] under different switch sizes.

1 Introduction

Input Queued (IQ) switch architecture has been very attractive due to its low memory bandwidth requirements compared to other known architectures. The well known head-of-line blocking on performance can be reduced or completely eliminated by virtual output queueing (VOQ) [2] at input line cards, and by controlling the switch operations with a scheduling algorithm.

The problem faced by scheduling algorithms with virtual output queues can be formalized as a maximum size or maximum weight matching on the bipartite graph in which nodes represent input and output ports and edges represent cells to be switched. The weight of edge connecting input i and output j is often chosen to be queue lengths or the ages of packets. We refer in this paper to queue lengths as edge weights.

The well known maximum weight matching (MWM) scheduling algorithm finds the matching (schedule) with maximum weight among all $N!$ matchings. MWM is

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 865–872, 2005.

known to deliver 100% throughput for any admissible traffic [3],[4],[5]. But it is too complicated for implementation. The best known implementations of MWM exhibit a computational complexity $O(N^3)$. This has led to several randomized approximations [1],[6] to MWM.

In [6], Tassiulas developed an adaptive scheduling method which can provide 100% throughput with low computational complexity. But it can induce a large average delay. In order to reduce the average delay, Giaccone et al. [1] proposed several algorithms, including *APSARA*, *LAURA*, and *SERENA*. For *APSARA*, it needs to compute all neighbors of the current matching, and then chooses a neighbor matching with the largest weight for next time slot. Because there are $N(N-1)/2$ neighbors of a matching, it is time-consuming to compute all the neighbors. Besides, much space is needed to store all the neighbors. Therefore *APSARA* is not practical when the switch size is very large. In order to reduce the number of neighbors, two variants of *APSARA* were proposed in [1]. One is *APSARA-L* and the other is *APSARA-K*. There are only N neighbors of a matching in *APSARA-L*. Simulation results in [1] show that *APSARA-L* performs quite competitively with *APSARA*. For *APSARA-K*, it chooses K neighbors at random and uses the heaviest of these. Note that K is much smaller than N. Obviously *APSARA-K* is more practical than *APSARA* and *APSARA-R*. But it was shown in [1] that *APSARA-K* does not perform as well as *APSARA-L*. Authors in [1] found that it is more important to remember the heavy edges than to remember the matching itself. This simple observation motivated the next algorithm *LAURA*, which iteratively augments the weight of the current matching by combining its heavy edges with the heavy edges of a randomly chosen matching. The computational complexity of *LAURA* is $O(I \cdot N \cdot \log_2 N + N)$, where I denotes the maximum number of iterations. *LAURA* seems to be impractical because of its high computational complexity. For *SERENA*, it uses the randomness in the arrivals process for finding good matchings to provide low average delays. However *SERENA* uses a complicated *MERGE* procedure to generate a heavy matching.

In this paper, we propose two algorithms for input queued switches using local search technique. Our first algorithm *GSA* tries to find a neighbor whose weight is larger than current matching. If such neighbor exists, the neighbor matching will be the new searching point. Notably, one nice feature of local search technique is that it can compute solutions in parallel on several processors. On the other hand, each line card of high-speed router has its own processor. Based on the above observations, we develop the second algorithm called *PGSA*. As a result of using parallel computing technique, *PGSA* significantly reduce the number of iterations compared with *GSA*. Note that *PGSA* with constant iterations works very well under different switch sizes. The simulation results *PGSA* with 3 iterations achieves very good delay performance compared with algorithms in [1].

The rest of the paper is organized as follows. In section **2**, we describe the input-queued switch crossbar architecture and some notation related to input-queued switches. In Section **3**-A, we describe the basic idea of local search. In Section **3**-B and Section **3**-C, *GSA* and *PGSA* algorithms are described respectively. In Section **4**, we measure the performance of *GSA* and *PGSA*. Finally, in Section **5** we conclude the paper.

2 Model and Notation

In this section we describe the model of an input-queued switch that is the main architecture studied in this paper.

Consider the $N \times N$ crossbar switch. We assume that the time is slotted and at each time slot, at most one packet can arrive at each input in one time slot. Fixed size packet is called a "cell". Cells arriving at input i and destined for output j are stored in a FIFO buffer called "virtual output queue"(*VOQ*), denote here by $VOQ_{i,j}$. Let $\lambda_{i,j}$ denote the arrival rate at $VOQ_{i\ j}$. The incoming traffic is called *admissible* if (a) $\sum_j \lambda_{i,j} < 1$, $\forall i$, and (b) $\sum_i \lambda_{i,j} < 1$, $\forall j$.

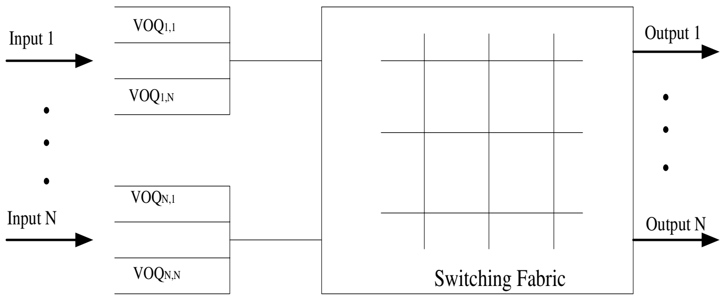


Fig. 1. Basic input-queued switch architecture

A matching[1] is represented by an $N \times N$ matrix $\mathbf{m} = [m_{i,j}]$ where if input i is connected to output j, we have $m_{i,j} = 1$, otherwise $m_{i,j} = 0$. The set of all possible matchings is denoted by $\mathcal{M}$. The matching matrix can be represented equivalently as a permutation $(\pi(1), \pi(2), \ldots, \pi(N))$ via the equation $\pi(i) = j$ iff $m_{i,j} = 1$. For instance, the matching

$$\mathbf{m} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

is equivalent to the permutation

$$(\pi(1), \pi(2), \pi(3)) = (1, 3, 2).$$

3 Scheduling Algorithms Using Local Search Technique

In this section, we first introduce the basic idea of local search, and then we describe the parallelized scheduling algorithm based on local search.

[1] Throughout this paper, we will use the words schedule, matching, permutation and solution interchangeably.

Recall that the goal of MWM is to find a matching with largest weight among the whole matching space. Hence it is natural to ask such a question if we can find a sub-optimal solution efficiently in a small part of matching space. This motivates us to study local optimization to find the answer.

A. Local Optimization

Local search, or local optimization, is a primitive form of continuous optimization in the discrete search space. Given a maximization problem with objective function f and feasible region R, a typical local search algorithm requires that, with each solution point $x_i \in R$, there is associated with a predefined neighborhood $N(x_i) \in R$. Given a current solution point $x_i \in R$, the set $N(x_i)$ is searched for a point x_{i+1} with $f(x_{i+1}) > f(x_i)$. If such a point exists, it becomes the new current solution point, and the process is iterated. Otherwise, x_i is retained as a local optimum with respect to the neighborhood structure.

B. A Greedy Scheduling Algorithm Based on Local Search (GSA)

Before we present *GSA* algorithm, we define the structure of neighborhood of a matching.

Definition 1: Given a matching m, let $\pi = (\pi(1), \pi(2),\ldots,\pi(N))$ be the corresponding permutation, where $\pi(i) = j$ iff $m_{ij} = 1$. A matching m' is said to be a neighbor of m iff there are exactly two inputs, say i_1 and i_2, such that m' connects input i_1 to output $\pi(i_2)$ and input i_2 to output $\pi(i_1)$. All other input-output pairs are the same under m and m'.

According to *Definition 1*, a neighbor of matching m can be generated by swapping two edges of m, leaving the other (N−2) edges unchanged. All the neighbors of matching m constitute the neighborhood N(m). For instance, matching m for a 3×3 switch and its three neighbors m'_1, m'_2, and m'_3 are given below

$m = (2,1,3)$ $m'_1 = (1,2,3)$ $m'_2 = (3,1,2)$ $m'_3 = (2,3,1)$.

Given a matching $m(t)$ for time slot t. *GSA* algorithm determines matching $m(t+1)$ for time slot $t+1$ as follows.

1) At the beginning of local search, *GSA* will choose a starting point (matching) for searching. As mentioned early in Section **1**, a heavy matching will continue to be heavy over a few time slots. Hence $m(t)$ is selected for the starting point.
2) In order to find a heavy matching for next time slot, it is necessary to do some iterations. During each iteration, *GSA* tries to find a neighbor whose weight is more than that of current matching. To ease the presentation, we assume the current matching is X_{best}. The initial value of X_{best} is set to the matching $m(t)$. Next step the algorithm will randomly generate a neighbor of X_{best}. Let *neighbor* denote such matching. If the weight of *neighbor* is more than that of X_{best}, then X_{best} is replaced with *neighbor*. The next iteration will begin with *neighbor*. Otherwise, the current matching X_{best} is still used for the next iteration.

Notably, for high speed switching systems, there is little time left for scheduling. Therefore it is critical to limit the number of iterations during one time slot. For *GSA*, we use variable *c_max* to control the maximum number of iterations. After finishing the limited iterations, X_{best} will point to a matching whose weight is no less than that of $m(t)$.

3) Finally, we obtain

$$m(t+1) = X_{best}$$

GSA has the following features: (a) During each iteration, only one neighbor of current matching is generated not the whole neighborhood. On the other hand, the neighbor of a matching is randomly selected from its neighborhood. (b) It is rather simple to compute the weight of neighbor matching. According to *Definition 1*, we only need to compute the weights of two newly swapped edges instead of the weights of all the edges. (c) The computation complexity of *GSA* is $O(c\_max)$. Hence we can adjust the value of *c_max* to achieve low time complexity of the algorithm.

C. Parallelized Greedy Scheduling Algorithm Based on Local Search

One nice feature of local search technique is that it can compute solutions in parallel on several processors. Fortunately, each line card of high speed router has its own processor. The two nice features motivate us to design a parallelized greedy scheduling algorithm called *PGSA*. We describe here how *PGSA* works.

Input: matching m(t) at time slot t.
Output: matching m(t+1) at time slot $t+1$.

Variable Description:

(a) Scheduler G_i is corresponding to the line card i, where $i = 1, 2, \ldots, N-1$.
(b) Scheduler G^* is corresponding to the line card N.

Algorithm Description:

1) Each scheduler G_i (i=1, 2, … , N−1) runs *GSA* algorithm in parallel. Note that matching $m(t)$ is selected as the starting searching point of each scheduler. On the other hand, each scheduler runs the same number of iterations which are dominated by variable *c_max*. After finishing iterations, each scheduler can achieve a matching. Let b_i be the matching obtained by scheduler G_i.

2) Scheduler G^* randomly chooses a matching R from the matching space $\mathcal{M}$. Note that this scheduler is essential to prove the stability of *PGSA*.

3) $m(t+1) = \underset{s\in\{b_1,b_2,\ldots,b_{N-1},R\}}{\arg\max}\ \mathrm{weight}(s)$.

That is, $m(t+1)$ is the matching which has the largest weight among the candidate set $\{b_1,b_2,\ldots,b_{N-1},R\}$.

Theorem 1: PGSA can achieve 100% throughput under Bernoulli i.i.d admissible traffic.

Proof: Because scheduler G^* randomly selects a matching from the matching space $\mathcal{M}$, *Theorem 1* can be proved by applying Proposition 1 in [6]. ■

4 Simulation Results

In this section, we study the average delay performance of *GSA* and *PGSA* compared with algorithms in [1]. Under different input traffic patterns mentioned below, we have studied the performance of the above algorithms with switch size 16, 32, 64, and 128. For each switch size, we got the similar comparing results. Due to space limitations, we only present a subset of simulations with switch size 32.

A. Input Traffic

We adopt same input traffic patterns used in [1]. That is, all inputs are equally loaded on a normalized scale. And $\rho \in (0, 1)$ denotes the normalized load. In convention, let $\lambda_{i,j}$ denote the arrival rate of Bernoulli i.i.d. Let $|j| = (j \bmod N)$. Two different input traffic patterns are described below.

a) *Uniform Traffic*: $\lambda_{i,j} = \rho / N, \forall i, j$.
b) *Diagonal Traffic*: $\lambda_{i,i} = 2\rho / 3$, $\lambda_{i,|i+1|} = \rho / 3\ \forall i$, and $\lambda_{i,j} = 0$ for all other i and j. Diagonal traffic is a very skewed loading. It is more difficult to schedule than uniform traffic.

B. Performance Measures

1) Performance Measures of *GSA*

In our experiments for *GSA*, we set *c_max* = *N*. Fig. 2 shows the average delay of *GSA* compared with *APSARA-L*, *LAURA* and *SERENA* under uniform traffic. It can be seen that the average delay of *GSA* is much smaller than that of *LAURA* and *APSARA-L*. Besides, *GSA* performs quite competitively with *SERENA*. Fig. 3 compares the average delay induced by *GSA*, *APSARA-L*, *LAURA*, and *SERENA* under the diagonal traffic. Clearly *GSA* outperforms *APSARA-L* and *LAURA*. Note that *SERENA* performs a little better than *GSA* under moderate load. But under heavy load (especially $\rho \geq 0.96$) *GSA* has better performance than *SERENA*.

2) Performance Measures of *PGSA*

To ease presentation, we use *PGSA-K* to represent *PGSA* algorithm running *K* iterations during one time slot. As shown in Fig. 4 and Fig. 5, the performance of *PGSA-N* is close to MWM under both uniform and diagonal traffic. Obviously, *PGSA-N* performs the best among *LAURA*, *APSARA-L*, and *SERENA*. With the increasing of line speed, little time is left to make arbitration. For the reason of scalability, it is necessary to reduce the number of iterations. Therefore we shall study the performance of *PGSA-K* where $K = 3$ and $K = \log_2 N$. According to simulation results illustrated above, it can be seen that *SERENA* outperforms *LAURA* and *APSARA-L*. Therefore we only compare the performance of *PGSA* with that of *SERENA* in the following simulations. Fig. 6 compares the average delay induced by *PGSA-K* ($K = 3, \log_2 N$), *SERENA* and MWM under diagonal traffic. We can see that *PGSA*-$\log_2 N$ and *PGSA*-3 have the similar performance with *SERENA* under moderate load. However, *PGSA*-$\log_2 N$ and *PGSA*-3 perform much better than *SERENA* under high load ($\rho > 0.9$). Especially when the traffic load is close to100%, the average delay of *SERENA* is almost 3 times of *PGSA*-3. Note that the number of iterations for *PGSA*-3 is constant which is independent of the switch size. It is meaningful to examine the performance of *PGSA*-3 under different switch sizes. Due to space limitations, we only present the simulation results of *PGSA*-3 compared with *SERENA* under 100% diagonal traffic in Fig. 7. It is shown that the performance of *PGSA*-3 is much better than that of *SERENA* under different switch sizes.

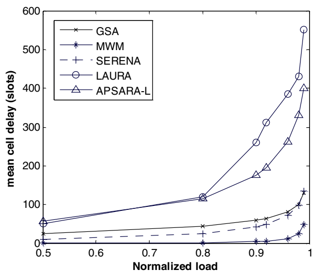


Fig. 2. The average delay of *GSA* under uniform traffic

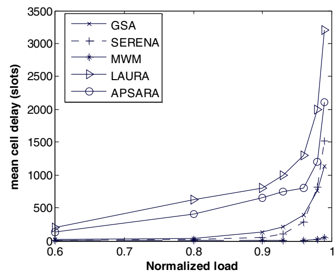


Fig. 3. The average delay of *GSA* under diagonal traffic

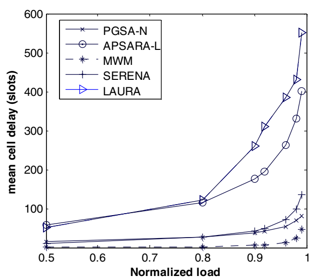


Fig. 4. The average delay of *PGSA-N* under uniform traffic

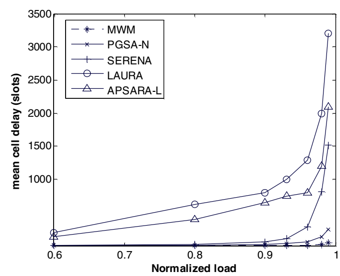


Fig. 5. The average delay of *PGSA-N* under diagonal traffic

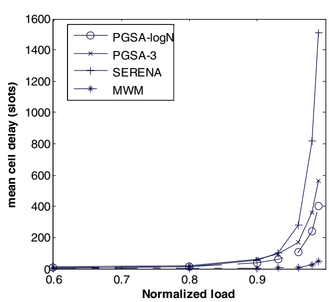


Fig. 6. The average delay of *PGSA*-$\log_2$ N, *PGSA*-3, *SERENA* and MWM

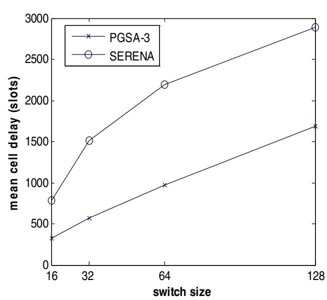


Fig. 7. Comparison on the performance of *PGSA*-3 and *SERENA* under 100% diagonal traffic load and different switch sizes

5 Conclusion

In this paper, we considered approximations to MWM using local search technique. Two algorithms were proposed in this paper. The first algorithm (*GSA*) reveals the basic idea of local search. The second algorithm (*PGSA*) makes full use of advantages of parallel computing. As a result of this, *PGSA* significantly reduces the number of iterations compared with *GSA* algorithm. On the other hand, *PGSA* has better scalability than *GSA*. *PGSA* works very well under different switch sizes.

References

1. P. Giaccone, B. Prabhakar, and D. Shah, "Towards simple, high-performance schedulers for high-aggregate bandwidth switches," in *Proc. IEEE INFOCOM '02*, New York, NY, June 2002.
2. Y. Tamir and G. Frazier, "High performance multi-queue buffers for VLSI communication switches," in *Proc. Of 15th Ann. Symp. on Comp. Arch.*, pp. 343-354,June 1988.
3. J.G. Dai and B. Prabhakar, "The throughput of data switches with and without speedup," in *Proc. IEEE INFOCOM '00*, vol. 2, Tel-Aviv, Israel, Mar. 2000, pp. 556-564.
4. L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," IEEE Trans. Automatic Control, vol. 37, no.12, Dec 1992, pp. 1936-1948.
5. N. Mckeown, V. Anantharam, and J. Walrand, "Achieving 100% throughput in an input-queued switch," in *Proc. IEEE INFOCOM'96*, pp. 296-302.
6. L. Tassiulas, "Linear complexity algorithms for maximum throughput in radio networks and input queued switches," in *Proc. IEEE INFOCOM'98*, vol. 2. New York, 1998, pp.533-539.

Multimedia Streaming in Home Environments

Manfred Weihs

Vienna University of Technology, Institute of Computer Technology,
Gusshausstraße 27-29/E384, 1040 Wien, Austria
weihs@ict.tuwien.ac.at

Abstract. Streaming of multimedia content, i. e. the transmission of audio and video data in real-time gains more and more importance due to the increasing availability of digital audio/video devices. In a typical home environment several networks can be used for that purpose. This paper gives an overview of the available technologies pointing out the features of IEEE 1394 and IP based networks. Furthermore it covers principal concepts for coupling of several streaming networks.

1 Introduction

Since the network bandwidth available in typical home environments has been increasing steadily, it is feasible to distribute multimedia content like video and audio digitally within the home using these networks. The topic of this work is streaming. That means multimedia data is transmitted at same (at least average) speed it should be presented at the sink device, the stream is transmitted in "real-time". In contrast to download-and-play the target device does not have to store a complete file, but it only buffers small parts in advance of presentation and discards them afterwards. Therefore it is usually not possible to replay a scene or to pause. The content can either be live (radio, television) or on demand (e. g. video clips).

One kind of network often installed in modern homes are field bus systems (for instance EIB, LonWorks etc.), which are used for home automation. They are used for control and monitoring purposes, offer rather low bandwidth and are therefore not well suited for multimedia streaming.

Another kind of network is IEEE 1394, which is integrated in many consumer electronic devices and becomes more and more important. This network technology is perfectly suited for multimedia streaming and will be evaluated more in detail later.

The most widespread class of networks installed in homes are IP based networks (mostly based on Ethernet). They are usually used to connect personal computers, but consumer electronic devices can be integrated into these networks as well. Most networks are still based on version 4 of the Internet Protocol (IP), but the successor (version 6) offers features that make it more suitable for streaming.

USB (Universal Serial Bus) might be mentioned as another kind of network. It could in principle be used for multimedia streaming (especially for audio, but

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 873–881, 2005.

as of version 2.0 also for video), but its main purpose is the connectivity between one personal computer and its peripherals. It is not used in a home networking concept, and therefore it is not covered here.

The aim of this paper is to give an overview of the available technologies and to outline the main features, advantages and disadvantages of various networks concerning streaming of multimedia content. It will also outline concepts of inter-networking between the various networks.

2 Requirements

A network that should be used for multimedia streaming has to fulfil several requirements. Some are absolutely essential, whereas others might be a bit relaxed.

2.1 Quality of Service

There are many different interpretations of the term "Quality of Service" (QoS) [1]. But the general meaning is, that in contrast to "best effort" services, some transmission parameters are guaranteed. Which parameters are guaranteed depends on the network type and the service it is used for. This guarantee usually involves some kind of reservation mechanism. A discussion of QoS parameters can be found in [2].

Parameters relevant concerning streaming of multimedia content are:

- Sufficient *bandwidth* must be available. Audio/video data streams often have a fixed data rate, that is known in advance (it is also possible that a stream adapts the data rate, if the available bandwidth changes).
- In case of interactive two way communication (e. g. video conferences) *latency*, i. e. the average transit delay, should be below a certain limit. However, for one way transmission latency is usually no problem.
- The *transit delay variation* (jitter) should be limited. The limit depends on the buffer used by the sink device to compensate the jitter.
- *Error rate* and *packet loss* should be low. Depending on the format of the data stream it might be more or less sensitive concerning data corruption.

If those parameters are guaranteed, Quality of Service is provided. If they are not guaranteed, streaming can be performed, but disruptions are to be expected in case these requirements are not met all the time.

2.2 Multicast

Since multimedia data usually has rather high bandwidth requirements, the available network bandwidth should be used economically. A major issue is that, if there is more than one sink for a stream within a network, it should be avoided to transmit the stream several times and therefore utilise a multiple of the necessary bandwidth. Some networks only support unicast transmission that is directed to one specific sink. In this case one connection for each sink is required leading to a waste of bandwidth.

For small networks in the home a possible solution could be the use of broadcast transmission, i. e. transmission targeting all nodes in the network. The major drawbacks thereof are the fact that also nodes not interested in the stream have to handle it and in case of networks consisting of more segments bandwidth is used on all segments, even on those without an interested sink device.

The ideal solution is multicast, transmission to a group of targets. This is usually done by use of special "group addresses". In this case the stream is transmitted exactly once on the network segments needed. In segmented networks the coupling units between them (usually routers or bridges) need some additional "intelligence" to figure out, on which segments the data has to be replicated.

Transmission that is targeted towards more than one device is usually not reliable. The data is transmitted once and there is no retransmission in case of errors. This exactly matches the requirements of streaming: An erroneous packet should not be retransmitted, because it would then probably be too late and therefore useless. The source should rather try to send the following packets in time than dealing with the erroneous packets. Timing is much more important than data integrity. The sinks must be able to cope with a certain (limited) amount of errors.

3 IEEE 1394

IEEE 1394 [3], also known as FireWire or i.Link, is a serial bus designed to connect up to 63 consumer electronic devices (like camcorders, CD players, satellite tuners etc.), but also personal computers and peripherals like hard disks, scanners etc. It supports up to 400 Mbit/s (the newer version 1394b supports up to 1600 Mbit/s and has architectural support for 3200 Mbit/s [4]), has an automatic configuration mechanism allowing hot plugging of devices and is very well suited for streaming of multimedia content [5].

It supports two different transmission modes: asynchronous and isochronous. The isochronous mode is designed to meet the requirements of streaming. There are 64 isochronous channels. Such channel can be regarded as a multicast group. In fact the packets are transmitted to each node on the network[1], but the hardware usually filters the packets and discards packets of isochronous channels that are not of interest. Bandwidth and channel can be reserved at the isochronous resource manager and are then guaranteed[2]. That means, that quality of service is provided. The source can send one isochronous packet every 125 µs. The allowed size of the packet corresponds to the reserved bandwidth. In case of errors no retransmission occurs, so timing is guaranteed, but not delivery.

[1] There are restrictions, if some nodes on the network do not support the transmission speed used by the source.

[2] The isochronous bandwidth is only guaranteed, if all nodes obey the rules of IEEE 1394 and reserve bandwidth and channel before they are used. Since this is not enforced by hardware or software, nodes violating these rules can compromise QoS.

Real-time transmission of audio and video data is well standardised by the series of IEC 61883. [6] defines general things like a common format for isochronous packets as well as mechanisms for establishing and breaking isochronous connections. This also includes timestamps, that are used for intra-media synchronisation and inter-media synchronisation [7]. The time base for these timestamps is the cycle time, which is synchronised between all nodes automatically by IEEE 1394. This is a very valuable feature of IEEE 1394: There is a clock that is automatically synchronised between all nodes and therefore does not drift.

Transmission of data in the DV (digital video) format used on video cassettes (and therefore typically provided by VCRs and cameras) is specified in [8, 9, 10]. An advantage of this format is the high quality and a low complexity of encoding and decoding, because it does not use the sophisticated inter-frame compression techniques known from MPEG-2. On the other hand it imposes rather high bandwidth requirements (about 25 Mbit/s for an SD format DV video stream, together with audio and other information it yields about 36 Mbit/s).

To distribute audio [11] gets involved. It is used by consumer electronic audio equipment like CD players, amplifiers, speaker systems etc. Usually audio is distributed as uncompressed audio samples at 44100 or 48000 Hz sample rate.

The transmission of MPEG-2 transport streams [12] over IEEE 1394 is specified in [13]. This allows the distribution of digital television over IEEE 1394, since both DVB and ATSC use MPEG-2 transport streams. MPEG-2 provides a good trade-off between quality and utilised bandwidth. A typical PAL TV program requires about 6 Mbit/s. [14] specifies how to distribute the content of a DVD over IEEE 1394 according to [11] and/or [13]. The alternate digital TV standard DirecTV system/DSS can be distributed according to [15]. So the ideal solution to distribute digital multimedia content is to use [11] for high quality audio data, whereas [13] should be used, if there is also video to distribute.

The working group P1394.1 is developing a standard for IEEE 1394 bridges [16]. That would allow to connect up to 1023 IEEE 1394 busses (each containing up to 63 nodes) to a network. An advantage of those networks is that traffic tends to be isolated on one bus. That includes isochronous traffic and bus resets. Bandwidth is therefore used very economically. If there is a listener on another bus than the corresponding talker (IEEE 1394.1 uses this terminology), there is a special protocol to setup the bridges in-between to forward isochronous packets.

It is also possible to transmit IP data on an IEEE 1394 bus. IPv4 over IEEE 1394 is specified in [17] and [18] specifies the transmission of IPv6 packets over IEEE 1394. Therefore it is possible to treat it as an ordinary IP based network and use the corresponding protocols described in the following section. However, this approach would not take advantage of the very special features of IEEE 1394 concerning real-time A/V streaming. The specifications of the series IEC 61883 are very well customised for IEEE 1394, so an IP based solution must have drawbacks (beside introducing additional protocol layers).

4 IP Based Networks

IP is a very widely used network protocol. The main reason is that it can be used on top of almost every lower layer networking system (Ethernet, token ring, IEEE 1394 etc.). In many homes (as well as offices) IP over Ethernet can be found, where usually still version 4 of IP is used. These networks were not designed to allow multimedia transmission and therefore have some weaknesses with regard to real-time data transmission (multicast is just an add-on, that is not widely supported, Quality of Service is very limited). Nevertheless these networks are very common. Since they are available almost everywhere and fulfil the basic requirements needed for audio and video transmission (although they are usually not guaranteed), it is very desirable to use these networks for that purpose. One problem concerning IP networks is, that transmission of audio and video is not very well standardised. There is a wide variety of data and transmission formats, open and proprietary ones, which are not compatible with each other.

In IP based networks many different protocols are commonly used for real-time transmission of multimedia. The most primitive one is HTTP. There are implementations that use HTTP to stream real-time audio data (e. g. icecast, shoutcast to mention two streaming server programs, that are freely available in the Internet). But besides some advantages (it is very simple and works well with proxies and firewalls) it also has many disadvantages. It does not support multicast, only unicast. Therefore, if more than one node is listening to the stream, several connections have to be established, that use the corresponding multiple of the necessary bandwidth. Furthermore HTTP is based on TCP, which is a reliable protocol, i. e. it does perform retransmission of packets, that were not transmitted correctly. This is completely inadequate for real-time transmission, because the retransmitted data will be late and therefore useless. In fact the use of HTTP is just an extension of progressive download (the presentation of a multimedia file starts while downloading), but it was never designed for real streaming.

A much more suitable protocol is RTP (Real-time Transport Protocol) as proposed by [19], which is usually based on UDP and in principle supports multicast as well as unicast, for transmission of multimedia data. It also includes timestamps, which are used for synchronisation purposes, but it has to deal with the fact that in IP networks there is no global time base which is synchronised automatically. RTSP (Real Time Streaming Protocol) can be used for setting up the transmission [20]. There are also proprietary protocols like MMS (Microsoft Media Server) or Real Delivery Transport (RDT) and others. They have similar features and are used by the tools of the corresponding companies (Microsoft, RealNetworks etc.). But the preferred protocols within the Internet are the open standards by the IETF.

Unfortunately there is no common format for audio and video data used in this context. Many different (open and proprietary) formats for audio and video exist and can be transmitted in an RTP stream. Concerning audio the most important open formats are uncompressed audio 16 bit, 20 bit or 24 bit [21, 22] and

MPEG compressed audio streams [23]. For video [23] specifies how to transmit MPEG video elementary streams. If audio and video are to be transmitted in one stream, MPEG-2 transport streams can be used as specified by [23], but the concept of RTP is to transport each media in a separate stream rather then using multiplexes. The most important proprietary formats are RealAudio and RealVideo.

There are mechanisms for QoS in IP networks, but they provide only soft guarantees and furthermore are not yet widely used in home environments. RSVP (Resource Reservation Protocol) is the standard protocol to reserve resources in IP networks [24, 25]. Multicast is supported in IP (on Ethernet IP multicasts are mapped to Ethernet multicasts), but if more then one subnet is involved, the routers in-between must be multicast-enabled.

IPv6, the successor of IP version 4, is not yet widely used, but has some advantages concerning multimedia streaming [26]. Multicast is integrated into IPv6, so every compliant implementation will support multicast. There are features (especially the flow labels should be mentioned) that make the provision of QoS for streams easier. The functionality of IGMP (Internet Group Management Protocol) is integrated into ICMPv6 (Internet Control Message Protocol) under the name "Multicast Listener Discovery" (MLD).

5 Interconnection

If there are several networks with streaming sources or sinks, it would be a benefit to the user to couple them [27]. This can easily be achieved, if two networks of the same type are to be coupled. IP networks can be coupled by using routers (if the lower layers are compatible, even bridges, switches or repeaters can be used, but that in fact would lead to one network). IEEE 1394 busses can can be coupled by the use IEEE 1394.1 bridges[3] [16]. These bridges are more powerful then bridges in the IP world, they do some kind of address translation, synchronisation of clocks a. s. o.

But the general problem is coupling of different kinds of network. There are three major approaches to achieve this, which are shown in Fig. 1. One approach is to set up a gateway that transforms the data in a way that on both networks multimedia data is carried in a format that is commonly used on that network. This means that for instance on an IEEE 1394 bus data is transmitted according to the series IEC 61883, whereas on IP data should be transported using RTP. The data format within that packets has to be transformed as well: While on IP audio is often compressed using RealAudio or some other (mostly proprietary) format, on IEEE 1394 it is usually transmitted as uncompressed PCM samples. The main advantage of this approach is, that conventional devices can be used as sources or sinks without the need for special adaptations. An example of such gateway for audio data between IP and IEEE 1394 is given in [7]. The gateway should of course also map the mechanisms for setting up a connection between

[3] These bridges are not yet available, there is a working group developing the standard.

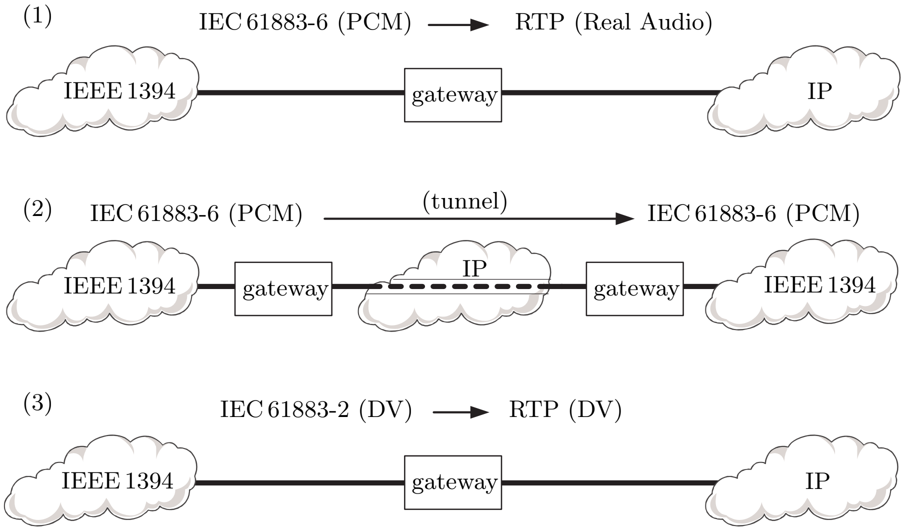


Fig. 1. Concepts for gateways between IEEE 1394 and IP based networks

the two networks, that means the use of the connection management procedures specified in [6] on IEEE 1394 and the use of RTSP on IP.

Another approach is to capsule the packets received on one network and retransmit that capsule on the other network. That would mean putting isochronous IEEE 1394 packets into IP packets (RTP packets or plain UDP packets). On the other hand RTP packets received from the IP network could be put into IEEE 1394 packets (asynchronous stream packets would be the most suitable type for that purpose). This is the easiest solution to couple the networks, but has the major drawback, that ordinary devices cannot handle them; IEEE 1394 enabled amplifiers can only handle isochronous packets containing data according to [11], but no RTP packets capsuled in IEEE 1394 packets. There is one scenario, in which such mechanism makes sense: If there is a second gateway which transforms the data back and transmits it again on a network of the same type as the source network, the intermediate network would just be used as a transit network. Devices on the intermediate network would not make use of the data, but on the sink network conventional devices can use the data. This mechanism could be referred to as "tunnelling". In essence that has the same effect as building a distributed IEEE 1394.1 bridge, where the two half-bridges are connected via an IP links. A similar approach is used for wireless IEEE 1394.1 bridges [28] with the difference that IEEE 1394 packets are not tunnelled through an IP network, but through IEEE 802.11 (so it is one layer below IP, on data link layer). It should be mentioned that a real IEEE 1394.1 compliant bridge also synchronises clocks (the IEEE 1394 cycle time) between both IEEE 1394 networks. In case of a tunnel consisting just of two separate gateways those clocks and therefore the isochronous cycles will drift. The other way round is even easier: Since IP can be carried in IEEE 1394 packets [17, 18], IP routers can be used to couple IP networks via IEEE 1394.

A third approach is a mixture between the above solutions. On each network the usual packet format is used, i. e. RTP on IP and CIP (common isochronous packet according to [6]) on IEEE 1394. But within those packets media data is coded as it was in the source network, which might be MPEG, DV, RealAudio etc. Such solution is described in [29]: Here DV (which is common on IEEE 1394, but unusual in IP networks) is transported in IP networks in form of RTP packets.

6 Conclusion

There are many types of networks within a typical home. Nevertheless with respect to streaming of multimedia data, there are two major systems: IEEE 1394 and IP based networks. In this article the main features of them were given as well as an overview over the standards involved into streaming on those networks. Since both kinds of network are used for streaming and have sources and sinks of the streams, there is a demand to couple them. A few concepts of gateways to couple the networks were given, where the most promising type of gateway is the one that ensures that on each network data is carried in a form that is commonly used on that network. Only that kind of gateway ensures that devices on both networks can make use of the multimedia data.

References

1. van Halteren, A., Franken, L., de Vries, D., Widya, I., Túquerres, G., Pouwelse, J., Copeland, P.: QoS architectures and mechanisms. Deliverable 3.1.1, AMIDST project (1999) Reference: AMIDST/WP3/N005/V09.
2. Vogel, A., Kerhervé, B., von Bochmann, G., Gecsei, J.: Distributed multimedia and QOS: A survey. IEEE MultiMedia **2** (1995) 10–19
3. IEEE Computer Society: IEEE standard for a high performance serial bus. Standard IEEE 1394-1995, IEEE Computer Society, New York, USA (1995)
4. IEEE Computer Society: IEEE standard for a high performance serial bus – amendment 2. Standard IEEE 1394b-2002, IEEE Computer Society, New York, USA (2002)
5. Anderson, D.: FireWire System Architecture: IEEE 1394a. Second edn. Addison-Wesley, Reading, MA, USA (1999)
6. IEC: Consumer audio/video equipment – digital interface – part 1: General. Standard IEC 61883-1, IEC, Geneva, Switzerland (1998)
7. Weihs, M., Ziehensack, M.: Convergence between IEEE 1394 and IP for real-time A/V transmission. In: Fieldbus Systems and their Applications 2003, IFAC, Elsevier (2003)
8. IEC: Consumer audio/video equipment – digital interface – part 2: SD-DVCR data transmission. Standard IEC 61883-2, IEC, Geneva, Switzerland (1998)
9. IEC: Consumer audio/video equipment – digital interface – part 3: HD-DVCR data transmission. Standard IEC 61883-3, IEC, Geneva, Switzerland (1998)

10. IEC: Consumer audio/video equipment – digital interface – part 5: SDL-DVCR data transmission. Standard IEC 61883-5, IEC, Geneva, Switzerland (1998)
11. IEC: Consumer audio/video equipment – digital interface – part 6: Audio and music data transmission protocol. Standard IEC 61883-6, IEC, Geneva, Switzerland (2002)
12. ISO/IEC: Information technology – generic coding of moving pictures and associated audio information: Systems. International Standard ISO/IEC 13818-1, Second edition, ISO/IEC (2000)
13. IEC: Consumer audio/video equipment – digital interface – part 4: MPEG2-TS data transmission. Standard IEC 61883-4, IEC, Geneva, Switzerland (1998)
14. DVD Forum: Guideline of Transmission and Control for DVD-Video/Audio through IEEE 1394 Bus, Version 1.0. DVD Forum. (2002)
15. IEC: Consumer audio/video equipment – digital interface – part 7: Transmission of rec. itu-r bo.1294 system b transport 1.0. Standard IEC 61883-7, IEC, Geneva, Switzerland (2001)
16. Johansson, P.: P1394.1 Draft standard for high performance serial bus bridges. P1394.1 Draft 3.0, IEEE Computer Society (2004)
17. Johansson, P.: IPv4 over IEEE 1394. RFC 2734, Internet Engineering Task Force (1999)
18. Fujisawa, K., Onoe, A.: Transmission of IPv6 packets over IEEE 1394 networks. RFC 3146, Internet Engineering Task Force (2001)
19. Schulzrinne, H., Casner, S., Frederick, R., Jacobson, V.: RTP: a transport protocol for real-time applications. RFC 1889, Internet Engineering Task Force (1996)
20. Schulzrinne, H., Rao, A., Lanphier, R.: Real time streaming protocol (RTSP). RFC 2326, Internet Engineering Task Force (1998)
21. Schulzrinne, H.: RTP profile for audio and video conferences with minimal control. RFC 1890, Internet Engineering Task Force (1996)
22. Kobayashi, K., Ogawa, A., Casner, S., Bormann, C.: RTP payload format for 12-bit DAT audio and 20- and 24-bit linear sampled audio. RFC 3190, Internet Engineering Task Force (2002)
23. Hoffman, D., Fernando, G., Goyal, V., Civanlar, M.R.: RTP payload format for MPEG1/MPEG2 video. RFC 2250, Internet Engineering Task Force (1998)
24. Braden, R., Zhang, L., Berson, S., Herzog, S., Jamin, S.: Resource reservation protocol (RSVP) – version 1 functional specification. RFC 2205, Internet Engineering Task Force (1997)
25. Wittmann, R., Zitterbart, M.: Multicast Communication - Protocols and Applications. Academic Press, San Diego, USA (2001)
26. Fahner, H., Feil, P., Zseby, T.: MBone - Aufbau und Einsatz von IP-Multicast-Netzen. dpunkt, Heidelberg, Germany (2001)
27. Skiczuk, P.: Network Protocol Architecture for Home Access Points. Ph.D. dissertation, Vienna University of Technology (2001)
28. Bard, S.: Wireless convergence of PC and consumer electronics in the e-home. Intel Technology Journal (2001)
29. Ogawa, A., Kobayashi, K., Sugiura, K., Nakamura, O., Murai, J.: Design and implementation of DV based video over RTP. In: Packet Video 2000. Number 31 in Proceedings, Forte Village Resort (Ca), Italy, University of Cagliari (2000)

Joint Buffer Management and Scheduling for Wireless Video Streaming

Günther Liebl[1], Hrvoje Jenkac[1], Thomas Stockhammer[2], and Christian Buchner[2]

[1] Lehrstuhl f. Nachrichtentechnik, TUM, D-80290 Munich, Germany
liebl@tum.de
[2] Nomor Research GmbH, D-83346 Bergen, Germany
stockhammer@nomor.de

Abstract. In this paper we revisit strategies for joint radio link buffer management and scheduling for wireless video streaming. Based on previous work [1], we search for an optimal combination of scheduler and drop strategy for different end–to–end streaming options. We will show that a performance gain vs. the two best drop strategies in [1], *ie* drop the HOL packet or drop the lowest priority packet starting from HOL, is possible: Provided that basic side-information on the video stream structure is available, a more sophisticated strategy removes packets from an HOL group of packets such that the temporal dependencies usually present in video streams are not violated. This advanced buffer management scheme yields significant improvements for almost all investigated scheduling algorithms and streaming options. In addition, we will demonstrate the importance of fairness among users when selecting a suitable scheduler.

1 Introduction

Optimization and adaptation of video streaming strategies to both wired and wireless clients, *eg* for High–Speed Downlink Packet Access (HSDPA), has become a challenging task. The heterogeneous network structure results in a number of conflicting issues: On the one hand, significant performance gains for video transmission over wireless channels can be achieved by appropriate adaptation. On the other hand, optimization of the media parameters or streaming server transmission strategies exclusively to wireless links will result in suboptimal performance for a wired transmission and vice versa. Hence, *cross-layer* design of the following components is required: Streaming server, wireless streaming client, media coding, intermediate buffering, channel resource allocation and scheduling, receiver buffering, admission control, media playout, error concealment, etc. Since the search for an optimal joint set of all parameters is usually not feasible, suboptimal solutions have to be considered, which yield sufficient performance gains by jointly optimizing a subset of the above parameters.

In this work we focus once again on strategies for joint radio link buffer management and scheduling for incoming IP–based multimedia streams at the

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 882–891, 2005.

radio link layer. Based on the wireless shared channel scenario in [1], we search for an optimal combination of scheduler and drop strategy for different end–to–end streaming options. In addition to the previously proposed drop strategies at the radio link buffers, we will investigate the gains achievable by incorporating basic side-information on the structure of the video stream. Our advanced drop strategy removes elements from an HOL group of packets such that the temporal dependencies usually present in video streams are not violated. We will assess the performance gain of this new scheme for an HSDPA scenario, and we will demonstrate the importance of fairness among users when selecting a scheduler.

2 Preliminaries for Wireless Video Streaming

2.1 End–to–End Streaming System

As stated in [1], assume that the media server stores a packet–stream, defined by a sequence of packets called *data units*, *ie* $\mathcal{P} = \mathcal{P}_1, \mathcal{P}_2, \ldots$. Each data unit $\mathcal{P}_n$ has a certain size r_n in bits and an assigned Decoding Time Stamp (DTS) $t_{\mathrm{DTS},n}$ indicating when this data unit must be decoded relative to $t_{\mathrm{DTS},1}$. After the server has received a request from a client it starts transmitting the first data unit $\mathcal{P}_1$ at time instant $t_{s,1}$ and continues with the following data units $\mathcal{P}_n$ at time instants $t_{s,n}$. Data unit $\mathcal{P}_n$ is completely received at the far–end at $t_{r,n}$ and the interval $\delta_n \triangleq t_{r,n} - t_{s,n}$ is called the channel delay (we assume that data units are either received correctly or lost in the network due to bit errors or congestion). The received data unit $\mathcal{P}_n$ is kept in the receiver buffer until it is forwarded to the video decoder at decoding time $t_{d,n}$. Without loss of generality we assume that $t_{\mathrm{DTS},1} = t_{d,1}$. Neglecting for now the session setup phase, the *initial delay* is defined as $\delta_{\mathrm{init}} \triangleq t_{d,1} - t_{s,1}$. Then, data units which fulfill $t_{s,n} + \delta_n \leq t_{\mathrm{DTS},n}$ can be decoded in time. Small variations in the channel delay can be compensated for by this receiver-side buffer, but long–term variances result in loss of data units. Several advanced streaming techniques have been proposed in the literature to cope with this “late-loss” [2]. However, most streaming systems available on the market do not apply any of them yet. Hence, we will not consider them here, but we note that their use is feasible in our framework and is currently investigated.

2.2 Source Abstraction for Streaming

According to [1], the video encoder $\mathcal{Q}_e$ maps the video signal $\mathbf{s} = \{s_1, \ldots, s_N\}$ onto a packet–stream $\mathcal{P} \triangleq \mathcal{Q}_e(\mathbf{s})$. We assume a one–to–one mapping between *source units* s_n, (*ie* video frames) and data units (*ie* packets). Encoding and decoding of s_n with a specific video coder $\mathcal{Q}$ results in a reconstruction quality $Q_n \triangleq q(s_n, \mathcal{Q}(s_n))$, where $q(s, \hat{s})$ measures the rewards/costs when representing s by $\hat{s}$. We restrict ourselves in the following to the Peak Signal–to–Noise Ratio (PSNR), as it is accepted as a good measure to estimate video performance. According to [3], the result of the encoding is a set of data units for the presentation which can be represented as a directed acyclic graph. If such a set is received by the client, only those data units whose ancestors have all been also received can be decoded. In case of a lost data unit, the corresponding source unit is represented by the timely–nearest received and reconstructed source unit (*ie* a direct or indirect ancestor). If there is no preceding source unit, *eg* I–frames, the lost source unit is concealed with a standard representation, *eg* a grey image. In case of consecutive data unit loss, the concealment is applied recursively. The concealment quality $\tilde{Q}_{n,\nu}(i)$, if s_n is represented with s_i, is defined as $\tilde{Q}_n(i) \triangleq q(s_n, \mathcal{Q}(s_i))$. We express the importance of each data unit $\mathcal{P}_n$ as the increase in quality at the receiver if s_n is correctly decoded, *ie*

$$I_n \triangleq \frac{1}{N}\left(Q_n - \tilde{Q}_n(c(n)) + \sum_{\substack{i=n+1 \\ n \vdash i}}^{N} \left[\tilde{Q}_i(n) - \tilde{Q}_i(c(n))\right]\right), \tag{1}$$

with $c(n)$ the number of the concealing source unit for s_n, and $n \vdash i$ indicating that i depends on n. Additionally, $\tilde{Q}_n(0)$ indicates concealment with a standard representation. The overall quality for a sequence of length N is then

$$\overline{Q} = \frac{1}{N}\sum_{n=1}^{N} Q_n = Q_0 + \sum_{n=1}^{N} I_n, \tag{2}$$

with Q_0 the minimum quality, if all frames are presented as grey. Hence, quality is *incrementally additive* w.r.t. to the partial order in the dependency graph.

2.3 Streaming Parameters and Performance Criteria

The video decoder might experience the absence of certain data units $\mathcal{P}_n$ due to loss related to bit errors or congestion in the network ($\delta_n = \infty$), late-loss at the client ($\delta_n > t_{\text{DTS},n} - t_{s,n}$), or the server not even having attempted to transmit the unit. Whereas the former two reasons mainly depend on the channel, the latter can be viewed as temporal scalability and a simple means for offline rate control and is not used here. Another important parameter in our end–to–end streaming system is the initial delay at the client. On the one hand, this value should be kept as low as possible to avoid annoying startup delay to the end user. On the other hand, longer initial delay can compensate for larger variations in the channel delay and reduce late-loss. Since we have ruled out more advanced streaming strategies, the single-user performance can be determined using a sequence of channel delays $\delta = \{\delta_1, \ldots, \delta_N\}$ for each data unit and a predefined initial delay δ_{init} as ($\mathbf{1}\{A\}$ equals 1 if A is true and 0 otherwise)

$$Q(\delta, \delta_{\text{init}}) = Q_0 + \sum_{n=1}^{N} I_n \mathbf{1}\{\delta_n \leq \delta_{\text{init}}\} \prod_{\substack{m=1 \\ m \prec n}}^{n-1} \mathbf{1}\{\delta_m \leq \delta_{\text{init}}\}. \tag{3}$$

2.4 Streaming in a Wireless Multiuser Environment

We assume that M users in the serving area of a base station in a mobile system have requested to stream multimedia data from one or several streaming servers.

We assume that the core network is over–provisioned such that congestion is not an issue on the backbone. The streaming server forwards the packets directly into the *radio link buffers*, where packets are kept until they are transmitted over a shared wireless link to the media clients. A scheduler then decides which users can access the wireless system resources bandwidth and transmit power, and a resource allocation unit integrated in the scheduler assigns these resources appropriately. Obviously, for the same resources available different users can transmit a different amount of data, *eg* a user close to the base station can use a coding and modulation scheme which achieves a higher bit–rate than one at the boundary of the serving area. In general, the performance of the streaming system should significantly depend on many parameters such as the buffer management, the scheduling algorithm, the resource allocation, the bandwidth and power share, the number of users, etc. As done in [1], we have concentrated on the first two aspects in our investigations. Hence, the performance criterion for the single user system has been extended by averaging (3) over all users, *ie*

$$Q(M, \delta_{\text{init}}) = \frac{1}{M} \sum_{m=1}^{M} Q(\delta_m, \delta_{\text{init}}). \tag{4}$$

3 Scheduling and Buffer Management Strategies

3.1 Scheduling

Several scheduling algorithms for wireless multiuser systems have already been proposed in the literature [5, 6, 7, 8]. We will briefly characterize them here:

1. **Basic scheduling strategies:** Well–known wireless and fixed network scheduling algorithms include, for example, the *Round Robin* scheduler, which serves users cyclically without taking into account any additional information.
2. **Channel–State Dependent Schedulers:** The simplest, but also most appealing idea for wireless shared channels – in contrast to fixed network schedulers – is the exploitation of the channel state of individual users. Obviously, if the flow of the user with the best receiving conditions is selected at any time instant, the overall system throughput is maximized. This scheduler is therefore referred to as *Maximum Throughput* (MT) scheduler and may be the most appropriate if throughput is the measure of interest. However, as users with bad receiving conditions are blocked, some fairness is often required in the system. For example, the *Proportional–Fair* policy schedules the user with the currently highest ratio of actual to average throughput.
3. **Queue–Dependent Schedulers:** The previously presented algorithms do not consider the buffer fullness at the entrance of the wireless system except that flows without any data to be transmitted are excluded from the scheduling process. Queue–dependent schedulers take into account this information, *eg* the *Maximum Queue* (MQ) scheduler selects the flow whose Head-Of-Line packet in the queue currently has the largest waiting time.

4. **Hybrid Scheduling Policies:** It might be beneficial to take into account both criteria, the channel state information and the queue information, in the scheduling algorithm. In [8] hybrid algorithms have been proposed under the acronyms *Modified Largest Weighted Delay First* (MLWDF) and *Exponential Rule*, which yield the most promising results among the standard solutions.

3.2 Radio Link Buffer Management

For better insight into the problem, we assume that a single radio link buffer can store N data units, independent of their size. If the radio link buffers are not emptied fast enough because the channel is too bad and/or too many streams are competing for the common resources, the wireless system approaches or even exceeds its capacity. When the buffer fill level of individual streams approaches the buffer size N, data units in the queue have to be dropped. We will discuss several possible buffer management strategies in the following:

1. **Infinite Buffer Size (IBS):** Each radio link buffer has infinite buffer size $\mathsf{N} = \infty$, which guarantees that the entire stream can be stored in this buffer. No packets are dropped resulting in only delayed data units at the client.
2. **Drop New Arrivals (DNA):** Only N packets are stored in the radio link buffer. In case of a full queue, new packets are dropped, which is the standard procedure applied in a variety of elements in a wired network, *eg* routers.
3. **Drop Random Packet (DRP):** Similar to DNA, but instead of dropping the newly arrived packet we randomly pick a packet in the queue to be dropped. This strategy is somewhat uncommon, but we have included it here since all other possibilities are only specific deterministic variants of it.
4. **Drop HOL Packet (DHP):** Same as DNA, but here we drop the Head–Of–Line (HOL) packet, *ie* the packet which resides longest in the buffer. This is motivated by the fact that streaming media packets usually have a deadline associated with them. Hence, to avoid inefficient use of channel resources for packets that are subject to late-loss at the client anyway, we drop the packet with highest probability of deadline violation.
5. **Drop Priority Based (DPB):** Assuming that each data unit has assigned a priority information, we drop the one with the lowest priority which resides longest in the buffer. Our motivation here is the fact that sophisticated media codecs, like H.264/AVC [4], provide options to indicate the importance of elements in a media stream on a very coarse scale. Hence, those packets are removed first which do not affect the end-to-end quality significantly.
6. **Drop Dependency Based (DDB):** We propose the following new strategy to avoid the major drawback of both DHP and DPB: Starting from the HOL of the buffer when dropping medium to high priority packets is suboptimal due to the temporal dependency within the media stream. For example, if I- or P-frames in a Group–of–Pictures (GOP) are removed, all the remaining frames in the GOP cannot be reconstructed at the media decoder. Thus, leaving them in the buffer and transmitting them leads to inefficient utilization of the scarce wireless resources. Provided that basic side-information on the

structure of the media stream is available to the buffer management (*eg* the GOP structure and its relation to packet priorities), an optimized strategy operates on the HOL group of packets with interdependencies: While all low priority packets can be deleted starting from the beginning of the HOL group, any medium or high priority packets should be first removed from the end of the HOL group to avoid broken dependencies. Since the structure of the media stream is usually fixed during one session, the buffer management only has to determine this information once during the setup procedure, which we believe to be feasible at least in future releases of wireless systems.

3.3 Streaming Server Rate Control

As in [1], we only consider two basic streaming server rate control models:

1. **Timestamp–Based Streaming:** In case of TBS the data units $\mathcal{P}_n$ are transmitted exactly at sending time $t_{s,n}$. If the radio link buffer is emptied faster than new data units arrive, then it possibly underruns. In this case, this flow is temporarily excluded from the scheduling.
2. **Ahead–of–Time Streaming:** In case of ATS, the streaming server is notified that the radio link buffer can still accept packets. Hence, the streaming server forwards data units to the radio link buffer even before their nominal sending time $t_{s,n}$ such that the radio link buffer never underruns and all flows are always considered by the scheduler. However, the streaming server eventually has to forward a single data unit no later than at $t_{s,n}$ regardless of the fill level notification. Thus, a drop strategy at the radio link buffer is still required. Note that this mode requires pre-recorded streams at the server and a sufficiently large decoder buffer at the media client.

4 Experimental Results

4.1 Definition of Test Scenario

We have used the same HSDPA scenario as in [1] for our performance evaluations, which consists of one serving base station and 8 tiers of interfering base stations. We omit the detailed system parameter settings here and refer the interested reader to the above publication. A total of $M = 10$ randomly distributed users are attached to the serving base station and each of them has requested a streaming service for the same H.264/AVC–coded QCIF sequence of length $N = 2698$ frames (looped six times) as in [1], with $QP = 28$, 30 fps, and no rate control. The GOP structure is IBBPBBP..., with an I–frame distance of 1 s. The PSNR results in $Q(N) = 36.98$ dB, and the average bit–rate is 178.5 kbit/s. We have evaluated the performance in terms of average PSNR $Q(M = 10, \delta_{\text{init}})$ vs. initial delay δ_{init} for selected parameter settings. All presented results with limited buffer size assume a restriction of $N = 30$ packets (*ie* 1 second of video).

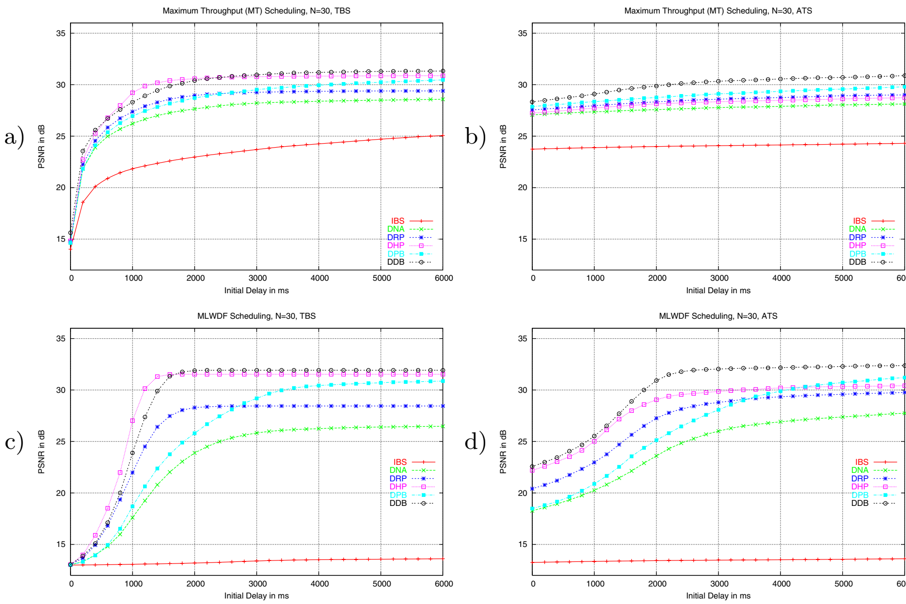


Fig. 1. Average PSNR versus initial delay for different buffer management strategies under MT (a,b) and MLWDF (c,d) scheduling

4.2 Buffer Management Performance for MT and MLWDF Policy

In Fig. 1a,b we compare all the different buffer management strategies under MT scheduling for both TBS and ATS. Regardless of the drop and streaming strategy, the system performance increases with larger initial delay as the probability of late-loss decreases. As the system is overloaded (about 30%), in case of IBS the fullness of the radio link buffers increases over the length of the streams. Since no dropping is performed, users with bad channel conditions experience significant HOL blocking and excessive initial delay at the media client is required for sufficient performance for both TBS and ATS. Nevertheless, due to the maximum throughput scheduler at least some users, namely those close to the base station, are served with good quality, but the worse users experience too high channel delays for this setup. This fact is especially evident in Fig. 1b, where due to the persistent occupation of the channel by the good users (who always have data to be sent in case of ATS) the quality for very short initial playout delay is already quite high, but then only increases very slowly. Hence, for improving the overall system performance it is beneficial to drop data units already at the radio link buffers (irrespective of the strategy) to reduce the excess load at the air interface and convert late-loss into controlled packet removals, *ie* achieve an in–time delivery of a temporally scaled version of the video stream. The simplest strategy DNA shows some gain, but is worse than dropping packets randomly. The by far best performance for both TBS and ATS is obtained by applying our newly proposed DDB algorithm: While DHP and DPB intersect at relatively

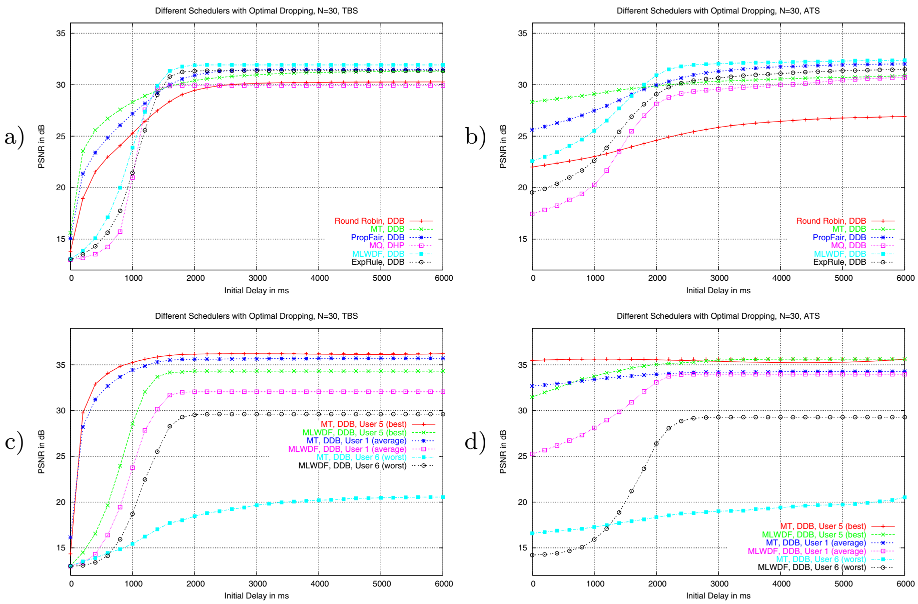


Fig. 2. a),b) Average PSNR versus initial delay for different schedulers and optimal buffer management strategy. c),d) Results for the best, worst, and average user

large initial delay, the curve for DDB shows good performance over the whole range of delay values, especially in case of ATS. Furthermore, an interesting observation can be made for all drop strategies and TBS: If the radio link buffer size is larger than the initial delay, an almost linear gain can be achieved by increasing the latter. However, soon after the initial delay matches the radio link buffer size, the PSNR curve runs into saturation, since the majority of packets is now dropped due to the finite buffer size and not due to deadline expiration.

If we evaluate the performance of our drop strategies for a hybrid scheduler, like MLWDF, which has been designed to account for some fair trade-off between channel and queue state, we observe that IBS should never be used: Both Fig. 1c,d show disastrous consequences for the average PSNR. Hence, considering the HOL delay in the scheduling metric leads to large performance degradations for all users, if the amount of HOL blocking is not limited. On the other hand, if the radio link buffer size is limited, applying our new DDB algorithm either yields close to optimum (TBS) or strictly optimum (ATS) performance.

4.3 Scheduler Comparison and Fairness

Figures 2a,b contain the average PSNR for six different combinations of scheduler and optimal drop strategy under TBS and ATS. Albeit for the MQ scheduler with TBS, optimal buffer management is achieved by our new DDB algorithm. The question which scheduler to use, however, is not as simple, but largely depends on the initial playout delay: For very short values, the MT scheduler performs best, while all other schedulers with queue-based metrics are not very efficient. For reasonable initial playout delays larger than one second, the MLWDF scheduler would be the better choice. However, the type of scheduler has to be chosen upon system startup without knowing individual initial playout delays. Therefore, the fairness among the users in the system has to be considered by looking at the PSNR of the best, worst, and average user depicted in Fig. 2c,d for both MT and MLWDF. While MT favors both the best and the medium user by suppressing the bad user significantly, MLWDF tries to achieve a trade-off between maximum throughput of the system and fairness. Hence, the best and medium user quality is slightly reduced, while the system tries to supply the bad user with sufficient quality as well. This is especially evident for ATS, where the gain of the bad user is large compared to the decrease of the other two. Obviously, for reasonable initial playout delays, this support of the bad users also results in an increase in average quality over all users, since more of them contribute to it.

5 Conclusion

In this paper we have revisited strategies for joint radio link buffer management and scheduling for video streaming over wireless shared channels. As a straightforward extension to previous work we have proposed a more sophisticated drop strategy at the radio link buffer that incorporates side-information on the temporal dependency structure in typical video streams. Albeit for one combination of scheduler and streaming mode, our newly proposed algorithm provides optimal performance over the whole range of (unknown) initial playout delays. Since the side-information only has to be determined once during the setup phase, we consider it to be feasible within future releases of wireless systems like HSDPA. Furthermore, our investigations have gained us some valuable insight into the applicability of certain types of schedulers for wireless video streaming: In particular, we showed that the combination of a queue-state dependent scheduler with infinite radio link buffer size leads to disastrous results for all users. However, DDB combined with a hybrid scheduler seems to yield a good trade-off between average quality of all users and fairness among individual users. Since including side-information on the video stream in the buffer management has proven to be successful, making part of it also available to the scheduler is the subject of ongoing research at our institute. First results already show that priority– and/or deadline–based scheduling policies yield significant improvements.

References

1. G. Liebl, H. Jenkac, T. Stockhammer, and C. Buchner, "Radio Link Buffer Management and Scheduling for Video Streaming over Wireless Shared Channels," in Proc. *Packet Video Workshop 2004*, Irvine, CA, USA, Dec. 2004.

2. B. Girod, M. Kalman, Y. J. Liang, and R. Zhang, “Advances in video channel–adaptive streaming,” in Proc. *IEEE Int. Conf. on Image Processing*, Rochester(NY), USA, Sept. 2002.
3. P. A. Chou and Z. Miao, “Rate-distortion optimized streaming of packetized media,” 2001, submitted. http://research.microsoft.com/ pachou.
4. *Advanced Video Coding for Generic Audiovisual Services*, ITU–T and ISO/IEC JTC 1, 2003.
5. M. Andrews, K. Kumaran, K. Ramanan, A. Stolyar, and P. Whiting, “Providing Quality of Service over a Shared Wireless Link,” *IEEE Communications Magazine*, vol. 39, pp. 150–154, February 2001.
6. H. Fattah and C. Leung, “An Overview of Scheduling Algorithms in Wireless Multimedia Networks,” *IEEE Trans. on Wireless Communications*, October 2002.
7. S. H. Kang and A. Zakhor, “Packet Scheduling Algorithm for Wireless Video Streaming,” in Proc. *Packet Video Workshop 2002*, Pittsburgh, USA, April 2002.
8. S. Shakkottai and A. L. Stolyar, “Scheduling Algorithms for a Mixture of Real-Time and Non-Real-Time Data in HDR,” in *Proceedings of the 17th International Teletraffic Congress (ITC-17)*, Salvador, Brazil, September 2001.
9. R. S. Tupelly, J. Zhang, and E. K. Chong, “Opportunistic scheduling for streaming video in wireless networks,” in *Proc. 37th Annual Conference on Information Science and Systems*, Baltimore, MD, USA, Mar. 2003.

Performance Analysis of a Video Streaming Buffer

Dieter Fiems, Stijn De Vuyst, and Herwig Bruneel

SMACS Research Group, Department TELIN, Ghent University,
St-Pietersnieuwstraat 41, B-9000 Gent, Belgium

Abstract. In this contribution, we investigate the performance of the output buffer of a video streaming server. We assume that the server encodes the video stream in a scalable way. When the output buffer gets congested, one may choose to drop the transmission of some of the layers in the packets, thus reducing the packet transmission time and expediting the restoration of the buffer size to normal levels. A discrete-time finite capacity queueing model with buffer size dependent transmission times is constructed and analysed using a probability generating functions approach. We conclude with some numerical examples.

1 Introduction

Scalable video coding is capable to cope with bandwidth fluctuations in the network, see a.o. [1, 2, 3] and the references therein. A scalable video stream consists of a base layer and one or more enhancement layers that may or may not be sent depending on the available bandwidth in the network. In this way, the video quality can be reduced gracefully if this is required by the network conditions.

In this contribution, we focus on the performance evaluation of an output buffer of a video streaming server with scalable coding capabilities. The packets generated by the scalable video codec contain the information of the base layer and of all of the enhancement layers when they arrive in the buffer. However, if there are a lot of packets waiting in the buffer, it may be beneficial not to transmit some of the layers in the packets. By dropping one or more of the upper enhancement layers, the transmission time of the packets is reduced and the packets are temporarily transmitted at a faster rate. As such, the scalable structure of the video packets allows us to prevent packet loss and to maintain an uninterrupted flow of video packets to the end user. The quality of the received video stream dynamically adapts to the congestion level in the output buffer in a controlled way.

In our performance model, the scalability of the video stream is captured by means of buffer length dependent transmission times. That is, the transmission time of a packet (and therefore also the video quality) depends on the number of packets in the buffer when the transmission of this packet starts. Using a probability generating function approach, we investigate the characteristics of

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 892–900, 2005.

the busy and idle period. The latter allow us to determine performance measures such as the packet loss ratio and the mean packet transmission time.

Different authors have investigated the characteristics of the busy period of queueing systems before, see a.o. [4, 5, 6, 7, 8] and the references therein. Both Zwart [4] and Baltrūnas et al. [8] focus on the tail behaviour of busy periods of $GI/G/1$ type queues. Ohta [6] and Agarwal [5] consider finite capacity queues. The former considers the discrete-time M/G/1/N queue whereas the latter considers the $GI/M/1$ queue. The busy-period for multi-server queueing systems is investigated by Artalejo and Lopez-Herrero [7]. None of these authors however allow buffer size dependence of the transmission times.

2 Performance Model

We consider a discrete-time system. That is, we assume that time is divided into fixed length intervals called slots. During the consecutive slots, multimedia frames – say packets – arrive at a finite capacity buffer and are transmitted on a first-in-first-out basis. The buffer can store up to N packets simultaneously, additional packets are dropped. The numbers of arrivals during the consecutive slots constitute a series of independent and identically distributed random variables with common probability mass function $a(n)$ ($n \geq 0$) and with corresponding probability generating function $A(z)$.

Transmission of packets is synchronised with respect to slot boundaries. This implies that arriving packets cannot start transmission during their arrival slot. To capture the dynamic adaptation of the multimedia quality to the congestion level, we assume that the transmission times of the consecutive packets depend on the number of packets present in the buffer when the transmission starts. Or, equivalently, the transmission times of the consecutive packets depend on the number of free buffer spaces when the transmission starts. Therefore, it is clear that the transmission times of the consecutive packets are not independent. However, we assume that the transmission times of those packets for which there are n free buffer spaces at the start of their transmission are independent and identically distributed. The probability mass function of the transmission times (expressed in terms of slots) given that there are n free buffer spaces when the transmission starts is given by $s(k|n)$ ($k > 0$). The corresponding conditional probability generating function is denoted by $S(z|n)$ for $0 \leq n \leq N-1$. Note that there is at least one packet – or $N-1$ free buffer spaces – in the buffer when a packet's transmission starts.

3 Idle and Busy Period Analysis

The system under consideration is busy during a slot whenever a packet is being transmitted during this slot and is idle whenever this is not the case. As such, the system alternates between being idle and being busy. Due to the independence in the arrival process, one easily shows that the consecutive idle and busy periods constitute two series of independent and identically distributed random variables. As the system remains idle as long as there are no arrivals (with probability $a(0)$), one may further observe that the length of the idle period is geometrically distributed. The common probability generating function $I(z)$ of the idle periods is given by,

$$I(z) = \frac{(1 - a(0))z}{1 - a(0)z} . \tag{1}$$

We now determine the joint probability generating function of the length of a busy period and the number of packets that are transmitted during such a busy period. We first focus on sub-busy periods, sometimes referred to as fundamental periods.

Let the sub-busy period C of a (tagged) packet denote the number of slots between the beginning of the slot where this packet starts transmission and the beginning of the slot where the buffer contains one packet less than at the start of this transmission for the first time. As such, the sub-busy period includes the transmission time S of the packet and a number of sub-busy periods C_k equal to the number of packet arrivals during the transmission time S (excluding packets that are lost). Analogously, the number of packet transmissions Q during a tagged packet's sub-busy period equals one, augmented with the number of packet transmissions Q_k during the sub-busy periods C_k. That is,

$$C = S + \sum_{k=1}^{A_S} C_k , \quad Q = 1 + \sum_{k=1}^{A_S} Q_k , \tag{2}$$

where A_S denotes the number of arrivals (excluding lost packets) during the tagged packet's transmission time,

$$A_S = \min \left(F, \sum_{k=1}^{S} A_k \right) . \tag{3}$$

In the former expression, F denotes the number of free buffer spaces at the start of a packet's sub-busy period and A_k denotes the number of packet arrivals during the kth transmission slot of this packet. The former equations are illustrated in figure 1.

The random variables C and Q depend on the number of free buffer spaces F at the start of the sub-busy period. Also, the consecutive (C_k, Q_k) $(k = 1 \ldots A_S)$ given the number of free buffer spaces F_k $(k = 1 \ldots A_S)$ at the start of the sub-busy periods constitute a series of independent random variables. Let $\Omega(x, z|n)$ denote the joint probability generating of the number of packet transmissions during and the length of a sub-busy period that starts when there are n unoccupied spaces in the buffer,

$$\Omega(x, z|n) = \mathrm{E}[x^Q z^C | F = n] . \tag{4}$$

There are $F_k = F - A_S + k$ free buffer spaces at the start of the sub-busy period C_k (see figure 1). Taking this into account, we find that plugging equations

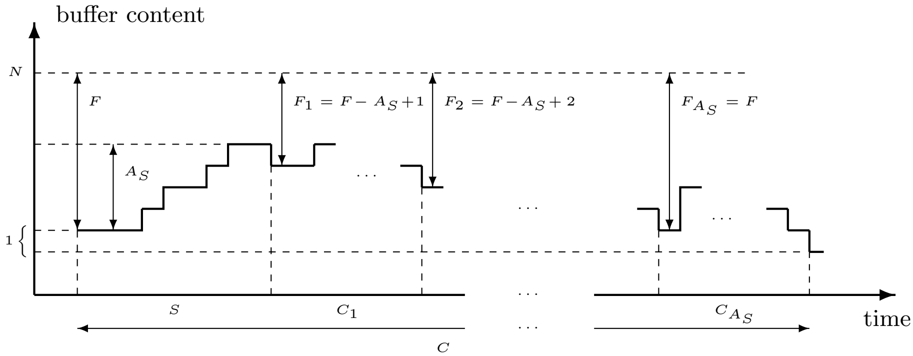


Fig. 1. The sub-busy period of a packet

(2) to (3) into the former expression leads to the following expression by means of some standard probability generating functions techniques:

$$\Omega(x, z|n) = \sum_{j=0}^{n} \Gamma(z, j|n)\, x \prod_{k=n-j+1}^{n} \Omega(x, z|k)\,. \tag{5}$$

The partial conditional probability generating function $\Gamma(z, j|n)$ is here defined as,

$$\Gamma(z, j|n) = \mathrm{E}\left[z^S\, \mathbf{1}(A_S = j) \right| F = n \Big] \ , \tag{6}$$

where $\mathbf{1}()$ denotes the indicator function. As such, $\Gamma(z, j|n)$ is the probability generating function of the packet transmission time given that there are j packet arrivals (excluding lost packets) in the buffer and conditioned on the fact that there are n free buffer spaces at the start of the transmission. Plugging equation (3) into (6) then leads to,

$$\Gamma(z, j|n) = \frac{1}{j!} \frac{d^j}{dx^j} S(zA(x)|n) \bigg|_{x=0} \qquad \text{for } j = 0 \dots n-1, \tag{7}$$

$$= S(z|n) - \sum_{k=0}^{n-1} \Gamma(z, k|n) \qquad \text{for } j = n, \tag{8}$$

for $n = 0 \dots N-1$.

In view of the former expressions, equation (5) expresses $\Omega(x, z|n)$ in terms of $\Omega(x, z|j)$ $(j = 0 \dots n)$ and known functions. Solving for $\Omega(x, z|n)$ then yields,

$$\Omega(x, z|n) = \frac{x\, \Gamma(z, 0|n)}{1 - x \sum_{j=1}^{n} \Gamma(z, j|n) \prod_{k=n-j+1}^{n-1} \Omega(x, z|k)}\,, \tag{9}$$

for $n = 0 \dots N-1$. Recursive application of the former expression then allows us to determine $\Omega(x, z|n)$ explicitly for all $n = 0 \dots N-1$.

We are now ready to focus on the probability generating function of the busy period. The busy period starts after a slot where a packet arrives at an empty buffer. One easily verifies that the probability mass function of the number of arrivals in the buffer (excluding dropped packets) during a slot where there is at least one arrival is given by $\tilde{a}(n) = a(n)/(1 - a(0))$ for $n = 0 \ldots N-1$. The probability $\tilde{a}(N) = 1 - \sum_{j=0}^{N-1} \tilde{a}(j)$ follows from the normalisation. Given that the busy period is initiated by n packets, it takes n sub-busy periods before the buffer is empty again. Further, there are $N - n$ free buffer spaces when the first packet starts transmission, $N - n + 1$ free buffer spaces (just) after the sub-busy period of the first packet, and so on. Therefore, we find following expression for the joint probability generating function of the number of packet arrivals during a busy period and the length of a busy period,

$$\Psi(x, z) = \sum_{n=1}^{N} \tilde{a}(n) \prod_{k=N-n}^{N-1} \Omega(x, z|k) \,. \tag{10}$$

The probability generating functions of the busy period $B(z)$ and of the number of packet transmissions during a busy period $P(z)$ are then given by $B(z) = \Psi(1, z)$ and $P(z) = \Psi(z, 1)$ respectively. These generating functions will allow us to retrieve a number of performance measures as we will see further.

4 Performance Measures

Let μ_B and μ_I denote the mean length of a busy and an idle period respectively and let μ_P denote the number of packet transmissions during this busy period. Using the moment generating property of probability generating functions we find from equations (1) and (10),

$$\mu_B = \sum_{n=1}^{N} \tilde{a}(n) \sum_{k=N-n}^{N-1} \mu_C(k) \,, \quad \mu_I = \frac{1}{1 - a(0)} \,, \quad \mu_P = \sum_{n=1}^{N} \tilde{a}(n) \sum_{k=N-n}^{N-1} \mu_Q(k) \,. \tag{11}$$

Here $\mu_C(k)$ and $\mu_Q(k)$ $(k = 0 \ldots N-1)$ denote the mean length of a sub-busy period and the mean number of packet transmissions during a busy period respectively, given that there are k free buffer spaces when the sub-busy period starts. Equation (9) and the moment generating property yield following set of recursive equations for these mean values,

$$\begin{aligned} \mu_C(k) = {} & \frac{\gamma(0|k)\left(1 - \sum_{i=1}^{k} \Gamma(1, i|k)\right) + \Gamma(1, 0|k) \sum_{i=1}^{k} \gamma(i|k)}{\left(1 - \sum_{i=1}^{k} \Gamma(1, i|k)\right)^2} \\ & + \frac{\Gamma(1, 0|k) \sum_{i=1}^{k} \sum_{j=k-i+1}^{k-1} \Gamma(1, i|k) \mu_C(j)}{\left(1 - \sum_{i=1}^{k} \Gamma(1, i|k)\right)^2} \,, \end{aligned} \tag{12}$$

$$\mu_Q(k) = \Gamma(1,0|k)\frac{1+\sum_{i=1}^{k}\sum_{j=k-i+1}^{k-1}\Gamma(1,i|k)\mu_Q(j)}{\left(1-\sum_{i=1}^{k}\Gamma(1,i|k)\right)^2}, \tag{13}$$

with $\gamma(j|k) = \frac{d}{dz}\Gamma(z,j|k)\big|_{z=1}$. Similar expressions may be retrieved for higher order moments.

Apart from moments of the idle- and busy-period, we can also determine a number of other performance measures. The packet loss ratio (PLR) is defined as the fraction of all packet arrivals that are lost. Consider a random busy period followed by an idle period, say a random cycle. There are no transmissions during the idle period. Therefore μ_P also denotes the mean number of packet transmissions during a cycle. Further, if we denote the mean number of packet arrivals per slot by $\mu_A = A'(1)$, the mean number of arrivals during a cycle equals $(\mu_B+\mu_I)\mu_A$. As packets that arrive in the system are either lost or transmitted, the packet loss ratio is given by, $\mathrm{PLR} = 1-\mu_P/[(\mu_B+\mu_I)\mu_A]$.

Another performance measure is the channel utilisation η, defined as the fraction of slots that a transmission is on-going. As a transmission is on-going only during busy-slots, one easily finds, $\eta = \mu_B/(\mu_B+\mu_I)$.

Finally, the mean packet transmission time m is given by $m = \mu_B/\mu_P$. The latter quantity is a measure for the average quality of the video stream as the video quality is related to the number of transmitted layers and hence also to the size of the transmitted packets.

5 Numerical Examples

We now illustrate our results by means of some numerical examples. We here assume that the video packets are generated according to a Poisson process. As such, the number of packets generated during a slot follows a Poisson distribution. The probability generating function $A(z)$ is given by $A(z) = \exp(\lambda\,(z-1))$. Here λ denotes the mean number of packet arrivals per slot, say the arrival intensity. Further, given the buffer size at the start of transmission, the packet transmission times are deterministically distributed. That is, $S(z|n) = z^{N_n}$, where N_n $(n = 0\ldots N-1)$ denotes the packet transmission time for a given number of free buffer spaces n. The packet transmission time decreases stepwise for decreasing values of the available buffer space. That is, a set of thresholds T_i $(i = 1\ldots K,\ T_1 = N-1,\ T_K = -1$ and $T_i > T_j$ for $i < j)$ is introduced and $N_n = \tilde{N}_i$ for $T_i \le n < T_{i+1}$ $(i = 1\ldots K-1$ and $\tilde{N}_i > \tilde{N}_j$ for $i < j)$. As such, the performance model under consideration is completely specified by the arrival intensity λ and by the series T_i $(i = 1\ldots K)$ and $\tilde{N}_i$ $(i = 1\ldots K-1)$.

In figure 2, we investigate the influence of the buffer size on the performance of the video buffer. In particular, we depict the channel utilisation η (left) and the packet loss ratio PLR (right) versus the buffer size N. The transmission time is fixed to $\tilde{N}_1 = 32$ slots, independent of the buffer size and different values of the packet intensity λ are assumed as depicted. The channel utilisation increases

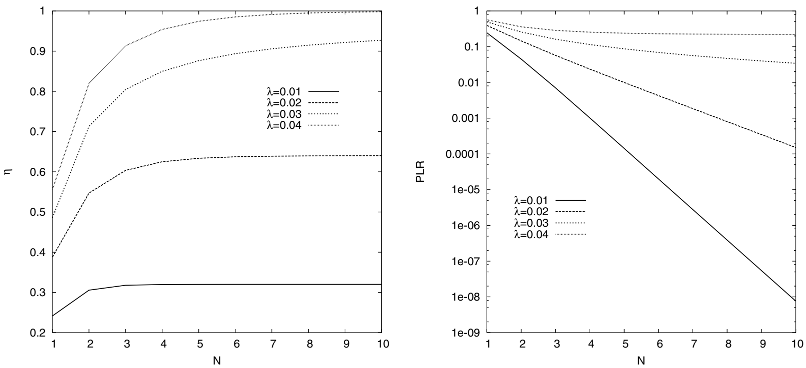


Fig. 2. The channel utilisation η (left) and the packet loss ratio PLR (right) vs. the buffer size N

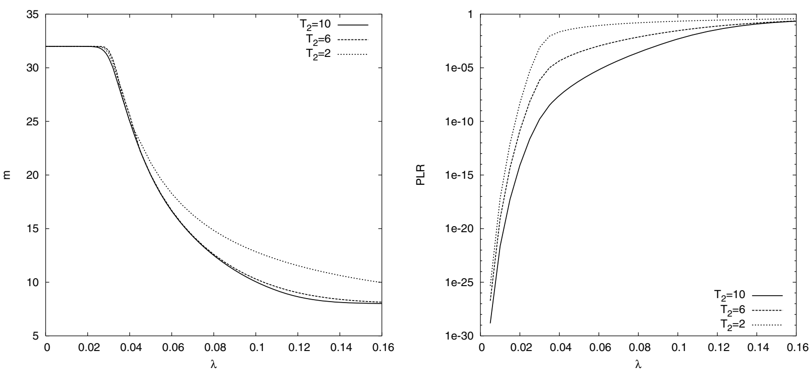


Fig. 3. The mean packet transmission time m (left) and the packet loss ratio PLR (right) vs. the arrival intensity λ

for increasing values of the buffer size and converges to $\min(1, \rho)$ for $N \to \infty$. Here $\rho = \lambda\tilde{N}_1$ denotes the offered load. That is, the utilisation tends to one if the system is overloaded ($\rho > 1$) and tends to the offered load if this is not the case. The packet loss ratio on the other hand decreases for increasing values of the buffer size as more packets can be stored in larger buffers. For increasing values of the buffer size, the packet loss ratio tends to $1 - 1/\rho$ if the system is overloaded and decreases exponentially if this is not the case.

Figure 3 depicts the mean packet transmission time m (left) and the packet loss ratio PLR (right) versus the arrival intensity λ. The buffer size equals 20 and there is a single threshold T_2 ($K = 2$). Different values of the threshold are assumed as depicted and the packet transmission time equals $\tilde{N}_1 = 32$ or $\tilde{N}_2 = 8$ slots depending on the number of unoccupied buffer spaces when the transmission starts. For low values of the arrival intensity and for all values of

T_2, the mean packet transmission time almost equals 32. The buffer occupancy is typically small if the arrival intensity is low. As such, the transmission time of the majority of the packets equals 32. If the arrival intensity increases, more and more packets start transmission when the buffer occupancy is high. As such, the transmission times of more and more of the packets equal 8 slots. Therefore, the mean packet transmission time converges to 8 for increasing values of the arrival intensity. The rate at which the mean packet transmission time converges to 8 depends on the value of T_2. Further, the packet loss ratio increases and tends to one for increasing values of the arrival intensity λ as expected. One observes that the reduction of the transmission time when there are less than T_2 free buffer spaces mitigates packet loss. The packet loss ratio is smaller for higher T_2, i.e., when the transmission time is decreased earlier (when there is more buffer space available).

6 Conclusions

In this contribution the performance of an output buffer of a video streaming server was investigated. For this, we constructed a queueing model with buffer size dependent transmission times. A busy period analysis led to expressions for performance measures such as the packet loss ratio and the mean packet transmission time. We showed by means of some numerical examples that a reduction of the transmission times can lead to lower values of the packet loss ratio.

Acknowledgements

This work has been partially supported by the “Interuniversity Attraction Poles Programme – Belgian Science Policy”.

References

1. Turelli, T., Parisis, S., Bolot, J.: Experiments with a layered transmission scheme over the internet. Technical Report 3295, INRIA (1997)
2. Radha, H., Chen, Y., Parthasarathy, K., Cohen, R.: Scalable internet video using MPEG-4. Signal Processing: Image Communication **15** (1999) 95–126
3. Kangasharju, J., Hartanto, F., Reisslein, M., Ross, K.: Distributing layered encoded video through caches. IEEE Transactions on Computers **51** (2002) 622–636
4. Zwart, A.: Tail asymptotics for the busy period in the $GI/G/1$ queue. Mathematics of Operations Research **26** (2001) 485–493
5. Agarwal, M.: Distribution of number served during a busy period of $GI/M/1/N$ queues - lattice path approach. Journal of Statistical Planning and Inference **101** (2002) 7–21

6. Ohta, C., Morii, M.: Moment calculating algorithm for busy-period of discrete-time finite-capacity $M/G/1$ type queue. IEICE Transactions on Communications **E85B** (2002) 293–304
7. Artalejo, J., Lopez-Herrero, M.: Analysis of the busy period for the $M/M/c$ queue: An algorithmic approach. Journal of Applied Probability **38** (2001) 209–222
8. Baltrūnas, A., Daley, D., Klüppelberg, C.: Tail behaviour of the busy period of a $GI/GI/1$ queue with subexponential service times. Stochastic Processes and their Applications **111** (2004) 237–258

Feedback Control Using State Prediction and Channel Modeling Using Lower Layer Information for Scalable Multimedia Streaming Service[1]

Kwang O. Ko, Doug Young Suh, Young Soo Kim, and Jin Sang Kim

Multimedia Research Center, Kyunghee University,
1, seochunri, giheungeuop, younginsi, kyungido, Korea
Tel: +82-31-201-2586, Fax: +82-31-203-1494
inverser@gmail.com, {suh, yskim, jskim}@khu.ac.kr

Abstract. cdma2000 EV-DO service was deployed in Korea in 2004. Even though it provides a bandwidth up to almost 2 Mbps, channel quality is time-varying so that service quality should be controlled adaptively. This paper discusses the measurement and analysis of the EV-DO channel quality. Based on the analysis, it proposes a proxy node that is located in the base station, informs the corresponding node of current channel quality. And information from the lower layer is also useful because it can be used to adapt the channel state more quickly and accurately . The proposed feedback control can be used well with scalable video coding.

1 Introduction

In South Korea, cdma2000 EV-DO service, which supports HDR(High speed Data Rate) is widely deployed. Theoretically, the EV-DO downlink is as wide as 2 Mbps. Apart from the costs, MMS(Multimedia Message Service), VOD(Video On Demand), and MOD(Music On Demand) services can be supported. However, since the IP-based core network of EV-DO service uses a best-effort protocol and the condition of the wireless channel can vary with time, real-time multimedia services can not yet be adequately supported in the current EV-DO network..

2 Experiment Test-Bed

Figure 1 shows the test-bed used for our measurements. A cdma2000 1x EV-DO mobile terminal is connected to a notebook computer in a passenger car. The channel was measured for about 20 minutes between 12:00 and 14:00.

[1] This work was supported by grant No. R01-2003-000-10149-0 from the Basic Research Program of the Korean Science & Engineering Foundation.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 901–908, 2005.

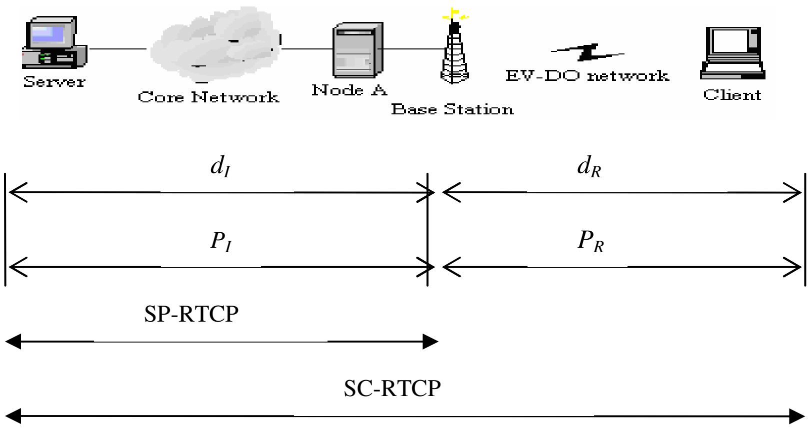


Fig. 1. Test-bed with proposed proxy node, Node A

Delay and packet loss rate between the server and Node A can be defined as d_I *and* P_I and those between Node A and mobile terminal can be defined as d_R *and* P_R. Then, total delay and packet loss rate can be described as $d= d_I + d_R$, $P= P_I+P_R - P_IP_R$. RTCP sessions between the server and Node A, and between Node A and the mobile terminal, are defined as SP-RTCP and SC-RTCP, respectively.

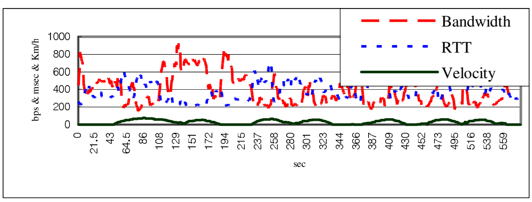


Fig. 2. Measurement of available bandwidth, RTT, speed of vehicle

2.1 End-to-End Measurements

Figure 2 shows the results of our experiment. We found that dynamic ranges of RTT and available bandwidth are much wider than expected. Such a wide variation makes it difficult to model channels for multimedia transmission in mobile networks. Average bandwidth and RTT are 389kbps and 390msec, respectively (the larger RTT, the smaller the available bandwidth).

2.2 Analysis of Measured Data Extraction of Gilbert Parameters

In order to obtain Gilbert parameters from the experiment, we define the channel condition in which RTT is longer than 400msec as good state, and shorter than 400msec as bad state. In the experiment, IDT(Inter Departure Time) is 20msec and packet size is 2000 bytes. From Gilbert model, we can calculate the state transition probabilities by using algorithm used in [1]. Compared to wired networks, mobile networks have special characteristics, including a long time delay and burst loss, which can create a time delay in receiving feedback information.

Table 1. Gilbert Model parameters from experimental results

	Holding-time	packet	time
Good State	3.885sec	0.064	0.005
Bad State	0.92	0.272	0.022

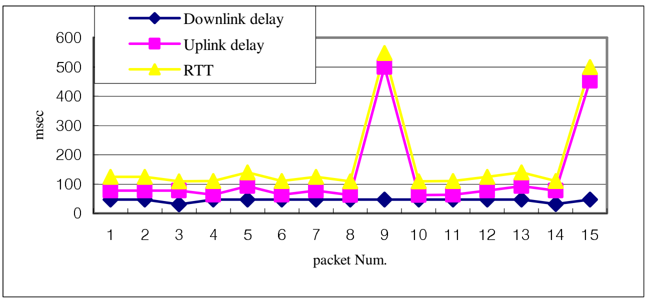


Fig. 3. Measurement of uplink and downlink delay of cdma2000 EV-DO channel

2.3 One Way Delay Analysis

Radio channel in cdma2000 EV-DO networks is asymmetric. Since RTT information in RTCP header can not distinguish downlink delay from uplink delay, it is not useful for feedback control. For a streaming service, status of downlink is more significant than that of uplink because downlink channel is used for streaming requiring high bandwidth. In Figure 3, RTT of Packet 9 is 547msec. If this value is used for feedback control as in RTCP based feedback control, server may consider that current channel state is bad and will lower streaming bit-rate even though downlink delay is 47msec and channel state is good enough.

3 Trace Generation Algorithm

We propose an approach to cdma2000 1x EV-DO networks based on the results of our experiment. Currently, there are two type of loss in mobile networks. One is due to fading and the other one is due to buffer overflow in the base-station. Because such losses force retransmissions of data, RTT is increased and available bandwidth is decreased.

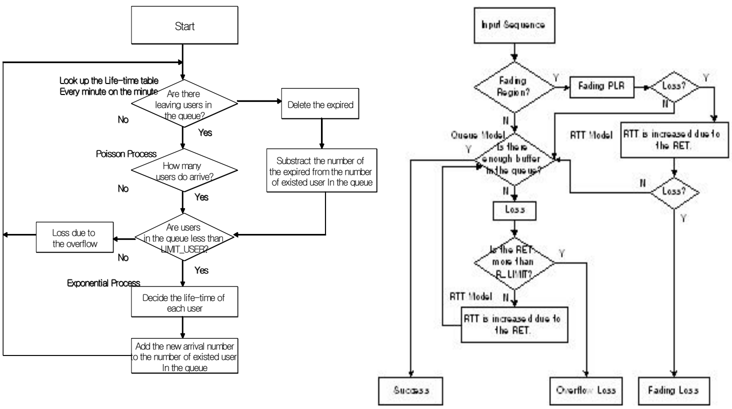


Fig. 4. Queue model and end-to-end channel modeling for generating new trace

Because the base-station knows its own buffer level and whether the mobile station is in a region susceptible to fading, it can report this information to the server. This information can be used to increase the effectiveness of the multimedia service. Figure 4 shows a queuing model that describes the buffer occupancy over time and channel modeling procedure that generates the trace, based on our experiment.

Because the base-station knows its own buffer level and whether the mobile station is in a region susceptible to fading, it can report this information to the server. Figure 4 shows a queuing model that describes the buffer occupancy over time and channel modeling procedure that generates the trace, based on our experiment. The number of users with access to the queue can be determined as a Poisson process every minute. Each user will remain in the queue for a period set by exponential distribution. If buffer occupancy reaches buffer capacity, the buffer overflows. From the Gilbert parameter, we determine whether the terminal location is in a region susceptible to fading. If it is, packets will be lost according to fading PLR, and RTT will be increased because of packet retransmission. If the retransmitted packets are also lost, retransmission will be repeated until R_LIMIT (the limit of the number of retransmissions). If the number of retransmissions exceeds R_LIMIT, the packet will be lost (due to fading). Next, using the Queue Model, we determine whether the buffer in the base-station can be used. If it can, the packet will be transmitted successfully. But otherwise, the retransmission mechanism will be used again because of buffer overflow.

Table 2. Gilbert parameter from new traces

	Holding-time	packet	time
Good State	3.393sec	0.074	0.006
Bad State	0.586	0.426	0.034

Table 2 shows the Gilbert parameter set of the newly generated traces. We generate 10000 trace packets using our trace generator. Each new packet has lower-layer information about loss, RTT, available bandwidth, the number of retransmissions. Because transition probabilities in Table 2 are close to transition probabilities in Table 1, we can say that our trace generator can generate the traces considered channel state. Using the information from the new generating trace, we can obtain the buffer occupancy in the base-station.

4 Proposed Transmission Architectures

4.1 Transport Layer Feedback

In general, end-to-end RTCP feedback is used in the transport layer for scalable video coding. But it's better than single-layer video streams under time-varying channel states. But when existing RTCP feedback is used, such a gain is decreased during the state transition period because of end-to-end delay. The server is sometimes also informed of the current channel state after RTT is passed. Because of such a delay, after state transition moments, the current state of the channel will be different from the state perceived at the server.. We refer to it as a 'false alarm' when the current channel state is good while the server perceives that it is still bad. The opposite case is defined as 'overflow.'

4.2 Feedback Control with the Proxy Node (Node A)

The server predicts the channel condition based on accumulated end-to-end feedback information. By using a proxy node, Node A, as shown in Figure 1, end-to-end information can be decomposed into wire-line Internet network information and that of the wireless channel. Error control and bit-rate control policies must be determined by source of QoS deterioration such as delay and packet loss.

There are two possible reasons for packet losses in a wireless channel; fading and congestion. The base station is aware of current channel conditions. During a fading state, available bit-rate is kept constant while PLR becomes higher. If VSF(variable spreading factor) is controlled dependently on path gain (i.e. SNR), available bit-rate can be lowered. Congestion means that since the wireless channel is shared with new comers (calls), allocated bandwidth is lowered.

Table 3. Adaptation policy of an audio/video traffic

state	fading	congestion	Proposed control
Good	No	No	OK
Bad	Yes	No	Lower bit-rate, Increase FEC
Bad	No	Yes	Lower bit-rate
Bad	Yes	Yes	Audio Service Only

Since wired networks are shared by large number of users, decrease of bandwidth at times of congestion does not enhance QoS immediately. A decrease of bandwidth for individual service is recommended, to lower congestion of the total network in TCP-friendly flow control. At congestion of wired network, however, FEC is helpful for individual real-time service, while it is not recommendable as a citizen of network.Loss prevention policy depends on loss pattern. If packet loss is randomly spread, FEC is helpful, while if bursty, retransmission or interleaving, FEC is helpful.

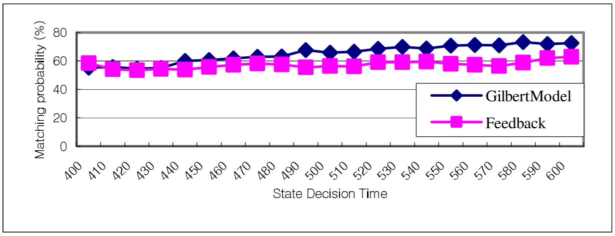


Fig. 5. Matching probability of the proposed (Gilbert) Model compared to previous feedback mechanism

4.3 Effect of Channel Prediction

Compared with cdma2000 1x networks, EV-DO networks provide a better environment for QoS control because RTT is much shorter. But for adaptive feedback control, enough feedback information has to be received on time. There is a tradeoff between accuracy of feedback information and delay of the information. In this paper, we argue that less accurate, but, faster information is more useful for bit-rate control and error control for real-time multimedia service. Figure 5 shows performance of the proposed feedback control compared to that of previous feedback mechanism. In [1], when bad state holding time is 1.5 times longer than RTT, feedback control becomes effective.

Since the proposed prediction method reduces feedback delay virtually, the percentage of duration of effective feedback control can be increased. For real-time service over wireless channel, delay of feedback information is more critical than accuracy of the information. The proposed prediction method reduces the delay while sacrificing (unneeded) accuracy.

4.4 Feedback Control of Layered Video Service

Adaptive control technique is very effective to control a layered multimedia service. A video can be encoded into multiple bitstreams so that it can be adapted to time varying network condition. For layered coding, the traffic controller is supposed to just select how many bitstreams to send. The layered approach is more useful in traffic control in routers in the middle of a network since they are not supposed to know video codecs.

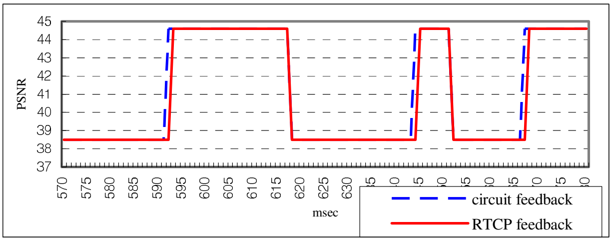


Fig. 6. Effect of faster feedback by using circuit switching channel for feedback information during the transition period

Cdma 2000 EV-DV allows both a packet-switching channel and a circuit-switching channel in a call. We propose that SP-RTCP be sent through the circuit switching channel so that the server can receive the report immediately. We compare previous feedback control systems with our proposed one using MPEG-4 scalable video. ‘Foreman’ is used for video data and frame rate is 15 frame/sec.

Figure 6 shows that, compared with previous feedback control, the PSNR gain can be observed at the moment after state transition. PSNR of existing feedback control is 41.66dB on average and average PSNR of proposed feedback control is 41.80dB. Loss in of PSNR because of false alarms and overflows can be reduced using the proposed feedback control at the moments of state transition

5 Conclusions

In this paper, we propose more effective feedback control techniques for real-time multimedia services in cdma2000 1x EV-DO. We use data gathered from the experiments. Using the traces from the experiment, we model the mobile network conditions and generate traces. As opposed to [2], [3], [5], we use the accumulated information to predict the channel state in order to reduce the temporal gap between control action decisions and channel use. This temporal gap is reduced by using a proxy node. We also propose to use the circuit-switching channel for feedback from the proxy node to the server. Performance of three approaches is demonstrated by using layered MPEG-4 video service.

References

[1] Yun Bum Jung, Doug Young Suh, Kwang Oh Ko and Sang Jo Lee, "Measurement and analysis of cdma2000 1x channel for Internet multimedia services," CIC2003, Oct. 2003.

[2] D. Morikawa, S. Ota, A. Yamaguchi, M. Ohashi, "A Feedback Rate Control of Video Stream in Best-Effort High-Speed Mobile Packet Network," Wireless Personal Multimedia Communications, vol. 2, pp. 807-811, Oct. 2002.

[3] T. Yoshimura, T. Ohya, T. Kawahara, and M. Etoh, “Rate and Robustness Control with RTP Monitoring Agent for Mobile Multimedia Streaming,” ICC 2002, vol. 4, May 2002.
[4] Multimedia Streaming Services , The Third Generation Partnership Project 2, TSG-S Specifications, S.R0021-0 v2. April 2002.
[5] G. Cheung, W. Tan, T. Yoshimura, “Rate-distortion optimized application-level retransmission using streaming agent for video streaming over 3G wireless network,” ICIP 2002, vol. 1 ,pp. I-529-I-532, Sept. 2002.
[6] Almudena Konrad, Anthony D. Joseph, Reiner Ludwig, Ben Y. Zhao,"A Markov-Based Channel Model Algorithm for Wireless Networks," ACM MSWiM, July 2002.
[7] Ping Ji, Benyuan Liu, Don Towsley, Zihui Ge, Jim Kurose, "Modeling Frame-Level Errors in GSM Wireless Channels", In Proc. of IEEE Globecom 2002.

Low Delay Multiflow Block Interleavers for Real-Time Audio Streaming*

Juan J. Ramos-Muñoz and Juan M. Lopez-Soler

Signals Theory, Telematics and Communications Department,
E.T.S. Ingeniería Informática, University of Granada, Spain
jjramos@ugr.es and juanma@ugr.es

Abstract. In spite of the Internet design principle of putting the complexity on the end-to-end entities, this work contributes to demonstrate what benefits can be expected by adding some processing capabilities to the network nodes for the class of interactive audio streaming applications. In particular, we deal with the bursty-error-prone nature of the Internet by proposing and evaluating a new multiflow block interleaver algorithm. After the conducted simulations, we show that our algorithm can efficiently mitigate the negative impact of long bursts. And what it is more, it is achieved by fulfilling the end-to-end audio time constraint requirements.

1 Introduction

Interactive multimedia streams, such as data from audio conferences, by definition must reach their destination hosts before a tight time limit (e.g. 300 ms for voice streams). In this context, the use of end-to-end recuperation techniques to face packet losses is highly restricted. That is, recovering a lost packet potentially can introduce a delay that in many cases is not affordable.

For any streaming application, besides of losing packets, the bursty-error-prone nature of the Internet has a supplementary negative impact in the final end-to-end provided quality. It is well established that in streaming applications packet losses are more harmful as they are consecutive, given that the subjective quality degradation increases as the burst length increases [1]. Therefore, to improve the quality of the provided service, some procedures must be considered to combat the unwished burstiness effect. To this end, it is desirable to scatter the pattern of losses, ideally without increasing both the bandwidth consumption and the end-to-end delay.

Traditionally, error control techniques operate end-to-end ([2]). However, with the advent of the Active Networks (AN) technology [3], and the Overlay Networks (ON) approach [4], new promising router and node functionalities

* This work has been partially financed by the Spanish Science and Technology Ministry under Research Project TIC2002-02978 (with 70% of FEDER funds) and Spanish MECD under National Program PNFPU (reference AP2002-3895).

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 909–916, 2005.

can be envisaged. Active routers, or overlay network nodes, are able not only to switch packets but also to process the ongoing information. In this new technology framework, a number of studies worth mentioning have experimentally confirmed the performance improvements that AN and ON technologies can introduce in real-time multicast applications (e.g. [5], [6], [7], [8], [9] and [10], among others).

In this work, we focus our interest on the burstiness of the packet losses problem. We take up again the packet interleaving approach but now by considering the new processing capabilities of the network elements. Because of intermediate network nodes can use different multimedia flows, we propose an interleaver algorithm that take advantage of that fact. Furthermore, given the router processing capabilities, the interleaver can be adapted to the network condition dynamics. We claim that our algorithm efficiently combat the bursty-error-prone nature of the Internet. After a number of simulations, we evaluate what performance improvements can be expected. For comparison purposes, we also simulate a single flow end-to-end interleaver. We experimentally demonstrate, under some circumstances, that if the number of different flows available to interleave is less than the expected burst length, our algorithm improves the so considered reference system with light impact in the end-to-end packet delay.

To this end, this paper has been organized into the following structure: in Section 2, we set out the notation and describe the basic interleaving theory. In Section 3 we propose the new multiflow block interleaver. After the conducted simulations, we present and discuss the performance of the proposed scheme in Section 4. Section 5 provides the main conclusions of this work. Finally, bibliographical references are also provided.

2 Basic Block Interleaving Theory

Let us define an interleaver as a device whose input is a sequence of symbols of a given alphabet, and whose output is a reordered sequence of the same input symbols. More specifically, if the input sequence is denoted by $\ldots, a_{-1}, a_0, a_1, a_2, a_3, \ldots$, and the output sequence is $\ldots, b_{-1}, b_0, b_1, b_2, b_3, \ldots$ the interleaver defines a permutation $\pi : \mathbb{Z} \mapsto \mathbb{Z}$ such that $a_i = b_{\pi(i)}$. This permutation is one-to-one map. Associated to π there is the corresponding de-interleaver defined simply by the inverse π^{-1}.

An (n_2, n_1) interleaver reorders the input sequence so that no contiguous sequence of n_2 symbols in the output sequence contains any symbols separated by fewer than n_1 symbols in the input sequence [11]. Therefore, it is verified that

$$|\pi(i) - \pi(j)| \geq n_1, |i - j| \leq n_2 - 1 \tag{1}$$

An interleaver is said to be periodic if it verifies that $\pi(i + p) = \pi(i) + p$, being p its period.

For interactive audio streaming applications, to reduce the end-to-end delay is a must. Therefore, an audio packet interleaver design must carefully consider the packet delay problem. An interleaver has an *spread* s, if any two symbols in an interval of length s are separated by distance of at least s at the output. Given an spread of $n_1 = s$ there is not block interleavers with period less than s^2 [12]. This means that the associated block interleaver matrix must be squared $(s \times s)$. Therefore, for getting the minimum delay packet reallocation, given an spread s, the algorithm (hereafter referred to as *Type I* (s))must be:

1. Arrange the symbols corresponding the input packets in a $(s \times s)$ matrix in rows, from left to right and from top to bottom.
2. Read the matrix by columns from bottom to top and from left to right, and accordingly send the packets.

In this case, the maximum delay in terms of number of symbols that any packet will suffer, D_{max}, will be equal to

$$D_{max} = s \cdot (s - 1) \tag{2}$$

3 Multiflow Block Interleavers

Although packet interleaving has been already considered for single audio flows [13], the introduced delay can make it potentially unfeasible for dealing with large spreads. More precisely, a single audio interleaver will be limited to s such as $(s \cdot (s - 1)) \cdot t_f < d_{max}$, where t_f is the inter-packet period, and d_{max} is the maximum end-to-end delay that any packet can tolerate. For example, for typical values of $d_{max} = 300$ ms and $t_f = 22$ ms, no isolated losses can be obtained if bursts length are expected to be longer than 4 packets.

However, if one notes that at any intermediate router more than one flow will be available, if we interleave more than just one flow, a reduction in the end-to-end packet delay can be potentially achieved.

Based on this idea, to deal with packets losses bursts shorter than $s + 1$ packets, we propose two packet interleavers by using n_f different flows. In our proposal the reading process will remain unchanged, but the writing algorithm is somehow slightly different. Implicitly, to work properly, all the flows are assumed to have the same period t_f, and of course, they must share some common path.

To fill the interleaver matrices, as a general rule, each flow will maintain the relative order with respect to the others. That is, the first flow will occupy the first rows, the second one will follow them, and so on. Additionally, for a given flow each row will be written from left to right according to the packet sequence number.

Let $(f^1, f^2, \ldots, f^{n_f})$ be the n_f available audio flows, and let s be the maximum expected burst length. To describe the writing matrix procedure, let us additionally define R^i_j, with $i = 1, \ldots, n_f$ and $j = 1, \ldots, n_m$, as the number of consecutive rows that the flow f^i will be assigned for filling the interleaver matrix j, being n_m the number of matrices.

Depending on n_f and s, we will consider two different cases.

1. Whenever $n_f \geq s$, the interleaver will be based on just one $(n_f \times 1)$ matrix $(n_m = 1)$, in which $R_1^i = 1, \forall i = 1, \ldots, n_f$. For this particular case, the interleaver output will be given by ..., f_i^1, f_j^2, ..., $f_k^{n_f}$, f_{i+1}^1, f_{j+1}^2, ..., $f_{k+1}^{n_f}$, ..., where the subscripts i, j, ..., k denote the sequence number for flows f^1, f^2, ..., f^{n_f}. We refer to this interleaver as *Type II* (n_f).
2. If $n_f < s$, we will refer to this interleaver as *Type II* (n_f, s). Under this condition, two different cases will be considered.
   - If $s = n_f \cdot i, i \in \mathbb{N} \Rightarrow n_m = 1$. That is, if the expected burst length is any integer multiple of the number of audio flows, only one interleaver $s \times s$ matrix will be used;
   - Otherwise, $n_m = n_f$ square $(s \times s)$ matrices will be required.

   Going ahead, if we denote $rem(x, y)$ as the remainder of the integer division x/y, the writing matrices algorithm will be as follows:
   - For the first matrix, we will set $R_1^i = \left\lfloor \frac{s}{n_f} \right\rfloor$, for $i = \{1, 2, \ldots, (n_f - rem(s, n_f))\}$. That is, the first $(n_f - rem(s, n_f))$ flows occupy $\left\lfloor \frac{s}{n_f} \right\rfloor$ rows each. And similarly, we will set $R_1^j = \left\lfloor \frac{s}{n_f} \right\rfloor + 1$, for $j = \{(n_f - rem(s, n_f) + 1), \ldots, (n_f - 1), n_f\}$. In other words, the last $rem(s, n_f)$ flows will be assigned with $\left\lfloor \frac{s}{n_f} \right\rfloor + 1$ rows each.
   - If applicable, for the next $j = 2, \ldots, n_f$ additional matrices, and for $i = 2, \ldots, n_f$ flows, if $R_{(j-1)}^i = (\left\lfloor \frac{s}{n_f} \right\rfloor + 1)$ and $R_{(j-1)}^{(i-1)} = \left\lfloor \frac{s}{n_f} \right\rfloor$ then $R_j^i = \left\lfloor \frac{s}{n_f} \right\rfloor$ and $R_j^{(i-1)} = (\left\lfloor \frac{s}{n_f} \right\rfloor + 1)$.

As it can be checked, no burst of length less or equal to s at the input will make two consecutive packet losses at the de-interleaver output.

3.1 Interleaver Analysis

With the aim to shed some light in the evaluation of the proposed algorithms, let us calculate the worst case incurred delay. In so doing, we will check whether the goal of reducing the burstiness effect is achieved without violating the end-to-end time constraint.

The maximum packet delay D_{max} can be expressed as:

$$D_{max} = \begin{cases} s \cdot (r \cdot (dnf + 1) - 1 - (r - 1) \cdot dnf) & if\ r \leq (n_f - r) \\ s \cdot (r \cdot (dnf + 1) - 1 - ((r - 1) \cdot dnf + 2 \cdot r - n_f - 1)) & if\ r > (n_f - r) \end{cases}$$

where $r = rem(s, n_f)$, and $dnf = (s - r)/n_f$.

For a given burst length equal to s, the lower maximum delay that we can obtain is achieved when $n_f = s - 1$, and when $s/n_f = 2$ and $r = 0$. That delay corresponds to $D_{max} = s$, that is to say $s \cdot t_f$ ms. Therefore, the maximum tolerated s, given a flow with a maximum per packet time to live d_{max} and a period of t_f must satisfy that $s < \frac{d_{max}}{t_f}$. For the previously provided numerical example, in which $s \cdot t_f < d_{max}$, $t_f = 22$ ms and $d_{max} = 300$ ms, it yields that $s < 14$, which is significantly less demanding compared to the upper bound of $s < 5$ for the *Type I* (n_f, s) *end-to-end interleaver*. The period of the proposed *Type II* (n_f, s) *interleaver* is equal to $p = \frac{s}{n_f} \cdot s$, if $s \equiv 0 \ (mod\, n_f)$, and $p = s^2$ in the other case.

4 Experimental Results

Experimental results are provided by means of simulations. Several scenarios have been tested in order to compare the suitability and benefits of the multiflow interleaving, using several error models which exhibit different error patterns.

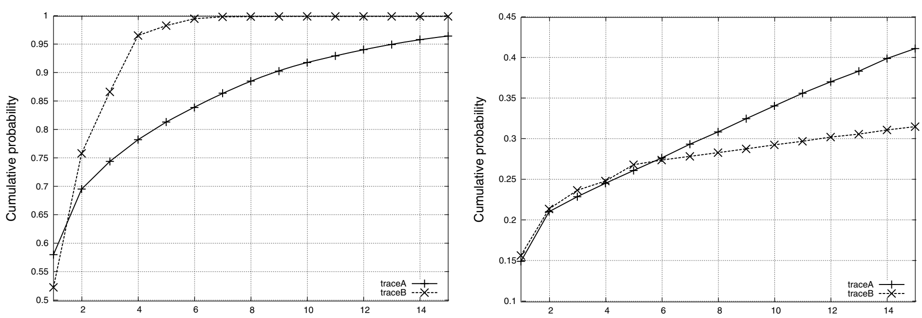
4.1 Simulation Framework and Error Models

A simple scenario is set, where periodic packets from n_f flows arrive into a router with period equal to $t_f = 22$ ms. After the router, we suppose that the unwished packet losses take place. We assume that the throughput of the router and the output bandwidth are enough to dispatch the packets with no additional switching delay.

A number of error models have been proposed in the literature but only a few of them model the burst of losses distribution and the interloss distance. In our experiments, we will use a simple model which takes into account both behaviors. It is based on a Markov chain trained with collected traces described in [14]. In the simulations we will use the exact model proposed in [14], hereafter referred to as error model *traceA*. Besides of that, we will use one additional 6th order Markov chain (*traceB*) trained with our own collected traces. CDFs of the traces obtained from the trained models are shown in Fig. 1.

4.2 Simulation Results

To evaluate the proposed schemes we use the objective measure given by the burst length at the output of the de-interleaver system. Perceived subjective audio quality will strongly depend on the length of the bursts for each flow.



a) Burst CDF b) Interloss CDF

Fig. 1. Burst losses and interlosses CDFs

Table 1. De-interleaver bursts for trace *traceB* and $n_f = 4$ flows

Scheme	D_{max}(sec)	1	2	3	4	5	6	> 6
generated losses		59.872	19.265	10.152	7.594	1.599	1.439	0.080
Type II (4)	0.0000	86.755	10.396	2.639	0.158	0.053	0.000	0.000
Type II $(4, 8)$	0.1760	92.439	7.512	0.049	0.000	0.000	0.000	0.000
Type I (8)	1.2320	99.223	0.731	0.046	0.000	0.000	0.000	0.000

The maximum burst length s for the interleavers is chosen based on the error model CDF, in such a way that the interleaver intends to isolate the $90\% - 95\%$ of the expected bursts. For this purpose, for *traceA* we set $s = 10$, and for *traceB* we use $s = 8$. Obviously, in an real scenario, the expected s value, can be actively estimated on-line, in order to be adapted to the network dynamics.

In Table 1 we show the average bursts distribution at the output for all the flows. The maximum delay, D_{max} expressed in seconds, experimented by any packet for error model *traceB* with $n_f = 4$ and s=8 is also shown. The following 6 columns show the percentage of bursts of length 1 to 6 and, finally, the ninth column shows the percentage of bursts longer than 6. The label *generated losses* refers to as the packet losses experimented at the de-interleaver input.

Even though the Type I scheme seems to result in better redistribution of losses, and even reaching the higher percentages of isolated losses, it is not suitable for any s longer than 4, since the maximum packet delay exceeds the allowed threshold, as it can be checked by using expression (2). Note that Type I (8)-interleaver surpasses the end-to-end delay constraint of 300 ms.

We also can see how the Type II (n_f)-interleaver is not suitable for cases where $n_f << s$. In fact, this scheme exhibits bursts of length equal to 5, while the Type I (s)-interleaver and the Type II (n_f, s)-interleaver exhibit less than a 0.05% of bursts longer than 2 consecutive packet losses.

Anyway, we must point out that the Type II $(4, 8)$-interleaver obtains fair distributions of losses, by just consuming an affordable amount of time, with a maximum delay per packet equal to 176 ms. In Table 2 we can verify how the Type II (n_f) interleaver does not work properly when $n_f << s$, resulting in a 3.840% of bursts longer than 6 onsecutive packets. Although Type I (10)-interleaver features the best output distribution, its $D_{max} = 1.9810$ sec surpasses the maximum threshold of $d_{max} = 300$ ms.

Therefore, the Type II (n_f, s)-interleaver achieves the best performance trade-off: a tolerable $D_{max} = 220$ ms, with just 71.190% of isolated losses. In addition, just the 0.5% of the bursts are longer than 3 consecutive lost packets.

Table 2. De-interleaver bursts for trace *traceA* and $n_f = 5$ flows

Scheme	D_{max} (sec)	1	2	3	4	5	6	> 6
generated losses		56.905	13.235	3.342	4.280	1.909	3.226	17.103
Type II (5)	0.0000	57.028	19.101	9.205	5.773	3.547	1.507	3.840
Type II $(5, 10)$	0.2200	71.191	18.185	5.562	4.549	0.468	0.000	0.044
Type I (10)	1.9800	87.981	11.281	0.718	0.020	0.000	0.000	0.000

Table 3. De-interleaver bursts for trace *traceA* and $n_f = 12$ flows

Scheme	D_{max} (sec)	1	2	3	4	5	6	> 6
generated losses		56.898	13.358	3.295	4.302	1.850	3.240	17.057
Type II (12)	0.000	70.533	18.930	7.268	2.893	0.235	0.094	0.047
Type II (12, 13)	0.2860	74.562	16.250	7.662	1.502	0.015	0.010	0.000
Type I (13)	3.4320	88.427	10.755	0.803	0.014	0.000	0.000	0.000

Finally, in Table 3 it is shown how the $n_f \times 1$-interleaver scheme is preferred in cases where $n_f \approx s$. It can be seen that although the Type II $(12, 13)$-interleaver results again in a better distribution of bursts, getting 74.562% of the losses isolated, it needs 0.2860 sec, nearly the maximum threshold. However, Type II(12)-interleaver reduces the percentage of bursts longer than 5 from the original 20.297% packets to the 0.141%, with a maximum theorical added delay equal to 0. This trade-off between resulting output loss distribution and the maximum delay introduced, must be considered in order to decide dinamically which inteleaver should be chosen in the case of $n_f \approx s$.

5 Conclusions

In this paper the block interleaving problem is revisited for audio applications. To increase the final audio quality we aim to scatter long bursts of packet losses. Our algorithm is designed to interleave packets from different flows which exhibit the same characteristics. To work properly, the interleaver must be placed in a common node before the path where losses are expected to occur.

Our proposed interleavers diminish the per packet delay compared to the end-to-end single flow interleaver. We show that the use of the end-to-end interleaver for audio streaming is very restricted to small bursts. However, the new proposed schemes can be efficiently used by using different n_f flows. We experimentally demonstrate that in this case, longer burst length can be dispersed.

We also show that the use of the classical minimum delay block interleaver (the Type I (s)-interleaver), although resulting in a great number of isolated losses, is restricted to network conditions in which the expected burst length is shorter than a given low threshold. To break this strong limitation, the multiflow interleavers are proposed.

Type II (n_f) and (n_f, s) interleavers diminish the packet interleaving delay. Although the Type II (n_f) interleaver is designed to work properly when $n_f \geq s$, it is also well suited when $n_f \approx s$, without introducing any additional delay. Compared to Type II (n_f), the Type II (n_f, s) interleaver behaves similarly. It scatters a high percentage of losses patterns, and reduces the maximum length of the bursts at the de-interleaver output. Furthermore, Type II (n_f, s) can be used under conditions that Type II (n_f) does not tolerate (long burst length and low number of different flows), however it introduces additional delay.

In this paper we have considered just the case in which the interleaving process is applied to packets from n_f different flows. However, it still remains, as an open question for future work, to include additional aggregated traffic with different characteristics (that is to say, with different inter-packet period). The additional aggregated traffic should be used to decrease the resulting bursts length taking into account the bursts and interloss distances distributions.

References

1. Liang, Y.J.; Apostolopoulos, J.G.B.: Model-based delay-distortion optimization for video streaming using packet interleaving. In IEEE, ed.: Record of the Thirty-Sixth Asilomar Conference on Signals, Systems and Computers. Volume 2., IEEE (2002) 1315–1319
2. Towsley, D., Kurose, J., Pingali, S.: A comparison of sender-initiated and receiverinitiated reliable multicast protocols. IEEE Journal on Selected Areas in Communications **15** (1997) 398–406
3. Tennenhouse, D.L., Smith, J.M., Sincoskie, W.D., Wetherall, D.J., Minden, G.J.: A survey of active network research. IEEE Communications Magazine **35** (1997) 80–86
4. Doval, D.; OMahony, D.: Overlay networks: A scalable alternative for P2P. IEEE Internet Computing **7** (2003) 79–82
5. Calderon, M., Sedano, M., Azcorra, A., Alonso, C.: Active network support for multicast applications. IEEE Network **12** (1998) 46–52
6. Lehman, L.W.H., Garland, S.J., Tennenhouse, D.L.: Active reliable multicast, San Francisco. CA, USA, IEEE INFOCOM-98, IEEE (1998) 581–589
7. Banchs, A., Effelsberg, W., Tschudin, C., Turau, V.: Multicasting multimedia streams with active networks. In: Proceedings of the 23rd Annual Conference on Local Computer Networks (LCN), Lowell, MA, USA, IEEE (1998) 150–159
8. Amir, Y.; Danilov, C.: Reliable communication in overlay networks. In IEEE, ed.: 2003 International Conference on Dependable Systems and Networks, IEEE (2003) 511–520
9. Pappas, V.; Zhang, B.Z.L.T.A.: Fault-tolerant data delivery for multicast overlay networks. In IEEE, ed.: Proceedings. 24th International Conference Distributed Computing Systems. (2004) 670–679
10. Zhu Y., Li B., G.J.: Multicast with network coding in application-layer overlay networks. IEEE Selected Areas in Communications **22** (2004) 107–120
11. Ramsey, J.: Realization of optimum interleavers. IEEE Transactions on Information Theory **16** (1970) 338–345
12. Kenneth Andrews, C.H., Kozen, D.: A theory of interleavers. Technical Report 97-1634, Computer Science Department, Cornell University (1997)
13. Perkins, C., Hodson, O., Hardman, V.: A survey of packet-loss recovery techniques for streaming audio. IEEE Network, September/October (1998) 40–48
14. Yajnik, M., Kurose, J., Towsley, D.: Packet loss correlation in the MBone multicast networ experimental measurements and markov chain models. Technical Report UM-CS-1995-115 (1995)

A Bandwidth Allocation Algorithm Based on Historical QoS Metric for Adaptive Video Streaming

Ling Guo[1], YuanChun Shi[1], and Wei Duan[2]

[1] Key Laboratory of Pervasive Computing, Tsinghua University
{Guoling02@mails.,shiyc@}tsinghua.edu.cn
[2] China United Telecommunications Corporation
duanw@chinaunicom.com.cn

Abstract. This paper introduces a dynamic bandwidth allocation algorithm in a video streaming multicast system. The approach is to introduce the vibration of received video quality into the QoS metric and make the receivers more negative in subscribing higher layers when bandwidth increases. A simulated annealing algorithm is applied in the server side to find the optimal allocation schema within the concurrent network situation at run time. Simulated experiments on NS-2 have been carried out to validate the algorithm. The result shows an improvement of 6.8 percents increase in received data rate and 6.0 percents decrease in data loss rate.

1 Introduction

The Internet has been experiencing explosive growth of audio and video streaming. Researchers have developed layered video multicast to provide video streaming to a large group of users. Various devices such as PDA, desktop, laptop, even mobile phones are widely used in various network conditions, as diversely as network conditions, such as LAN,ADSL,GPRS and etc. Layered video codec is used to suit the heterogeneous environment[1].

As we have mentioned above, the perceptual quality of a video is determined by many factors, such as image size and frame rate. Besides, Internet applications desire asymptotically stable flow controls that deliver packets to end users without much oscillation[2]. In a best-effort network, most of the video streaming systems use flow control mechanisms like AIMD to be fair to other applications. AIMD is known for drastically decrease of accept window when timeout or data loss occurs. The oscillation of bandwidth makes it even more difficult to get a stable video streaming over Internet. Recently, many approaches of congestion control have been raised to avoid fluctuation in video quality [2, 3]. Some others devised reschedule mechanisms of the buffered data to compensate the network delay or jitter [11]. In a layered video streaming system, comparing to network congestions, the bandwidth allocation mechanism have greater impact on the traffic. As far as we know, no work has been ever done in exploring the feasibility of improving the allocating mechanism to gain more stable video streaming.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3420, pp. 917–925, 2005.

The rest of this paper is organizing as the following. Part two introduces related works. The third part discusses the metric of continuous QoS. Then, the bandwidth allocation algorithm which uses simulated annealing is presented in part four. Then part five gives out the experimental result on NS-2 simulation to validate the allocation algorithm. In Part six, the paper ends with future work and conclusion.

2 Related Works

To transmit video packets over Internet, researchers have extensively explored many possibilities.

At first, sender-driven-congestion-control for adaptively encoded video was proposed and developed in unicast filed. The key point of the method is to adjust its encoding data rate in accordance with the network condition. [5,6,7]. The sender-driven algorithms are also extended to multicast, but as the video has only one layer, if the group has a low bandwidth node the whole multicast group will be impacted.

Receiver-driven adaptation algorithms were proposed after the emergence of layered video. The video source generates a fixed number of layers, and each user attempts to subscribe as many layers as possible. With the development of layered codec, it is possible to dynamically adjust the amount of layers as well as the data rate of each layer. Algorithms that take advantage of this improvement came into scene, such as SAMM (Source-Adaptive Multilayered Multicast Algorithms) [8]. Some of the layered algorithms set priority on layers and expect the network nodes selectively drop the higher layers when congestion occurs. Some other approaches just admit the concurrent infrastructural Internet as a QoS unaware network and try to compensate it in the application level. One of them is proposed by Liu [1, 4]. The paper describes a method to find the optimal allocation schema by a recursive function within an acceptable overhead. But the algorithm doesn't consider bandwidth vibration. It always try to make full use of the bandwidth.

As to perceptive QoS, many other aspects left unconsidered in the most QoS metrics, such as intra-frame synchronization and constant quality of video streams. Reza [9] managed to reveal the important impact of buffer and congestion on the perceptual QoS of video streaming. Reza points out that in order to gain smooth video, the buffer should always have enough data and can survive at least a TCP back-off in the near future. Therefore, when bandwidth increases, instead of simply joining a higher layer, the author proposes that the allocation algorithm should make sure that buffers should always have enough data to survive at least a TCP back-off. They also propose a method to allocate bandwidth among the active layers to prevent buffer underflow.

As it discusses above, the bandwidth allocation algorithm is designed to make full use of the available bandwidth, but they usually failed to consider some temporal requirements that intrinsically lay in streaming video. While researchers in congestion control reveal to us the relationship between jitter and bandwidth utilization, but the method of congestion control is not direct and may cause some side effects. Based on this, we propose a bandwidth allocation algorithm that integrates temporal characters.

3 Historical QoS Metric

3.1 QoS of Streaming Video

The quantitative metric of QoS is the basic of the allocation algorithm. In a dynamic environment such as the Internet, both user and service-provider factors are variable. The end-user wants to make full use of the bandwidth while the service provider pursues cost-effectiveness of bandwidth resources and the end-systems'QoS. It is a trade-off to decide which one to use. Some peer-to-peer systems use received data as QoS metrics. Nevertheless, multicast applications usually take the overall cost-effectiveness as the metrics. Usually, the computation resources and the output bandwidth of the server are limited. The server should be fair and efficient in allocation resources. One example of the cost-effective metric is the bandwidth utility [1,4]. In this paper, bandwidth utility is used as the basic QoS metric.

$$q = \frac{r}{b} \tag{1}$$

where r is the received data rate and b is the available bandwidth of the receiver.

3.2 Continuous Video Quality

Usually in video streaming systems, the end-user has a data consumption rate. The consumption rate is decided by the decoder and not necessarily constant. When the received data rate is lower than the consumption rate, a jitter will take place. In the best-effort Internet, the vibration on bandwidth happens constantly.

The continuality of video streaming also has much to do with history, which refers to the QoS performance in the past. Apparently, if the video quality increases drastically and decrease abruptly, it will cause discomfort change to users. In a multicast video streaming system, changes in QoS are mainly caused by the change of video layers. To introduce a history factor into QoS metric in such a system is our attempt.

3.3 Streaming System Model

Firstly, a system model is introduced as the basis for further discussion. It is real-time video streaming system using a TCP friendly application level multicast protocol. The server uses a layered codec to produce M layers. The cumulative data rate vector of M layers is $\rho^{'} = \{p_0, p_1, \ldots p_M\}$. There are N receivers $\{R_{\mathrm{i}} \mid 0 \le i < N\}$ connected in through heterogeneous networks. At time t, receiver R_{i} subscribes $c_i\,(t)$ layers and has a received data rate $\omega_i(t)$. Totally, a source data flow with the rate of $p_{c_i(t)+1}$ would be sending to user i at time t. In addition, in every time span Δt, each receiver R_{i} measures its own bandwidth $\Gamma_i(t)$ and reports it to the server.

Meanwhile, the server detect its bandwidth capability B(t) every Δt . Based on these reports, the server adjusts the allocation schema to get an optimal overall QoS:

$$M(t,L)=\sum_{i=0}^{n-1}\mathrm{Q}_i(t,l_i) \tag{2}$$

where $L=\{l_0,l_1...l_i...l_{n-1}\}$ is the vector of the new allocation schema and $\mathrm{Q}_i(t,l_i)$ is the estimated QoS of receiver R_{i} with l_i layers. The QoS metric will be discussed below. After the server finds out an optimal allocation schema, it will send notifications to receivers who need to drop or join a layer.

3.4 Bandwidth Burst

Consider the condition when a bandwidth burst happens, according to best-effort allocation algorithms, the user will subscribe a high layer immediately. After a while, the bandwidth drops to its average level then the user has to drop the highest one or two layers. This short-term subscribe-drop pair not only brings fluctuation in receiver's QoS but also intrigues buffer underflow and overflow at the receiver's side. What's more, during this subscription and drop process, the server sends out more data than what the receiver can receive. So sometimes when bandwidth bursts occurs, the best-effort bandwidth allocation algorithms will cause a short time of high QoS video and latter a jitter in client's side, we call this saw tooth in QoS, which is not desired by the receivers.

3.5 Historical QoS Metric

We introduce a historical factor into the QoS metric to avoid saw tooth. Suppose $q_i(t)$ is the QoS value of user i . We use bandwidth utility as the QoS metric as in (1). The historical QoS metric we defined is composed of a basic QoS metric and a historical effect factor:

$$Q_i\ (t,l_i)=q_i(t,l_i)*\chi(\eta_{\mathrm{i}}(t)) \tag{3}$$

Where $\eta_{\mathrm{i}}(t)=\dfrac{\Gamma_i(t)-\Gamma_i(t-1)}{\Gamma_i(t-1)}$ and $q_i(t,l_i)$ is the basic metric, $\eta_{\mathrm{i}}(t)$ is the bandwidth change variation. $\chi(\kappa)$ is the effect function of $\eta_{\mathrm{i}}(t)$. The goal of this function is to reduce the possibility of subscribing higher-level layer when the bandwidth increases. When $\eta_{\mathrm{i}}(t)>1$, the history effect factor of $\chi(\kappa)$ should be less than one. The higher $\eta_{\mathrm{i}}(t)$ is, the less the effect factor is.

Now with the new QoS metric, R_{i} has a much lower estimated QoS value when bandwidth bursts. The historical factor is the changing rate of the bandwidth.The more the bandwidth increases, the little the historical factor is. When bandwidth burst happens, $\eta_{\mathrm{i}}(t)$ in (3) is smaller than 1. It is more likely that the bandwidth allocation algorithm will choose not to add a layer. If in the next time span Δt, bandwidth drops, the receiver will not change the subscription layer. If bandwidth does not drop, it is likely that it is not a burst. Then in the next Δt, the historical effect factor would be one and likely get the opportunity to add a layer. Therefore, and the historical factor makes the allocation algorithm more stable to avoid some short-term subscribe-drop pairs.

As mentioned above, the effect function $\chi(\kappa)$ in (3) should increase slowly when $\kappa > 1$ and remain "1" when $\kappa < 1$. We found $\chi(\kappa) = \begin{cases} e^{-a(\kappa-1)} (\kappa > 1) \\ 1 (\kappa < 1) \end{cases}$ is a simple function fulfilling the requirements:

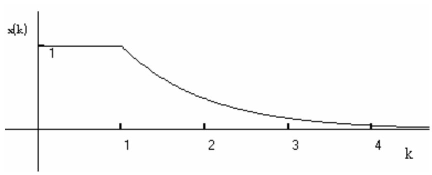


Fig. 1. Effect function of Vibration on QoS

According to (1) and (3), the quantitative QoS metric of user i at time t is

$$\begin{cases} Q_i(t) = \dfrac{r_i(t)}{b_i(t)} \left(\dfrac{\Gamma_i(t)}{\Gamma_i(t-1)} \le 1 \right) \\ Q_i(t) = \dfrac{r_i(t)}{b_i(t)} * e^{-a * \frac{\Gamma_i(t) - \Gamma_i(t-1)}{\Gamma_i(t-1)}} \left(\dfrac{\Gamma_i(t)}{\Gamma_i(t-1)} > 1 \right) \end{cases} \tag{4}$$

It means that when bandwidth does not increase, the QoS equals the bandwidth utility; if the bandwidth increases, the historical factor is less than 1 and the QoS is less than the bandwidth utility.

4 Dynamic Allocation Algorithm

In dynamic allocation algorithm, we use the QoS metric in (4) to measure the QoS. The problem is to find out a subscription schema to get the maximal overall QoS. For that purpose, a simulated annealing algorithm is used to search for optimal allocation schema. According to (4), the optimization goal of the simulated anneal algorithm is:

$$\sum_{i \in \{\frac{\Gamma_i(t)}{\Gamma_i(t-1)} \le 1\}} \frac{r_i(t)}{b_i(t)} + \sum_{j \in \{\frac{\Gamma_i(t)}{\Gamma_i(t-1)} > 1\}} \frac{r_j(t)}{b_j(t)} * e^{-\frac{\Gamma_i(t) - \Gamma_i(t-1)}{\Gamma_i(t-1)}} \tag{5}$$

The searching space S is the entire possible subscription schema in the current network condition:

$$S = \{C_t(c_0(t)...c_i(t)...c_{n-1}(t)) \mid \sum_{i=1}^{n} p_{c_i(t)} \le B(t)\} \tag{6}$$

While, in this algorithm, we have a constraint on the server side bandwidth:

$$\sum_{i=0}^{n-1} r_i(t) \le B(t) \tag{7}$$

The method to find the next point is important to the efficiency of the algorithm. It can choose a new point as well as examined points. In the experiment, we found that if the possibility of choosing a new point is equal to the possibility of choosing an old one, the algorithm would spent a lot of time hovering between several points and it needs a large MARKOV value to get the optimal point. Therefore, we set different possibility value at new points and old points, the algorithm can get to the optimal very fast.

5 Simulation on NS-2

NS-2 simulation is carried out to validate the algorithm. In the experiment, all the links are duplex links with the delay of 2ms. Queues with FIFO drop-tail and maximum delay of 0.5 sec are used. The maximal package size is set to 1000 bytes. To simulate the layered video streaming, we use a video trace file of a temporal scalable video with three layers. The data rates of three layers are [210.78, 110.92, 60.575kbps]. The transport protocol is UDP.

The simulation topology is like the figure 2 as below. In which, node2/4/5/6/7 are all receivers of the layered video streaming. A FTP flow is set between node0 and node3. The allocation algorithm executes on the server side every 4 seconds. All the simulations run for 500 seconds to get stable results.

The simulated annealing algorithm is used to search for an optimal allocation schema at run-time. In order to demonstrate the effect of the historical QoS factor, we carry out two comparative experiments, A and B. They are all the same except that in A, a historical effect function is used.

In order to get the available bandwidth, we use Packet Pair algorithm [10, 12].

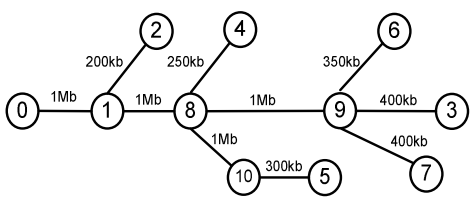


Fig. 2. Topology of the simulation scenario

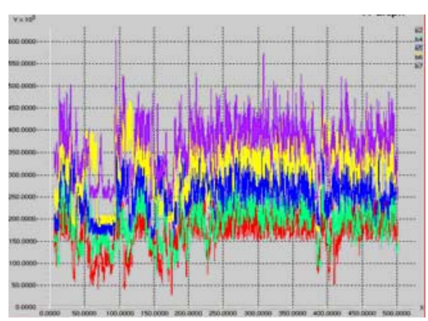
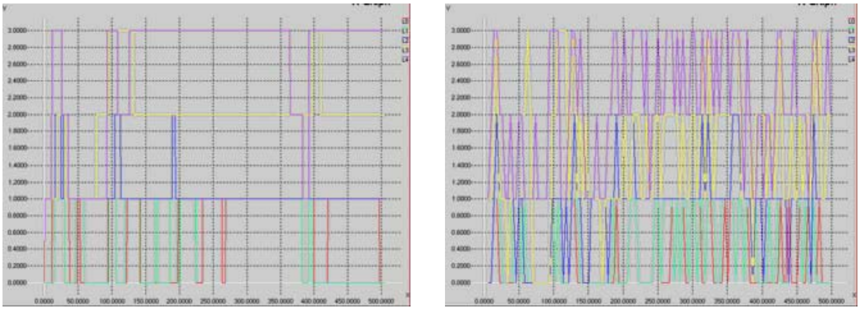


Fig. 3. Available bandwidth of experiment A



a) **b)**

Fig. 4. Subscription Record of Experiment A and B

In Fig. 4, B has much denser fluctuations than A has. That is because in A, the bandwidth incensement means lower QoS than B does. The historical effect factor is usually lower than "1" when bandwidth increases. Therefore, the possibility for the allocation algorithm to add the layers is lower than B. Fig. 3 is the estimated available bandwidth in experiment A. In Fig. 4, the bandwidth reaches a stable status after a period of adjustment. In Fig. 4 A), subscribed layers increases and decreases as the bandwidth does. For example, in time 0~50, the bandwidth increases and hold for a while. In 30s, bandwidth decreases. Correspondingly, in Fig. 4 A) all the four receivers add a layer in the time span. Then decrease after 30s.

From the statistics in Table 1, A have higher average successfully received data rates than B. In addition, generally data loss ratio is lower in A than in B. Except that in B, Node2 has a lower loss ratio and a slightly higher data rate. Node2 is connected with Node1 through a connection of 200bps, which is lower than the data rate of the first layer. Therefore, the feasible choice of Node2 is to subscribe the first layer or to subscribe nothing. The effect of the historical factor is to reduce the possibility of adding a layer when the bandwidth increases. Moreover, for Node2, sometimes, the bandwidth utility is zero and the historical factor multiple does not have any influence to it.

Table 1. Statistics of experiment result A and B. The shadowed column is the statistic of A and the other is B's. Sending rate is calculated in the server side according to the subscribed layers. Data Rate is calculated according to the successfully received data packet in each node

Node	Bandwidth(kbps)	Sending Rate(kbps)		Loss Rate(%)		Data Rate(kbps)	
N2	200	143.748	145.408	12.508	9.2	125.779	132.030
N4	250	209.043	202.402	3.252	3.583	202.245	195.150
N5	300	275.264	242.915	6.287	8.349	257.958	222.634
N6	350	342.639	308.413	5.139	6.57	325.031	288.150
N7	400	388.490	365.277	3.998	4.473	372.958	348.938

In above, the result shows that historical effect factor improves the video streaming by increasing the data rate by 6.89 percents and decreasing the loss rate by 6.07 percents.

6 Future Work and Conclusion

In this paper, we introduce a historical factor into QoS Metic to get a smoother video. The allocation algorithm is more conservative and helps the multicast video streaming system to maximize the overall QoS through the optimizing of subscribing schema. Simulated annealing algorithm is used to get an optimal allocation schema at run-time. Experiments on NS-2 are conducted to demonstrate the algorithm. Experiment results show that in most cases, the historical effect factor can avoid frequent fluctuation of subscribed layers and improve video streaming QoS.

Further work would include a real implementation with layered codec and heterogeneous network condition. Besides, mobile network connections are not stable and the capability of the mobile device varies. Layered video streaming on mobile network is also widely discussed. The algorithm would be extended to mobile network scenarios.

References

[1] J. Liu, B. Li, and Y.-Q. Zhang, Adaptive Video Multicast over the Internet, IEEE Multimedia, Vol. 10, No. 1, pp. 22-31, January/February 2003.

[2] Min Dai, Dmitri Loguinov, "Analysis of Rate Distortion Functions and Congestion Control in Scalable Internet Video Streaming", Proceedings of the 13th international workshop on Network and operating systems support for digital audio and video, June 2003

[3] R. Johari and D. Tan, End-to-End Congestion Control for the Internet: Delays and Stability," IEEE/ACM Transactions on Networking, vol. 9, no. 6, December 2001.

[4] Jiangchuan Liu, Bo Li and Ya-Qin Zhang, "An End-to-End Adaptation Protocol for Layered Video Multicast Using Optimal Rate Allocation", in IEEE Transaction on Multimedia Vol 6, No. 1, February 2004.

[5] M. Gilge and R. Gusella, "Motion video coding for packet-switching networks——an integrated approach", in SPIE Conf. Visual Communications and Image Processing, Nov. 1991.

[6] Y. Omori, T. Suda, G. Lin, and Y. Kosugi, "Feedback-based congestion control for VBR video in ATM networks", in Proc. 6th Int. Workshop Packet Video, 1994.

[7] C.M. Sharon, M.Devetsikiotis, I. Lambadaris and A.R. Kaye, "Rate control of VBR H.261 video on frame relay networks", in Proc. Int. Conf. Communications(ICC), 1995.

[8] Brett J. Vickers, Cello Albuquerque, and Tatsuya Suda, "Source-Adaptive Multilayered Multicast Algorithms for Real-Time Video Distribution" in IEEE/ACM Transaction on Networking, Vol, 8, NO. 6, Dec. 2000.

[9] Reza Rejaie, and Mark Handley, "Quality adaptation for congestion controlled video playback over the Internet", in Proc. IEEE Infocom, March 1999.

[10] Gili Manzanaro J., Janez Escalada, L., Hernandez Lioreda, M., Szymanski,M. “Subjective image quality assessment and prediction in digital video communications. COST 212 HUFIS Report, 1991.
[11] Laoutaris N., Stavrakakis I, “Intrastream Synchronization for Continuous Media Streams: A Survey of Playout Schedulers”, IEEE Transaction on Network, May-June 2002, Vol, 16, Issue, 3, Pages, 30 - 40
[12] Arnaud Legout and Ernst W. Biersack. PLM: Fast Convergence for Cumulative Layered Multicast Transmission Schemes. In Proc. of ACM SIGMETRICS'2000, pages 13--22, Santa Clara, CA, USA, June 2000.

Author Index

Abdalla, H. II-66
Achir, Mounir II-442
Adalier, Ahmet I-842
Adamovic, Ljiljana II-335
Afandi, Raja II-662
Åhlund, Christer I-204
Ahmad, Iftekhar I-117
Ahn, Gaeil II-689
Ahn, Seongjin I-818
Ahn, Young-Kyu I-421
Ai, Jing I-467
Akinlar, Cuneyt II-156
Altenbernd, Peter II-1071
Altunbasak, Hayriye II-699
Amirat, Y. II-164
Amvame-Nze, G. II-66
An, Sunshin II-91, II-488
Anelli, Pascal I-84, II-275
Anh, Le Tuan II-141
Asatani, Koichi II-859
Assi, Chadi I-34
Aswathanarayaniah, Srinivas I-117

Badonnel, Remi II-83
Bahng, Seungjae I-153
Bai, Yan I-654
Bambos, Nicholas I-849
Barreto, P.S. II-66
Bartusek, Karel II-384
Basney, Jim II-662
Bassil, Carole II-810
Bestak, Robert I-100
Bienkowski, Marcin I-413
Bleul, Holger II-606
Blough, Douglas M. I-802
Bobek, Andreas I-430
Bodendorf, Freimut I-690
Bohn, Hendrik I-430
Bölöni, Ladislau I-467
Bonilla, Rafael II-662
Boreli, Roksana II-192, II-617
Bossi, Stefano I-662
Bouras, Christos I-766
Boussif, Malek I-388
Branch, Joel W. I-438
Brännström, Robert I-204
Brinkmann, André I-413, II-800
Bruneel, Herwig I-620, I-892
Brunstrom, Anna I-247, I-774
Buchner, Christian I-882
Byun, Taeyoung I-459

Cahit, Ibrahim I-842
Cai, Liang II-819
Cai, Zhiping II-746
Cap, Clemens II-99
Caraballo Moreno, Joaquín II-625
Cariou, Laurent II-8
Cecconi, Luca I-92
Cha, Si-Ho I-794
Chae, Donghyun II-488
Chan, Agnes H. II-827
Chan, Chia-Tai II-728
Chang, Ray-I II-835
Chen, Chienhua II-34
Chen, Chun II-819
Chen, Gilbert G. I-438
Chen, Ing-Yi I-186
Chen, Jenhui II-58
Chen, Maoke I-508
Chen, Yaw-Chung II-728
Chen, Yue I-19
Chen, Yun-Lung II-34
Cheng, Liang I-561, I-662
Chiou, Chung-Ching II-58
Cho, Byung-Lok I-421
Cho, Dong-Hoon I-358
Cho, Jinsung I-374
Cho, Kuk-Hyun I-794
Cho, Sarm-Goo I-421
Choi, Byoung-Sun II-125
Choi, Cheon Won I-397
Choi, Dong-You II-904, II-920
Choi, Jin-Ghoo II-258
Choi, Jin-Hee II-258, II-1055
Choi, Jun Kyun I-342
Choi, Seung Sik II-772
Choi, Sunwoong II-1080

Choi, WoongChul I-794
Chu, Chih-Chun II-835
Chu, Yul I-654
Chung, Jinwook I-818
Čičić, Tarik II-173, II-1097
Collier, Martin II-335
Cousin, Bernard II-844
Cui, Yong II-202, II-480
Cusani, Roberto I-92
Cvrk, Lubomir I-27, II-673

Dai, Kui II-1114
Dai, Qin-yun II-353
Davik, Fredrik II-551
Davoli, Renzo I-527
de Carvalho, H.P. II-66
de Castro, Marcel C. II-116
Delicato, Flávia I-569
Deng, Ke II-26
de Rezende, José Ferreira I-569
de Siqueira, Marcos A. II-116
De Vuyst, Stijn I-892
Dhanakoti, Niranjan II-42
Dhinakaran, Beatrice Cynthia I I-125
Diaz, Michel I-125
Ding, Le II-401, II-928
Dinh Vo, Nhat Minh II-327
Ditze, Michael II-1071
Doğançay, Kutluyıl II-531
Domingo-Pascual, Jordi II-266
Dou, Wenhua I-318
Dragios, Nikolaos D. II-634
Dreibholz, Thomas II-564
Drissi, Jawad I-169
Duan, Wei I-917
Dutt, Nikil I-662

El Abdouni Khayari, Rachid I-535
Elst, Günter I-286
El Zarki, Magda I-662
Eyrich, Michael II-192

Fabini, Joachim II-496
Fathi, Hanane I-366
Fdida, Serge II-275
Feng, Dengguo II-964, II-980
Ferreira, Adrian Carlos I-449
Festin, Cedric Angelo M. I-518
Festor, Olivier II-83
Fiems, Dieter I-892
Figueiredo, Carlos Mauricio S. I-585
Figueiredo, Fabricio L. II-116
Finger, Adolf I-286
Firkin, Eric C. II-575
Fitzek, Frank I-366
Flores Lucio, Gilberto I-635
Fort, David II-844
Fourmaux, Olivier II-625
Francis, J. Charles I-382
Frattasi, Simone I-366
Freire, Mário M. I-44
Fritsch, Lothar II-1130
Fuin, David I-672

Galetzka, Michael I-286
Galmés, Sebastià II-585
Gan, Choon Hean I-239
Gao, Bo II-1063
Gao, Wen I-865
Garcia, Eric I-672
Garcia, Johan I-247
Garcia, Mario Hernan Castaneda I-231
Gescheidtova, Eva II-384
Giles, Stephen I-239
Gineste, Mathieu I-144
Gjessing, Stein II-173, II-551, II-1097
Göger, Gernot I-52
Golatowski, Frank I-430
Gomes, Cristiana I-60
Gonzáles-Sánchez, José Luis II-266
Gopalan, Srividya II-42
Göschka, Karl Michael I-680
Grimminger, Jochen II-699
Grinnemo, Karl-Johan I-774
Grolmusz, Vince II-454
Gruber, Claus G. I-133
Gu, Huaxi I-826
Gu, RongJie I-740
Guang, Cheng I-758
Guette, Gilles II-844
Guo, Chengcheng I-740
Guo, Huaqun II-50, II-754
Guo, Lei I-68
Guo, Ling I-917
Guyennet, Hervé I-672

Ha, Jun I-397
Ha, Nam-koo I-731, II-210
Habib, Eduardo I-449

Hafid, Abdelhakim I-169
Hahm, Hyung-Seok II-662
Han, Dong Hwan I-161
Han, Ki-Jun I-358, I-459, I-731, I-810, II-210
Han, Ningning II-184
Han, Wenbao II-242
Hansen, Audun Fosselie II-173, II-1097
Harivelo, Fanilo I-84
He, Liwen II-463
He, Simin I-865
Hegland, Anne Marie II-471
Heidebuer, Michael II-800
Helard, Jean-Francois II-8
Henning, Ian D. I-635
Herborn, Stephen II-617
Hirotsu, Toshio II-284
Hladká, Eva II-876
Ho, Chen-Shie I-186
Hoceini, S. II-164
Holub, Petr II-876
Hong, Choong Seon II-141
Hong, Feng I-826
Hong, Jinkeun II-953
Hong, Kyung-Dong I-178
Hong, Seok-Hoon I-421
Hong, Sung Je II-884
Hou, Jia I-406, II-1
Hsu, Chih-Shun I-577
Hu, Tim Hsin-Ting II-617
Hu, Xiu-lin II-353
Huda, Md. Nurul II-218
Huo, Wei I-34
Hur, Sun I-194
Huth, Hans-Peter II-699
Hwang, Jae-Hyun II-1055
Hwang, Jin-Ho I-326, II-1138
Hwang, Sungho I-459

Iannello, G. II-718
Imai, Hideki II-944
Imase, Makoto I-749
Isailă, Florin II-762
Ishikawa, Norihiro II-892
Itano, Kozo II-284
Ito, Mabo Robert I-654

Jameel, Hassan I-1
Jang, Yeong M. II-18
Jenkac, Hrvoje I-882
Jeon, Cheol Y. II-18
Ji, Zhongheng I-334
Jian, Gong I-758
Jie, Yang I-714
Jo, Seung-Hwan II-234, II-1122
Jordan, Norbert II-496
Jun, Kyungkoo II-543
Jung, Won-Do II-234, II-1122

Kahng, Sungtek II-772
Kaleshi, Dritan II-1012
Kalim, Umar I-1
Kamioka, Eiji II-218
Kämper, Guido II-1071
Kampichler, Wolfgang I-680
Kamruzzaman, Joarder I-117
Kang, Euisuk II-297
Kang, Guochang I-826
Kang, Ho-Seok II-868
Kang, Sangwook II-488
Kang, Seokhoon II-543
Karimou, Djibo II-107
Kato, Kazuhiko II-284
Katsuno, Satoshi I-9
Katz, Marcos I-366
Kellerer, Wolfgang II-781
Kesselman, Alex II-133
Khanvilkar, Shashank II-597
Khokhar, Ashfaq II-597
Khurana, Himanshu II-662
Kikuchi, Shinji I-544
Kim, Bara I-161
Kim, Dae-Young I-374
Kim, Dongkyun I-594
Kim, Heung-Nam II-234, II-1122
Kim, Jae-Hyun I-258
Kim, Jeong Su II-1
Kim, Jin Sang I-901
Kim, Jin-Nyun I-810
Kim, Jong II-884
Kim, JongWon II-1003
Kim, Joo-Ho II-504
Kim, Ki-Hyung II-234, II-1106, II-1122
Kim, Kiseon I-153, II-936
Kim, Kiyoung II-689
Kim, Kwan-Ho I-421
Kim, Kyung-Jun I-810, II-210
Kim, Min-Su I-358, I-459

Kim, Namgi II-1080
Kim, Pyung Soo I-214
Kim, Seungcheon I-483
Kim, Sung-Un I-178, I-326, II-1138
Kim, Won II-1138
Kim, Young Soo I-901
Kim, Young-Bu I-178
Kim, Yun Bae I-194
Kim, Yunkuk II-488
Kinoshita, Kazuhiko II-521
Király, Zoltán II-454
Kitatsuji, Yoshiori I-9
Klobedanz, Kay II-1071
Ko, Kwang O. I-901
Kobara, Kazukuni II-944
Koide, Hiroshi I-9
Komosny, Dan II-673
Koo, Insoo I-153, II-936
Koo, Jahwan I-818
Koodli, Rajeev II-361
Korkmaz, Turgay I-318
Korzeniowski, Miroslaw I-413
Kowalik, Karol II-335
Krasser, Sven II-699
Krishnamurthy, Vikram II-912
Kubánek, David II-410 , II-417
Kubasek, Radek II-384
Kumar, Mukesh I-706
Kumar, Praveen II-42
Kumar, Sanjeev I-834, II-997
Kuo, Sy-Yen I-186
Kure, Øivind II-471
Kurth, Christoph I-680
Kvalbein, Amund II-551, II-1097
Kwak, Deuk-Whee II-1003

Lamotte, Wim I-268
Lattenberg, Ivo II-410
Lee, Byeong-jik I-731, II-210
Lee, Chun-Jai I-178
Lee, Chun-Liang II-728
Lee, Gyu Myoung I-342
Lee, Heesang I-194
Lee, Hyun-Jin I-258
Lee, Jae-Dong I-178, I-326
Lee, Jae-Kwang I-628, II-125
Lee, Jihoon II-343
Lee, Jong Hyuk II-772
Lee, Jun-Won I-326, II-1138
Lee, Mike Myung-Ok I-421
Lee, Moon Ho I-406, II-1
Lee, Seoung-Hyeon I-628
Lee, SookHeon II-297
Lee, Suk-Jin I-178
Lee, Sungyoung I-1, I-698, I-714, II-327
Lee, Won-Goo I-628
Lee, Young-koo I-698
Lei, Shu I-714
Leinmüller, Tim II-192
Li, Dequan II-980
Li, Dong II-184
Li, Guangsong II-242
Li, Lei I-350
Li, Lemin I-68
Li, Minglu I-19
Li, Xing I-508
Li, Ying I-19
Li, Yuliang II-1012
Li, Zhengbin II-149
Liao, Chih-Pin I-577
Liao, Jia Jia II-149
Liebl, Günther I-882
Lilith, Nimrod II-531
Lin, Dongdai II-964
Lin, Xiaokang II-226
Liu, Hui-shan II-480
Liu, Fang II-1114
Liu, Xianghui II-746
Liu, Yi II-184
Liu, Zengji I-826
Lochin, Emmanuel II-275
Loeser, Chris II-800
Lopez-Soler, Juan M. I-909
Lorenz, Pascal I-44, I-646
Loureiro, Antonio Alfredo F. I-449, I-585
Lu, Xi-Cheng I-554, II-433, II-793
Luo, Ming II-26, II-401
Luo, Wen II-75
Lysne, Olav II-173

Ma, Huiye II-1063
Ma, Jun II-1114
Ma, Yongquan II-643
Maach, Abdelilah I-169
Magoni, Damien I-646
Malpohl, Guido II-762
Mammeri, Zoubir I-277
Mansour, Yishay II-133
Mao, Guoqiang I-492
Martin, Steven I-296

Martins, Jose A. II-116
Masuyama, Hiroshi I-221
Mateus, Geraldo Robson I-60, I-475
Matsutani, Hiroki II-361
Matyska, Ludek II-876
McMahon, Margaret M. II-575
Mellouk, A. II-164
Menezes, Gustavo Campos I-475
Minet, Pascale I-296
Mitrou, Nikolas M. II-634
Moeneclaey, Marc I-620
Mogensen, Preben E. I-388
Moh, Sangman II-369
Mohapatra, Shivajit I-662
Molnar, Karol I-27
Monsieurs, Patrick I-268
Moon, Bo-Seok II-504
Morabito, Giacomo II-1023
Munoz, Alvaro I-834
Munro, Alistar II-1012
Murai, Jun II-361
Murakami, Kazuya I-221
Murakami, Koso II-307, II-521
Myoupo, Jean Frédéric II-107

Nagamalai, Dhinaharan I-628, II-125
Nakamura, Eduardo Freire I-585
Nakamura, Fabíola Guerra I-475
Ngoh, Lek Heng II-50, II-754
Nguyen, Ngoc Chi II-327
Nilsson, Anders II-361
Nogueira, António I-603
Noh, Jae-hwan I-731
Noh, Seung J. I-194
Noh, Sun-Kuk II-904, II-920
Noh, Wonjong II-91, II-343

Oh, Hui-Myung I-421
Oh, Moon-Kyun I-178
Oh, Sung-Min I-258
Ohmoto, Ryutaro I-76
Ohsaki, Hiroyuki I-749
Oie, Yuji I-9
Oliveira, J.S.S. II-66
Oliveira, José Luis I-603
Oliveira, Leonardo B. I-449
Orhan, Orhan I-413
Ouvry, Laurent II-442
Owen, Henry L. I-802, II-699

Palazzo, Sergio II-1023
Palmieri, Francesco I-306
Pantò, Antonio II-1023
Park, Chang-kyun II-904
Park, Chul Geun I-161
Park, Jae Keun II-884
Park, Jin Kyung I-397
Park, Jong-Seung II-772
Park, Ju Yong II-1
Park, Jun-Sung II-234, II-1122
Park, Myong-Soon II-297, II-504
Park, Seung-Min II-234, II-1122
Park, Soohong I-214
Park, Sung Han II-1031
Peng, Wei II-793
Perera, Eranga II-192
Pescapé, A. II-718
Ping, Xiaohui II-184
Pinho, Teresa I-603
Pirmez, Luci I-569
Poropatich, Alexander II-496
Prasad, Ramjee I-366
Primpas, Dimitris I-766
Protti, Fabio I-569
Puigjaner, Ramon II-585
Puttini, R. II-66

Qiu, Zhiliang I-826
Qu, Haipeng II-964, II-980
Quintão, Frederico Paiva I-475

Radusinovic, Igor I-857
Radzik, Tomasz II-250
Rakotoarivelo, Thierry I-125
Ramos-Muñoz, Juan J. I-909
Rathgeb, Erwin P. II-564, II-606
Ravelomanana, Vlady I-109
Razzano, Giuseppe I-92
Reed, Martin J. I-635
Rhee, Kyung Hyune II-972
Rhee, Yoon-Jung II-852
Rocha, Flavia, M. F. II-116
Rodošek, Robert II-318
Rodrigues, Joel J.P.C. I-44
Ross, Kevin I-849
Rossi, Davide II-737
Rouhana, Nicolas II-810
Rudskoy, A. II-681
Rust, Luiz I-569
Ryu, Jung-Pil I-459

Sajjad, Ali I-1
Salvador, Paulo I-603
Sasama, Toshihiko I-221
Sathiaseelan, Arjuna II-250
Savaş, E. II-707
Schattkowsky, Tim II-653
Scherner, Tobias II-1130
Schimmel, Jiri II-425
Schmidt, Thomas C. II-1039
Schneider, Johannes I-382
Schollmeier, Rüdiger II-781
Schomaker, Gunnar II-800
Senac, Patrick I-125, I-144
Seneviratne, Aruna I-125, II-192, II-617
Seo, Hyun-Gon II-234, II-1106, II-1122
Serhrouchni, Ahmed II-810
Shami, Abdallah I-34
Shankar, Udaya A. II-156
Shao, Ziyu II-149
Sharma, Navin Kumar I-706
Shemanin, Y.A. II-681
Shen, Hong I-722, II-989
Shen, Lin II-202
Sheu, Jang-Ping I-577
Shi, Xiaolei I-231
Shi, Yi I-784
Shi, YuanChun I-917
Shim, Young-Chul II-868
Shin, Chang-Min II-234, II-1122
Shin, Jitae I-818
Shin, Seokjoo I-153
Shin, SeongHan II-944
Shin, Woo Cheol I-397
Shinjo, Yasushi II-284
Shinohara, Yusuke II-307
Siddiqui, F. II-1047
Silva, C.V. II-66
Šimák, Boris II-392
Simonis, Helmut I-611
Slagell, Adam II-662
Smekal, Zdenek II-384
Soares, A.M. II-66
Sokol, Joachim II-699
Song, Jung-Hoon I-358
Sørensen, Søren-Aksel I-518
Sørensen, Troels B. I-388
Soy, Mustafa I-690
Speicher, Sebastian II-99
Spilling, Pål II-471
Spinnler, Bernhard I-52
Sponar, Radek II-417
Sridhar, V. II-42
State, Radu II-83
Stathopoulos, Vassilios M. II-634
Steyaert, Bart I-620
Stockhammer, Thomas I-882
Stromberg, Guido I-231
Su, Purui II-964, II-980
Suh, Doug Young I-901
Sun, Shutao I-865
Sunar, Berk II-707
Sung, Mee Young II-772
Suzuki, Hideharu II-892
Suzuki, Shinichi II-284
Sysel, Petr II-425
Szymanski, Boleslaw K. I-438

Tak, Sungwoo II-1088
Takahashi, Takeshi II-859
Takeyama, Akira I-544
Tan, Guozhen II-184
Tarlano, Anthony II-781
Tellini, Simone I-527
Teyeb, Oumer M. I-388
Teyssié, Cédric I-277
Tian, Hui I-722
Tode, Hideki II-307
Tominaga, Hideyoshi II-859
Tong, Ting II-149
Tsuru, Masato I-9
Turgut, Damla I-467
Turrini, Elisa II-737
Tüxen, Michael II-564

Ueno, Hidetoshi II-892
Uwano, Shuta I-76

Valadas, Rui I-603
Veiga, Hélder I-603
Veljovic, Zoran I-857
Venkatasubramanian, Nalini I-662
Ventre, G. II-718
Vieira, João Chambel II-266
Vilaça, Marcos Aurélio I-449
Vlček, Miroslav II-392
Vodisek, Mario II-800
Vollero, L. II-718
Vrba, Kamil II-410, II-417
Vrba, Vit I-27

Wählisch, Matthias II-1039
Wakikawa, Ryuji II-361
Walraevens, Joris I-620
Wang, Dongsheng II-643
Wang, Kun I-826
Wang, Pi-Chung II-728
Wang, Xi II-377
Wang, Zhiying II-1114
Wang, Ziyu II-149
Wei, Ding I-758
Wei, Wei I-334
Weihs, Manfred I-873
Wigard, Jeroen I-388
Wijnants, Maarten I-268
Winjum, Eli II-471
Wolf, Michael II-192
Wong, Duncan S. II-827
Wong, Hao Chi I-449
Wong, Wai Choong II-50, II-754
Wu, Jianping II-75
Wu, Shih-Lin II-58
Wu, Ya-feng II-377

Xia, Quanshi I-500, I-611
Xiaoling, Wu I-714
Xu, Anshi II-149
Xu, Ke II-75, II-202, II-480
Xu, Ming-wei II-202, II-480
Xu, Yin-long II-377
Xuan, Hung Le I-698

Yamada, Shigeki II-218
Yamagaki, Norio II-307
Yamai, Nariyoshi II-521
Yamazaki, Katsuyuki I-9
Yan, PuLiu I-740
Yang, Jeongrok II-936
Yang, Jong-Phil II-972
Yang, Junjie I-334
Yang, Seung Jei II-1031
Yang, Weilai I-802
Yang, Xiaohu II-819
Yang, Xinyu I-784
Yang, Yuhang II-1063
Yaprak, E. II-1047
Ye, Qing I-561
Yin, Jianping II-746
Yin, Qinye II-26, II-401, II-928
Yin, Shouyi II-226
Yokoyama, Ken I-544
Yoo, Chuck II-258, II-1055
Yoo, Gi-Chul I-594
Yoo, See-hwan II-1055
Yoon, Hyunsoo II-1080
Yoshida, Shinpei I-749
Yu, Fei II-912
Yu, Hongfang I-68
Yu, Hong-yi II-353

Zaborovskii, V.S. II-681
Zahradnik, Pavel II-392
Zaslavsky, Arkady I-204, I-239
Zeadally, S. II-1047
Zeman, Vaclav II-673
Zeng, Guo-kai II-377
Zeng, Ming I-784
Zeng, Qingji I-334
Zeng, Yanxing II-928
Zhang, Changyong II-318
Zhang, Huimin I-350
Zhang, Jianguo II-928
Zhang, Lin I-350
Zhang, Xiao-Zhe II-433
Zhang, Yiwen II-26, II-401
Zhang, Zonghua II-989
Zhao, Jun II-353
Zhao, Rui I-784
Zhao, Wentao II-746
Zheng, Qianbing II-793
Zheng, Yanfeng I-865
Zheng, Yanxing I-318
Zhu, Feng II-827
Zhu, Ke I-554
Zhu, Pei-Dong I-554, II-433, II-793
Zhu, Qiaoming I-19
Zöls, Stefan II-781
Zou, Tao I-740