Sergey Balandin
Yevgeni Koucheryavy
Honglin Hu (Eds.)

# Smart Spaces and Next Generation Wired/Wireless Networking

LNCS 6869

11th International Conference, NEW2AN 2011, and
4th Conference on Smart Spaces, ruSMART 2011
St. Petersburg, Russia, August 2011, Proceedings

Springer

Lecture Notes in Computer Science 6869

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

Sergey Balandin Yevgeni Koucheryavy
Honglin Hu (Eds.)

# Smart Spaces and Next Generation Wired/Wireless Networking

11th International Conference, NEW2AN 2011, and
4th Conference on Smart Spaces, ruSMART 2011
St. Petersburg, Russia, August 22-25, 2011
Proceedings

Springer

Volume Editors

Sergey Balandin
FRUCT Oy / Tampere University of Technology
Department of Communications Engineering
Korkeakoulunkatu 1, 33720 Tampere, Finland
E-mail: sergey.balandin@fruct.org

Yevgeni Koucheryavy
Tampere University of Technology
Department of Communications Engineering
Korkeakoulunkatu 1, 33720 Tampere, Finland
E-mail: yk@cs.tut.fi

Honglin Hu
Shanghai Research Center for Wireless Communications (WiCo)
6th Floor, Building 1, Lane 280, Linhong Road
Changning District, Shanghai 200335, China
E-mail: hlhu@ieee.org

ISSN 0302-9743 e-ISSN 1611-3349
ISBN 978-3-642-22874-2 e-ISBN 978-3-642-22875-9
DOI 10.1007/978-3-642-22875-9
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2011933582

CR Subject Classification (1998): C.2, B.8, C.4, D.2, K.6, D.4.6, K.6.5

LNCS Sublibrary: SL 5 – Computer Communication Networks and Telecommunications



*Typesetting:* Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

# Preface

We welcome you to the joint proceedings of the 11th NEW2AN (Next Generation Teletraffic and Wired/Wireless Advanced Networking) and 4th ruSMART (Are You Smart) conferences held in St. Petersburg, Russia, during August 22–25, 2011.

Originally the NEW2AN conference was launched by ITC (International Teletraffic Congress) in St. Petersburg in June 1993 as an ITC-Sponsored Regional International Teletraffic Seminar. The first implementation was entitled "Traffic Management and Routing in SDH Networks" and held by R&D LONIIS. In 2002 the event received its current name, NEW2AN. In 2008 NEW2AN received a new counterpart in smart spaces, ruSMART, hence boosting interaction between researchers, practitioners and engineers from different areas of ICT. NEW2AN/ruSMART are established conferences with a unique cross-disciplinary mix of telecommunications science in Russia. NEW2AN/ruSMART have always featured outstanding keynotes from universities and companies across Europe, USA and Russia. This year NEW2AN / ruSMART were co-located with the 11th International Conference on Intelligent Transportation Systems (ITS) Telecommunications, an international forum on recent advances in information and communication technologies for safe, efficient and green transport.

The 11th NEW2AN technical program addressed various aspects of next-generation network architectures. New and innovative developments for enhanced signaling protocols, QoS mechanisms, cross-layer optimization and traffic characterization were also addressed. In particular, issues of QoE in wireless and IP-based multiservice networks were dealt with, as well as financial aspects of future networks. It is also worth mentioning the emphasis placed on wireless networks, including, but not limited to, cellular networks, wireless local area networks, personal area networks, mobile ad hoc networks, and sensor networks.

The 4th Conference on Smart Spaces (ruSMART 2011) provided a forum for academic and industrial researchers to discuss new ideas and trends in the emerging area of smart spaces that create new opportunities for fully-customized applications and services for users. The conference brought together leading experts from top affiliations around the world. This year there was active participation by industrial world-leader companies and particularly strong interest from attendees representing Russian R&D centers, which have a good reputation for high-quality research and business in innovative service creation and applications development.

This year the technical program of NEW2AN/ruSMART/ITST benefited from joint keynote speakers from European, Russian and USA universities, companies and authorities.

We wish to thank the Technical Program Committee members of both conferences and associated reviewers for their hard work and important contribution.

This year the conferences were organized in cooperation with the FRUCT Program, ITC (International Teletraffic Congress), IEEE, Tampere University of Technology, Popov Society, and supported by NOKIA. The support of these organizations is gratefully acknowledged.

Finally, we wish to thank the many people who contributed to the organization. In particular, Jakub Jakubiak (TUT, Finland) carried a substantial load of submission and review, website maintaining, did an excellent job on the compilation of camera-ready papers and interaction with Springer. Many thanks go to Natalia Avdeenko and Ekaterina Livshits (Monomax Meetings & Incentives) for their excellent local organization efforts and the conference's social program preparation.

We believe that the 11$^{\text{th}}$ NEW2AN and 4$^{\text{th}}$ ruSMART conferences provided an interesting and up-to-date scientific program. We hope that participants enjoyed the technical and social conference program, the Russian hospitality and the beautiful city of St. Petersburg.

June 2011

Sergey Balandin
Yevgeni Koucheryavy
Honglin Hu

# Organization

## NEW2AN International Advisory Committee

| | |
|---|---|
| Nina Bhatti | Hewlett Packard, USA |
| Igor Faynberg | Alcatel Lucent, USA |
| Jarmo Harju | Tampere University of Technology, Finland |
| Andrey Koucheryavy | ZNIIS R&D, Russia |
| Villy B. Iversen | Technical University of Denmark, Denmark |
| Paul Kühn | University of Stuttgart, Germany |
| Kyu Ouk Lee | ETRI, Korea |
| Mohammad S. Obaidat | Monmouth University, USA |
| Michael Smirnov | Fraunhofer FOKUS, Germany |
| Manfred Sneps-Sneppe | Ventspils University College, Latvia |
| Ioannis Stavrakakis | University of Athens, Greece |
| Sergey Stepanov | Sistema Telecom, Russia |
| Phuoc Tran-Gia | University of Würzburg, Germany |
| Gennady Yanovsky | State University of Telecommunications, Russia |

## NEW2AN Technical Program Committee

| | |
|---|---|
| TPC Chair | |
| Roman Dunaytsev | Tampere University of Technology, Finland |
| Mari Carmen Aguayo-Torres | University of Malaga |
| Ozgur B. Akan | METU, Turkey |
| Khalid Al-Begain | University of Glamorgan, UK |
| Sergey Andreev | State University Aerospace Instrumentation, Russia |
| Tricha Anjali | Illinois Institute of Technology, USA |
| Konstantin Avrachenkov | INRIA, France |
| Francisco Barcelo | UPC, Spain |
| Sergey Balandin | FRUCT, Finland |
| Thomas M. Bohnert | SAP Research, Switzerland |
| Torsten Braun | University of Bern, Switzerland |
| Chrysostomos Chrysostomou | University of Cyprus, Cyprus |
| Nirbhay Chaubey | Institute of Science and Technology for Advanced Studies and Research (ISTAR), India |
| Ibrahim Develi | Erciyes University, Turkey |
| Eylem Ekici | Ohio State University, USA |
| Sergey Gorinsky | Washington University in St. Louis, USA |
| Markus Fidler | NTNU Trondheim, Norway |

| | |
|---|---|
| Giovanni Giambene | University of Siena, Italy |
| Stefano Giordano | University of Pisa, Italy |
| Ivan Ganchev | University of Limerick, Ireland |
| Victor Govindaswamy | Texas A&M University, Texarkana, USA |
| Vitaly Gutin | Popov Society, Russia |
| Andreas Kassler | Karlstad University, Sweden |
| Maria Kihl | Lund University, Sweden |
| Tatiana Kozlova Madsen | Aalborg University, Denmark |
| Yevgeni Koucheryavy | Tampere University of Technology, Finland (Conferene Chair) |
| Jong-Hyouk Lee | INRIA, France |
| Vitaly Li | Kangwon National University, Korea |
| Leszek T. Lilien | Western Michigan University, USA |
| Saverio Mascolo | Politecnico di Bari, Italy |
| Maja Matijaševic | University of Zagreb, FER, Croatia |
| Paulo Mendes | INESC Porto, Portugal |
| Pedro Merino | University of Malaga, Spain |
| Ilka Miloucheva | Salzburg Research, Austria |
| Dmitri Moltchanov | Tampere University of Technology, Finland |
| Edmundo Monteiro | University of Coimbra, Portugal |
| Seán Murphy | University College Dublin, Ireland |
| Marc Necker | University of Stuttgart, Germany |
| Nitin Nitin | Jaypee University of Information Technology, India |
| Mairtin O'Droma | University of Limerick, Ireland |
| Evgeni Osipov | Lulea University of Technology, Sweden |
| George Pavlou | University of Surrey, UK |
| Simon Pietro Romano | Università degli Studi di Napoli "Federico II", Italy |
| Alexander Sayenko | Nokia Siemens Networks, Finland |
| Dirk Staehle | University of Würzburg, Germany |
| Sergei Semenov | Nokia, Finland |
| Burkhard Stiller | University of Zürich and ETH Zürich, Switzerland |
| Weilian Su | Naval Postgraduate School, USA |
| Arvind Swaminathan | Qualcomm Inc., USA |
| Veselin Rakocevic | City University London, UK |
| Dmitry Tkachenko | IEEE St. Petersburg BT/CE/COM Chapter, Russia |
| Vassilis Tsaoussidis | Demokritos University of Thrace, Greece |
| Christian Tschudin | University of Basel, Switzerland |
| Andrey Turlikov | State University Aerospace Instrumentation, Russia |
| Kurt Tutschku | University of Vienna, Austria |
| Alexey Vinel | SPIIRAN, Russia |
| Lars Wolf | Technische Universität Braunschweig, Germany |

## NEW2AN Additional Reviewers

Cruz Luis
Cruz Tiago
Díaz Zayas Almudena
Dimitrova Desislava
Govindaswamy Visvasuresh Victor
Granjal Jorge
Hoene Christian
Jakubiak Jakub
Markovich Natalia
Martin-Escalona Israel
Mendes Paulo
Metzger Florian
Petrov Dmitry
Podnar Zarko Ivana
Poryazov Stoyan
Pyattaev Alexander
Rak Jacek
Recio Perez Alvaro Manuel
Riliskis Laurynas
Staub Thomas
Vassileva Natalia

## ruSMART Executive Technical Program Committee

| | |
|---|---|
| Sergey Boldyrev | Nokia Research Center, Helsinki, Finland |
| Nikolai Nefedov | Nokia Research Center, Zurich, Switzerland |
| Ian Oliver | Nokia Research Center, Helsinki, Finland |
| Alexander Smirnov | SPIIRAS, St. Petersburg, Russia |
| Vladimir Gorodetsky | SPIIRAS, St. Petersburg, Russia |
| Michael Lawo | Center for Computing Technologies (TZI), University of Bremen, Germany |
| Michael Smirnov | Fraunhofer FOKUS, Germany |
| Dieter Uckelmann | LogDynamics Lab, University of Bremen, Germany |
| Cornel Klein | Siemens Corporate Technology, Germany |
| Maxim Osipov | Siemens CT, Embedded Linux, Russia |

## ruSMART Technical Program Committee

| | |
|---|---|
| Sergey Balandin | Nokia Research Center, Finland |
| Michel Banâtre | IRISA, France |
| Mohamed Baqer | University of Bahrain, Bahrain |
| Sergei Bogomolov | LGERP R&D Lab, Russia |
| Gianpaolo Cugola | Politecnico di Milano, Italy |
| Alexey Dudkov | University of Turku, Finland |
| Kim Geunhyung | Dong Eui University, Korea |
| Didem Gozupek | Bogazici University, Turkey |
| Victor Govindaswamy | Texas A&M University, USA |
| Prem Jayaraman | Monash University, Australia |
| Jukka Honkola | Innorange Oy, Finland |
| Dimitri Konstantas | University of Geneva, Switzerland |
| Reto Krummenacher | STI Innsbruck, Austria |
| Alexey Kashevnik | SPIIRAS, Russia |

| | |
|---|---|
| Dmitry Korzun | Petrozavodsk State University, Russia |
| Kirill Krinkin | Academic University of Russian Academy of Science, Russia |
| Juha Laurila | Nokia Research Center, Switzerland |
| Pedro Merino | University of Malaga, Spain |
| Aaron J. Quigley | University College Dublin, Ireland |
| Luca Roffia | University of Bologna, Italy |
| Bilhanan Silverajan | Tampere University of Technology, Finland |
| Markus Taumberger | VTT, Finland |

## ruSMART Additional Reviewers

| | |
|---|---|
| Cugola Gianpaolo | Medeisis Arturas |
| Dominici Michele | Recio Perez Alvaro Manuel |
| Govindaswamy Visvasuresh Victor | Salmeron Alberto |
| Jakubiak Jakub | |

# Table of Contents

## Methods for Studying Smart Spaces



## Smart Spaces Solutions



# II II NEW2AN

## Wireless PHY and Power Control

## Ad Hoc Networks



## WSN



## Special Topics

## Simulation + Fundamental Analysis I



## Traffic Modeling and Measurement



## Simulation + Fundamental Analysis II

## Network Performance and QoS



## Cooperative



## P2P



## Overlay Networks and Content

## Applications and Services



## API and Software



## Video

# Part I

# ruSMART

# ECSTRA – Distributed Context Reasoning Framework for Pervasive Computing Systems

Andrey Boytsov and Arkady Zaslavsky

Department of Computer Science, Electrical and Space Engineering,
Luleå University of Technology, SE-971 87 Luleå, Sweden
{andrey.boytsov,arkady.zaslavsky}@ltu.se

**Abstract.** Pervasive computing solutions are now being integrated into everyday life. Pervasive computing systems are deployed in homes, offices, hospitals, universities. In this work we present ECSTRA – Enhanced Context Spaces Theory-based Reasoning Architecture. ECSTRA is a context awareness and situation awareness framework that aims to provide a comprehensive solution to reason about the context from the level of sensor data to the high level situation awareness. Also ECSTRA aims to fully take into account the massively multiagent distributed nature of pervasive computing systems. In this work we discuss the architectural features of ECSTRA, situation awareness approach and collaborative context reasoning. We also address the questions of multi-agent coordination and efficient sharing of reasoning information. ECSTRA enhancements related to those problems are discussed. Evaluation of proposed features is also discussed.

**Keywords:** Context awareness, situation awareness, context spaces theory, multi-agent systems, distributed reasoning, collaborative reasoning.

## 1 Introduction

Pervasive computing paradigm focuses on availability and graceful integration of computing technologies. Pervasive computing systems, like smart homes or micromarketing applications, are being introduced into everyday life. Context awareness is one of the core challenges in pervasive computing, and that problem has received considerable attention of the research community.

The majority of pervasive computing systems are massively multiagent systems – they involve potentially large number of sensors, actuators, processing devices and human-computer interaction.

In this paper we present ECSTRA - system architecture for multiagent collaborative context reasoning, and situation awareness. ECSTRA builds on context spaces approach [6] as context awareness and situation awareness backbone of the system.

The paper is structured as follows. Section 2 discusses the related work. Section 3 briefly addresses context spaces theory, an approach that constitutes the basis for low-level context reasoning and situation awareness in ECSTRA. Section 4 describes the structure of ECSTRA framework and addresses each of its architectural components

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 1–13, 2011.

in details. Section 5 describes the implemented mechanism for collaborative context reasoning and context information dissemination in ECSTRA. Section 6 provides and analyzes evaluation results. Section 7 discusses the directions of future work and concludes the paper.

## 2 Related Work

The research community proposed various approaches to address the problems of context awareness, situation awareness and distribution of context information.

Padovitz et. al. in [6] propose ECORA architecture. Being a predecessor to ECSTRA, ECORA utilizes situation awareness mechanism on the basis of context spaces approach. ECORA was a source of an inspiration for ECSTRA, but ECSTRA was designed and developed from scratch. Comparing to ECORA, our approach has extended support for multiagent reasoning, enhanced support for sharing and re-using reasoning information and also enables integration of context prediction and proactive adaptation components [1,2].

In the paper [4] authors introduced ACAI (Agent Context-Aware Infrastructure) system. ACAI approach introduced multiple types of cooperating agents: context management agent, coordinator agents, ontology agents, reasoner agents and knowledge base agents. Context is modeled using ontologies. Comparing to ACAI, our approach features less specialized agents that are less coupled with each other. Agents are acting and sharing information without establishing sophisticated hierarchy. It makes our approach more robust and flexible to common disturbing factors like agent migration or communication and equipment failures. Instead of using the ontologies like ACAI, ECSTRA uses the methods of context spaces theory, that provide integrated solution from low-level context quality evaluation to situation awareness.

The work [9] featured CDMS (Context Data Management System) framework. That approach introduced the concept of context space (the set of context parameters needed by context aware application) and physical space (the set of raw sensor data provided by environment). Dissemination of context data from physical spaces is arranged using the P2P network. The approach [9] provided very advanced solutions for context data retrieval: query evaluation, updates subscription, matching between context spaces elements and relevant physical spaces elements. Comparing to CDMS, our approach features much higher degree of independence between context aware agents. It ensures better capabilities of information exchange between peer reasoning agents, and it ensures the robustness to different agent entering or leaving the system. In our framework context reasoning is completely decentralized. Context aware agents are capable of exchanging the information between each other, not just from-bottom-to-top manner. We utilize relatively flat publish/subscribe system, which allows us to employ loose coupling between multiple context aware applications, and make the system even more robust to reasoning agent migration. Our approach to context reasoning is based on context spaces theory and situation awareness principle, which provides simple, but yet flexible and insightful way to reason about real life situations.

The work [8] proposed MADIP multiagent architecture to facilitate pervasive healthcare systems. As a backbone MADIP utilized secure JADE [10] agent framework in order to maintain scalability and security criteria. ECSTRA uses Elvin publish/subscribe protocol [11], which provides sufficient functionality for the search and dissemination of necessary context information, and also ensures independence and loose coupling of reasoning agents. Exact mechanisms for context awareness and situation awareness are not in the focus of [8], while it was the main concern when developing ECSTRA.

## 3 Theory of Context Spaces

The context reasoning and situation reasoning mechanisms of ECSTRA framework are based on context spaces theory. The context spaces theory is an approach that represents the context as a point in a multidimensional space, and uses geometrical metaphors to ensure clear and insightful situation awareness. The main principles of context spaces theory are presented in [5].

A domain of values of interest is referred to as a *context attribute*. For example, in smart home context attributes can be air temperature, illuminance level, air humidity, etc. Context attributes can as well be non numerical, like on/off switch position or open/closed window or door.

In spatial representation a context attribute is an axis in multidimensional space. Multiple relevant context attributes form a multidimensional space, which is referred to as *context space* or *application space*.

The set of values of all relevant context attributes is referred to as a *context state*. So, the context state corresponds to a point in the multidimensional application space. Sensor uncertainty usually makes the point imprecise to a certain degree. Methods to represent context state uncertainty include context state confidence levels and Dempster-Shafer approach [7].

The real life situations are represented using the concept of a *situation space.* Reasoning over the situation space converts context state into a numerical confidence level, which falls within [0;1] range. In context spaces approach a situation confidence value is viewed as a combination of contributions of multiple context attributes.

Confidence level can be determined using formula (1).

$$conf_S(X) = \sum_{i=1}^{N} w_i * contr_{S,i}(x_i) \quad (1)$$

In formula (1) confidence level for situation S at context state X is defined by $conf_S(X)$. The context state vector X consists of context attribute values $x_i$, the importance weight of i-th context attribute is $w_i$ (all the weights sum up to 1), the count of involved context attributes is N, and the contribution function of i-th context attribute into situation S is defined as $contr_{S,i}(x_i)$.

Contribution function is often a step function that is represented by formula (2).

$$contr(x) = \begin{cases} a_1, x \in (b_1, b_2] \\ a_2, x \in (b_2, b_3] \\ \dots \\ a_m, x \in (b_m, b_{m+1}] \end{cases} \quad (2)$$

Practically, any boundary can be included or excluded, for as long as the set of intervals covers entire set of possible values and the intervals do not overlap. For non-numeric context attributes the intervals are replaced with the sets of possible values.

Next section discusses our proposed ECSTRA framework.

## 4 ECSTRA Framework

ECSTRA (Enhanced Context Spaces Theory-based Reasoning Architecture) is a distributed mutiagent context awareness framework, with its architectural elements distributed across the pervasive computing system. ECSTRA is presented in figure 1.

Environmental parameters are measured and supplied by sensors. From the perspective of this research, human-computer interaction can be viewed as providing sensor input as well. However, raw sensor data is not the context information yet. Pervasive computing system needs to label that information, put necessary tags on it, evaluate its uncertainty and reliability, and distribute it within the system. These functions are carried out by the gateways. Gateways process information obtained through sensor networks, translate it into context attributes (most importantly, assign the unique names and define the uncertainties) and publish it to the special publish/subscribe service. Therefore, gateways process the sensor readings and create low-level context information out of it. Sensor uncertainty estimations can be either provided by sensors themselves or calculated by the gateway. Usually gateways are deployed on the devices directly connected to the sensor network base stations.

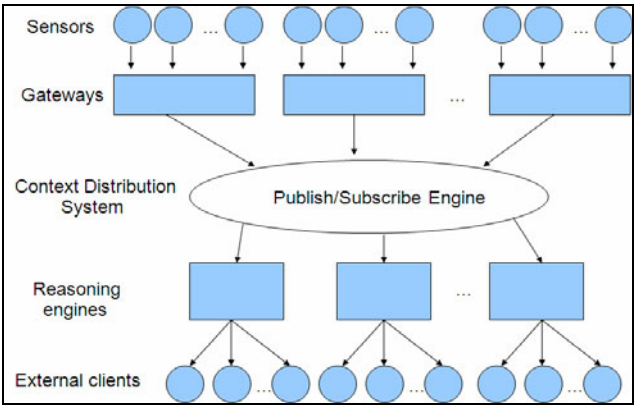



**Fig. 1.** Enhanced Context Spaces Theory-based Reasoning Architecture (ECSTRA)

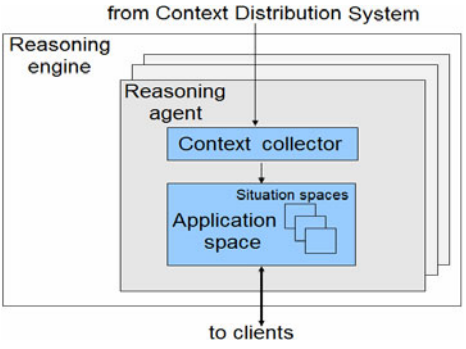



**Fig. 2.** Reasoning Engine Structure

ECSTRA uses publish/subscribe system as a tool to distribute the context information. We use Avis open source implementation [12] of Elvin publish/subscribe protocol [11]. Reasoning agents subscribe to necessary context attributes information, and gateways publish the data they have. With separate publish/subscribe approach the context dissemination process becomes very flexible and almost transparent. Both the gateways and the reasoning agents can migrate freely, and it takes just resubscription in order to restore the proper information flow.

Reasoning engines are container entities that consist of one or more reasoning agents. The structure of reasoning engine and reasoning agents is depicted in figure 2.

Reasoning agents directly perform the context processing and situation reasoning. Every reasoning agent works with some part of the context. Sharing reasoning process between several reasoning agents can help to make reasoning agents more lightweight and parallelize the reasoning process.

Reasoning agent comprises context collector and application space.

Context collector block has the following functions:

**1. Aggregating context data to a single context state.** Sensor readings arrive one at a time, but application space requires the entire context state vector at any moment.

**2. Delivering new context states to the application spaces.** Context collectors update the context state upon receiving of any new context attribute from publish/subscribe engine.

**3. Managing the subscription to context information.** It is the responsibility of context collector to subscribe to the necessary context information and to re-subscribe after the agent migration or restart.

**4. Track the deterioration of quality of context over time.** If there is no information received in a while, the context collectors update the context quality estimations and increase the expected uncertainty.

The application space block and the situation space blocks within it correspond to application space and situation spaces of context spaces theory. Application space handles context reasoning process and defines all the questions of situation algebra (reasoning about situation relations in terms of logical expressions, see [5]) syntax and semantics. Situation space handles all the questions of situation representation. Currently ECSTRA supports original context spaces theory situation definition, fuzzy

situation inference [3], and a set of specially developed flexibility-optimized situation space formats, which are the subjects of ongoing work.

Another function of application space is sending notification to the clients that new context data have arrived. Application space operates only with context states as input data, and between the changes of context state the reasoning results remain the same. The client does not have to be subscribed to context data change, in order to request reasoning from application space. If the client is interested in receiving the context state itself right after the update, it can subscribe directly to the context collector using the application space to context collector interface.

As it was noted before, the context state within the application space does not change in the time between the notifications of context collector. That allows reducing the reasoning efforts by introducing the cache of reasoning results. If any ECSTRA client requested the reasoning about the certain situation, either like a single situation or within the situation algebra expression, the reasoning results are put into reasoning results cache. Later, if the context state did not change yet, but the same situation is requested again (once again, either as a single situation or a situation within algebra expression), ECSTRA takes the information from the cache. After context state changes, situation confidence levels might change as well, and cached results can no longer be trusted. Therefore reasoning results cache is cleaned when the new context state is received.

External clients are not a part of ECSTRA (some exceptions are provided in section 5). They connect to the application space, send reasoning requests and obtain the result of reasoning. For example, CALCHAS context prediction and proactive adaptation framework [2] can be a client of ECSTRA. Usually reasoning requests are presented in the format of situation algebra expression, and reasoning results are either confidence level or Boolean value, that represents the validity of requested expression with respect to current context state.

As a part of this work, we implemented several enhancements to original ECSTRA architecture, which take the advantage of multi-agent and distributed nature of pervasive computing system.

## 5 Distributed Context Reasoning

The architecture of ECSTRA was designed to allow distributed reasoning and information sharing between agents. ECSTRA allows sharing high-level reasoning results. This approach encapsulates low-level reasoning efforts of different agents and reduces the amount of overlapped reasoning efforts between several reasoning agents.

Several particular features of ECSTRA enable distributed context reasoning. Those are context aware data retrieval, subscription-based reasoning result sharing and multilayer context preprocessing.

### 5.1 Context Aware Data Retrieval

Consider the following motivating scenario: the user is in a smart home. User's context is managed by PDA and the light level is requested from the sensor in the room. Practically it makes sense to treat the current light level around the user as a

user's context attribute. However, the context attribute "CurrentLightLevel" will be represented by different sensors depending on the context. Here the required sensor will depend on user location.

To summarize, sometimes single context attribute corresponds to different sensors in different occasions. The context aware data retrieval aims to overcome that problem. The idea is to adaptively resubscribe to different context sources, depending on the current context information itself.

The method is to enhance collector with two additional functions: resubscription on request, and masking the global name of context attribute to replace it with its local name. As a result, for the application space the technique is completely transparent, and the computational core does not require any modifications. New block, subscription manager is introduced to manage the subscription switching.

Context aware data retrieval architecture is depicted in figure 3.

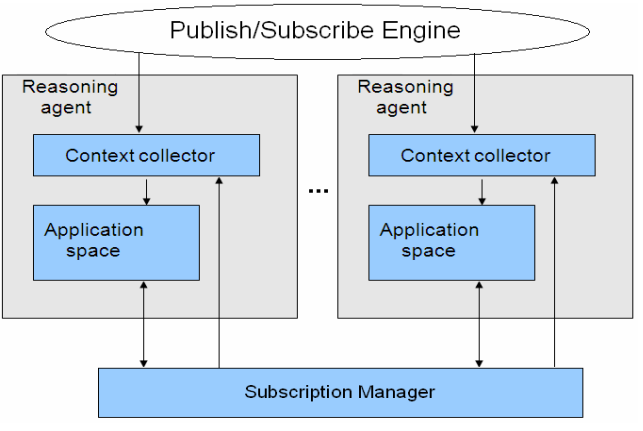



**Fig. 3.** Context Aware Data Retrieval – Architecture

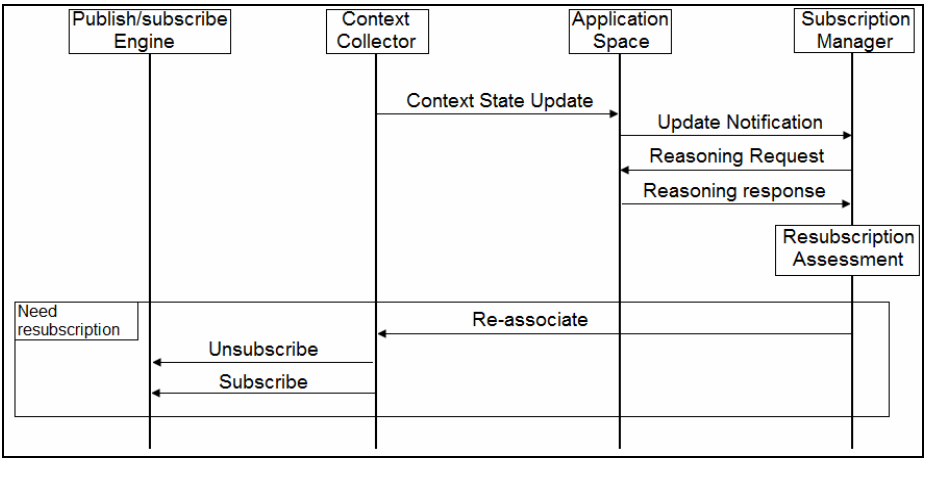



**Fig. 4.** Context Aware Data Retrieval – Protocol

Context collector accepts commands from subscription manager. Subscription manager, in turn, contains the set of rules that define resubscription procedures. Subscription manager acts as a client to the application space.

The simplified protocol of resubscription decision making is depicted in figure 4.

The efficient use of that technique can allow to significantly reduce the number of context attributes under consideration, and to bring the number of involved context attributes down to the tractable numbers even for large-scale systems. Also it can significantly simplify the development of situation spaces.

### 5.2 Reasoning Results Dissemination

In practice several remote clients can be interested in the results of reasoning agent work. Moreover, situation reasoning results can be taken as context attributes by other reasoning agents (if carried out properly, it can simplify the reasoning activities). The possible enhancement for that case is to reason about the situation and then to return the reasoning result into publish/subscribe engine. In ECSTRA it is implemented in a manner, presented in figure 5.

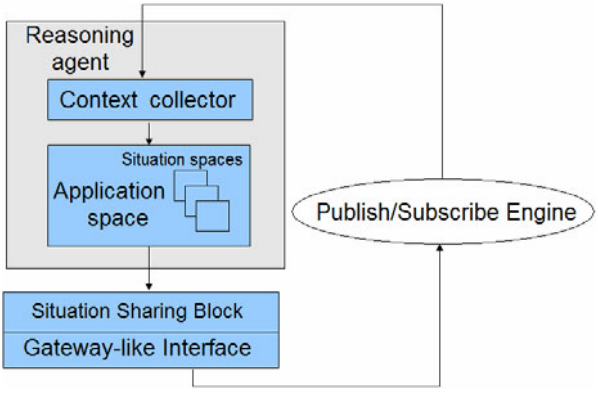



**Fig. 5.** Sharing of Reasoning Results

As depicted in figure 5, situation sharing block returns the results of situation reasoning to publish/subscribe system. The subscribers to those results can be either other reasoning agents, or remote external clients themselves. The outer interface of situation sharing block resembles the interface of a gateway. It allows packing the shared information in context attribute-like format. This format allows managing context attributes and shared situations in a unified manner. The situation reasoning results (in the format of confidence level or binary occurrence) can be subscribed to and taken as an input by other reasoning engine, and this can create hierarchical distributed reasoning structure.

This approach can significantly reduce the necessary amount of reasoning activities and allow efficient sharing of information between reasoning agents and the external clients. Also this approach can naturally construct a hierarchy of reasoning activities, and this hierarchy will be relatively flexible and robust to agent migration and replacement, especially if combined with context aware data retrieval.

### 5.3 Multilayer Context Preprocessing

If both the number of context attributes and the number of reasoning agents are large, the efforts for context preprocessing might be significant. Context preprocessing efforts can be reduced by applying the multilayer context preprocessing technique, depicted in figure 6.

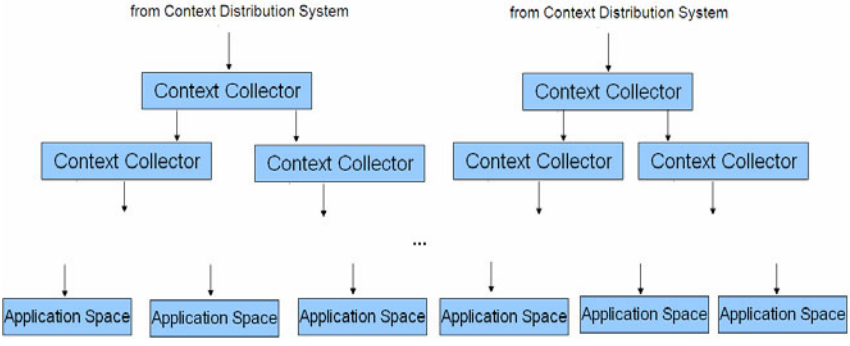



**Fig. 6.** Multilayer Context Preprocessing

If N reasoning agents within reasoning engine have the common subset V of context state vector, the system can construct the context collector for context state V. Then obtained context state V can be used by all N context collectors directly. As a result, instead of N times preparing the vector V, that vector will be derived just once, and then distributed among all the interested reasoning agents.

So, multilayer context preprocessing approach can reduce the context preparation efforts. It is completely tolerant to the migration of context sources. However, multilayer structure can cause problems during the reasoning agents migration. As a result, multilayer context preprocessing should be used when there are many reasoning agents, but those reasoning agents are not likely to migrate separately.

## 6 Evaluation of Situation Reasoning

The theoretical analysis of situation reasoning complexity is presented in table 1. The definition of the situation space is presented in section 3 in expressions (1) and (2). We refer to the total number of involved context attributes as N, to the number of involved intervals on i-th context attribute as $m_i$, and to the total number of involved intervals on all the context attributes as $M = \sum_{i=1}^{N} m_i$ .

It should be noted that $M \geq N$ – there is at least one interval per context attribute. In practice often M>>N. Summarizing table 1, we expect that reasoning time will be linearly dependent on the number of intervals.

**Table 1.** Situation Reasoning Complexity

| Operation | Order | Details |
|---|---|---|
| + | O(N) | On formula (1) the sum contains N summands. There are no more additions involved. |
| * | O(N) | On formula (1) every summand has one multiplication within. There are no more multiplications involved. |
| comparison | O(M) | Consider formula (2). For every context attribute the necessary interval will be taken the last in the worst case. That gives O(M) comparisons as the worst case estimation. |

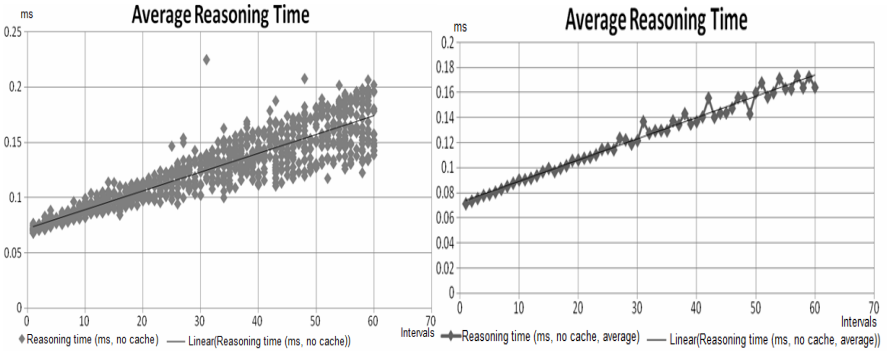



**Fig. 7.** Situation Reasoning Efficiency

The experiment was performed as follows. Situation spaces and context states were randomly generated. There were 1000 randomly generated situations in a single application space. For every generated situation the number of intervals was generated uniformly between 1 and 60. Then the distribution of intervals between context attributes was generated uniformly. For every situation the reasoning was performed 10000 times without using the results cache, and then average reasoning time was taken as a result. Testing was performed on Lenovo ThinkVantage T61 laptop.

The results are depicted in figure 7.

The exact analysis of hetersocedactic properties of situation awareness complexity is being done as a part of advanced situation awareness, which is mentioned as future work direction in section 8. For the purpose of current research averaging out the results for every number of intervals (right plot in figure 7) provides enough accuracy.

Situation cache provides significant improvement in reasoning time. We performed testing on similar equipment. The number of generated situations varied from test to test, and for every situation reasoning results were in the cache. The reasoning time was calculated as an average for 1000 random reasonings. The results are presented in figure 8.

More detailed results can be found in table 2.

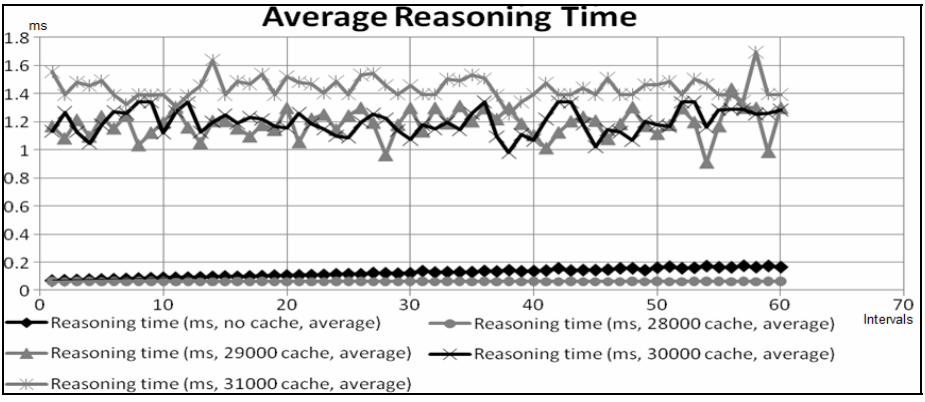



**Fig. 8.** Situation Cache Efficiency

**Table 2.** Situation Cache Efficiency

| **Cache Size (situations)** | 25000 | 26000 | 27000 | 28000 | 29000 | 30000 | 31000 |
|---|---|---|---|---|---|---|---|
| **Average Reasoning Time (ms)** | 0.0653 | 0.0628 | 0.0623 | 0.0623 | 1.1836 | 1.1972 | 1.4400 |

As expected, if the reasoning result is taken from the cache, reasoning time does not depend on the size of situation space, but it does depend on the number of situations in the cache. As it is shown in table 2 and figure 8, until the amount of situations in the cache reach 29000, reasoning time is ~0.06 ms, which is less than time for reasoning about 1-interval situation. However, when the number of situations in the cache reaches 29000, the reasoning time starts to grow rapidly.

So the general recommendation is to use the situation cache even for the situations with low number of intervals, unless the count of situations in the cache exceeds 28000.

## 7 Conclusion and Future Work

In this work we presented the pervasive ECSTRA computing framework and application, that is capable of context reasoning and situation reasoning. ECSTRA is designed to fit multi-agent and highly distributed nature of pervasive computing systems.

We identified the following possible directions of future work:

- **Advanced context aware data retrieval.** Currently the context aware resubscription technique is defined by static rules. It might work well for the relatively small systems, but for large-scale pervasive computing systems like smart towns it will result in enormous amount of rules. In order to address large-scale context aware data retrieval problem, we need advanced language of re-subscription rules, combined with efficient context attribute naming technique.

- **Advanced situation awareness.** The situation reasoning techniques of context spaces theory provide a memory efficient and fast situation awareness solution, but sometimes they lack flexibility, and many real-life situations cannot be defined in the terms of expressions (1) and (2). The search of new situation definitions and analysis of its efficiency is the subject of ongoing work.
- **Reliable distributed context delivery.** Reasoning agents are mostly vulnerable when they are migrating. If the context information update arrives during the migration of the agent (after unsubscribing, but before subscribing), it can as well be lost. In that case, loose coupling between sender and receiver, the important benefit of publish/subscribe system, becomes a disadvantage. The possible remedy for that problem is establishing some kind of knowledge storage agents that contain up-to-date data. Another possible option is introducing the request for data within publish/subscribe space.
- **Smart situation cache.** Currently situation cache can be either on or off. If situation cache is on, it stores any situation reasoning results. In section 6 we proved that situation cache significantly reduces reasoning time, unless there are tens of thousands of situations in it. Situation cache can be further enhanced by smart decision making about whether to put the situation in it or not. Situations in the cache can be preempted depending on number of intervals (and, therefore, expected saved time). Another possible enhancement is to allow entire situation algebra expressions in the cache. The implementation details of those techniques, as well as efficiency of those methods, are yet to be determined.

## References

1. Boytsov, A., Zaslavsky, A., Synnes, K.: Extending Context Spaces Theory by Predicting Run-Time Context. In: Balandin, S., Moltchanov, D., Koucheryavy, Y. (eds.) NEW2AN/ ruSMART 2009. LNCS, vol. 5764, pp. 8–21. Springer, Heidelberg (2009)
2. Boytsov, A., Zaslavsky, A.: Extending Context Spaces Theory by Proactive Adaptation. In: Balandin, S., Dunaytsev, R., Koucheryavy, Y. (eds.) NEW2AN/ruSMART 2010. LNCS, vol. 6294, pp. 1–12. Springer, Heidelberg (2010)
3. Delir Haghighi, P., Zaslavsky, A., Krishnaswamy, S., Gaber, M.M.: Reasoning About Context in Uncertain Pervasive Computing Environments. In: The Proceedings of the European Conference on Context and Sensing (EuroCSS 2008), Switzerland. Springer, Heidelberg (2008) (accepted for Publication)
4. Khedr, M., Karmouch, A.: ACAI: agent-based context-aware infrastructure for spontaneous applications. Journal of Network and Computer Applications 28, 19–44 (2005)
5. Padovitz, A., Loke, S.W., Zaslavsky, A.: Towards a theory of context spaces. In: Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops, pp. 38–42 (2004)
6. Padovitz, A., Loke, S.W., Zaslavsky, A.: The ECORA framework: A hybrid architecture for context-oriented pervasive computing. Pervasive and Mobile Computing 4(2), 182–215 (2008)
7. Shafer, G.: A Mathematical Theory of Evidence. Princeton University Press, Princeton (1976)
8. Su, C., Wu, C.: JADE implemented mobile multi-agent based, distributed information platform for pervasive health care monitoring. Applied Soft Computing 11, 315–325 (2011)

9. Xue, W., Pung, H., Ng, W., Gu, T.: Data Management for Context-Aware Computing. In: IEEE/IFIP International Conference on Embedded and Ubiquitous Computing, EUC 2008, pp. 492–498 (2008)
10. Bellifemine, F., Caire, G., Trucco, T., Rimassa, G.: JADE Programmer's Guide, http://jade.tilab.com/doc/programmersguide.pdf (accessed on April 6, 2011)
11. Elvin Protocol Specifications, http://www.elvin.org/specs/index.html (accessed on April 6, 2011)
12. Avis Event Router Documentation, http://avis.sourceforge.net/documentation.html (accessed on April 6, 2011)

# A Framework for Context-Aware Applications for Smart Spaces

M. Mohsin Saleemi, Natalia Díaz Rodríguez, Johan Lilius, and Iván Porres

Turku Centre for Computer Science (TUCS)
Department of Information Technologies, Åbo Akademi University
Turku, Finland
{msaleemi,ndiaz,jlilius,iporres}@abo.fi

**Abstract.** This paper presents an approach for developing context-aware intelligent applications for Smart Space-based infrastructure. The goal is to model and process context information using our development tool and Nokia's Smart-M3 architecture. We propose an adaptable and scalable context ontology, an ambient computing framework based on Smart Spaces and a rule based reasoning to infer high level context. Our approach deals with key issues in context aware ubiquitous computing such as adaptive and proactive changes in the environment, incorporation of novel sources of context information and automatic code generation from the context ontology to provide seamless interoperability.

## 1 Introduction

Context consists of any information that can be used to characterize the situation or state of an entity [7]. Entities can include anything e.g. a person, a physical object, an application or a device that is used to interact with the user. The concept of context has very broad view which can essentially consider anything as a context such as physical objects, applications, environment and the users. For example, physical context of a person can include his location, time etc; his social context can include his social relations e.g. family and friends etc, his activity context can include any tasks he performs in his daily life (watching TV, listening music, talking on phone, etc). In pervasive and context-aware computing, a user should be able to readily accomplish an action which possibly can include cooperation and collaboration with others using multiple devices and networks as he moves in the environment. In this way, a whole new universe of intelligent applications would automatically adapt to the user's intention. For example, your smart phone could notice that your favorite program starts in 5 min. based on your profile information or a fan page on Facebook and the TV guide available on the broadcaster's web page. Then it could use GPS to find that you are not at home and deduce that it needs to start the PVR (Personal Video Recorder) at home. This kind of intelligent applications need the context information from different sources to adapt to the user's preferences without involving human interactions. The context-aware intelligent applications can be realized by exposing the context information, internal data and functionality of

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 14–25, 2011.

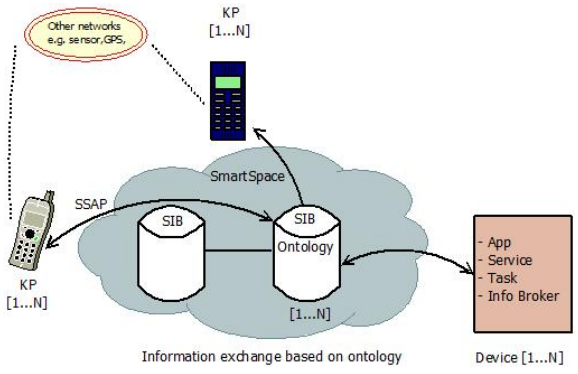



**Fig. 1.** Smart-M3 Architecture

the devices and ensuring data interoperability between them. This requirement is due to the variety of devices to be used and the need for interacting with each other within the context.

To enable the above mentioned cross-domain scenario and to solve the interoperability issue, one way is through the notion of *Smart Space*. A Smart Space is an abstraction of space that encapsulates both the information in a physical space and the access to this information allowing devices to join and leave the space. In this way the Smart Space becomes a dynamic environment whose identity changes over time when a set of entities interact with it to share information. For example, communication between the mobile phone and the PVR in the above scenario does not happen point-to-point but through the Smart Space whose members are the mobile phone and the PVR. We have developed a programming interoperability solution for rapid application development in Smart Spaces that can be extended to support context-aware intelligent applications [10]. Our solution is based on Nokia open source Smart-M3 architecture [4], an ideal choice for developing pervasive applications as it includes: 1) A blackboard software architecture which is cross-domain, cross-platform and enables knowledge share and reuse. 2) Ontology governance process (information stored in RDF) ensuring seamless information interoperability.

## 2 Smart-M3 Architecture

The Smart-M3 (Multi part, Multi device and Multi vendor) architecture [12] [4] provides a particular implementation of Smart Space where the central repository of information is the Semantic Information Broker (SIB). The Smart-M3 space is composed of one or more SIBs where information may be distributed over several SIBs for the later case. The devices see the same information, hence it does not matter to which particular SIB in a M3 space a device is connected. The information is accessed and processed by entities called Knowledge Processors (KPs). KPs interact with the M3 space by inserting, retrieving or querying

information in any of the participating SIBs using access methods defined by the Smart Space Access Protocol (SSAP). Smart-M3 provides information level interoperability to the objects and devices in the physical space by defining common information representation models such as the Resource Description Framework (RDF). Since Smart-M3 does not constrain to a specific structure of information, it enables the use of ontologies to express the information and relations in a KP application providing multi domain support. Figure 1 shows the overall Smart-M3 architecture.

## 3 Application Development for Smart-M3

We use an ontology-driven development approach for Smart-M3 for mapping ontologies to Object Oriented Programming [3] [10]. Our approach consists of two parts. The first part is the generator that creates a static API from an OWL ontology. This mapping generates native Python classes, methods and variable declarations which can then be used by the KP developer to access the data in the SIB as structured and specified in the OWL ontology. The second part is the middleware layer which abstracts the communication with the SIB. Its functionality is the handling of RDF Triples (*Subject, Predicate, Object*) with the generated API. This consists of inserting, removing and updating Triples and committing changes to the Smart Space. It also provides functionality for synchronous and asynchronous queries. Our approximation enables application developers to use the generated API to develop new KPs and applications without worrying about the SIB interface as the generated API takes care of the connection to the SIB each time an object is created.

In this application development approach, the concept of application is not the traditional control-oriented application running on a single device but a number of independently operated KPs which may run on different devices and are grouped together to be perceived as a single application. For instance, chat, calendar synchronization and multi-player games are examples of applications using this approach where a set of KPs, each handling a specific task, run on multiple smart devices and coordinate and interact with each other through the SIB to make a complete application. This coordination between KPs is done in the form of data exchange through the SIB where KPs subscribe to or query for specific data to perform a specified task. Application ontologies are used to describe data in the SIB and directs the KPs to access and manipulate data related to their functionality.

## 4 Context Ontology Model

Context information can be modeled using a variety of approaches such as key-value models, graphical models, object oriented models, logic based models and ontology based models [5]. We have chosen to use ontology based context modeling because of several reasons. Firstly, as our Smart-M3 architecture provides an interoperability solution based on ontology models, we can benefit from automatic code generation and ontology reasoning. Secondly, ontologies are the most

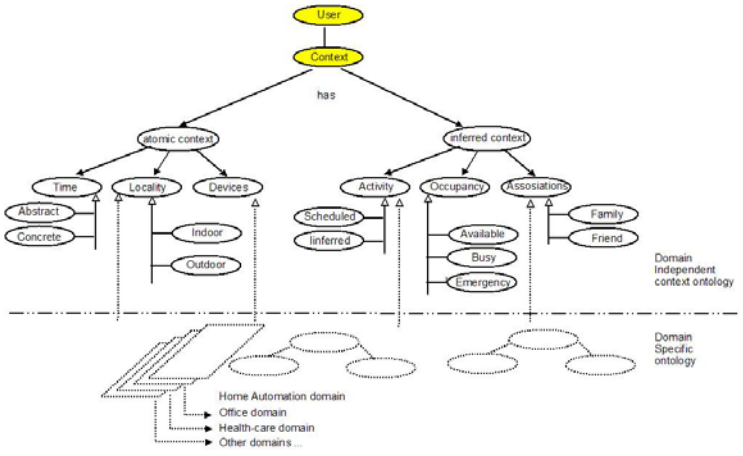



**Fig. 2.** Context Ontology of a user

promising and expressive models [5] fulfilling requirements for modeling context information. Thirdly, ontology based models provide advantages of flexibility, extendibility, genericity and expressiveness which are key factors in context-aware systems.

Information about the user's context is significant if enables the ambient system and applications to adapt to the user's preferences. In this paper we refer to all information that characterize the situation of a user as his *context*. In order to make the system more adaptive to the user's behavior, we propose to use multiple dimensions as the context of a user. Figure 2 shows these dimensions in an example context ontology. We divided the user's context in two broad categories, namely *atomic context* and *inferred context*. Atomic context refers to the context data acquired directly from context providers. The sources of atomic context can be any source providing relevant information to describe a user's situation. The inferred context refers to the information deduced from the given context data. We modeled user's context using six context dimensions: *Time*, *Locality*, *Devices*, *Activity*, *Occupancy* and *Associations*. Although this is not the only set, we believe that it is enough to capture most of the concepts. The user's context can be fully described in most of the domains by using these dimensions.

As the context ontology defines the basic terms and relations related to the user context that are applicable in every possible domain, we have defined it as an independent layer as shown in Figure 2. The user interactions involve a number of devices and appliances available in the environment which make the context dimensions of the ontology consider their activities and associations as domain independent types. The upper ontology in Figure 2 represents the core concepts to model user's situations in the environment and it appears feasible to have an unified upper context ontology capable of dealing large communities of users in wide range of domains. The lower part of Figure 2 shows the domain specific ontologies which describe concepts related to the domain in question by

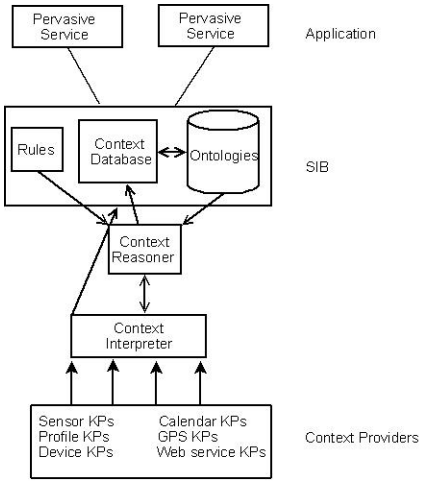



**Fig. 3.** System architecture

enumerating the concepts presented in the upper context ontology. For example, activity is a domain independent type of context but the tasks which are performed under this concept in the office domain such as *meeting, presenting* etc. are different when compared with tasks in the home-automation domain. Each specific domain has its own definition of the user's activity in that particular domain and even for the particular applications in that domain.

The system can thus, deduce information not explicitly given in the ontology, e.g. if the user is in the living room and the TV is ON then it implies that the activity is watching TV. Similarly, the system can deduce that the user is busy if he is talking on the phone, even if his calendar shows no activity at that time. In this way, by using the context information from different dimensions, the system can adapt to the user's current behavior and make the decisions rather dynamically.

## 5 System Architecture

Figure 3 shows the overall architecture of our context-aware system that supports pervasive applications from different domains. It consists of the following components.

*Context Providers*: A range of context provider KPs give atomic information about the user's context. Context providers cover from low level data obtained from sensors, GPS, RFID, WiFi, etc. to data from web services or user profiles. Atomic context information from these providers is used to infer new context information at higher level using inference rules and the context reasoner. The system provides two levels of inferences. At the first level, atomic context information infers new information while in the second the inferred context information is used to infer higher-level information.

*Context DataType Interpreter*: Context information from different data sources has diverse chronological features and data formats. The system needs a type conversion to map the value of one input type to another value of another input type to allow new and innovative information sources to be used. E.g. the temperature sensor's integer value at a given timestamp must be converted to the actual temperature in Celsius. GPS coordinates should be mapped to give the corresponding building number. The context interpreter is responsible for converting raw data to meaningful context information that can be put into ontology for use by other components of the system. In this way the context ontology is extended automatically. For this purpose all atomic data sources should specify their functionality in terms of input and output data types as well as other meta information such as how long is the validity of its data, how accurate is the data, its nature i.e data is sensed or defined etc. This can be done by using Web Ontology Language for Services (OWL-S) [1] which has capability to specify characteristics and functionalities of all the information sources. OWL-S would facilitate Context DataType Interpreter to easily map data values from heterogeneous devices.

*Context Reasoner/Rules Interpreter*: This module is responsible for inferring new higher level context from given atomic context information. The context reasoning is based on inference rules defined by KP developers which are then provided to the Python Rule Module. Also other information sources and inference rules can be provided as separate libraries. This module may trigger the execution of rules based on the current context information which in turn infers new contexts. It enables the context-aware system to be tailored for specific application scenarios. For example, if a user is in his bedroom, the bed sensorOn is true and the lightOn is false, the reasoner can infer that the user's activity is *sleeping* and put this inferred context information in the Semantic Information Broker. The context reasoner can also infer context properties using ontology reasoning by specifying inter-ontology relationships. This means inferring hierarchy classes and properties related to objects in the context (a sensor is attached to a sofa and the sofa is in the living room; thus, the sensor is in the living room).

*Ontologies*: OWL ontologies define the context. They represent both data directly sensed from context providers and information inferred from this data using inference rules. The context ontology describes generic concepts related to the individual and consists of atomic information from the information sources, inferred information from the atomic information using the inference rules and the inference rules. The core context ontology of an individual is extended when the new inferred information using the inference rules is added to it as shown in Figure 4.

*Inference Rules*: The inference process by the reasoner requires a set of inference rules that are used to infer new context related to an individual. These inference rules are created using a predefined format (section VI) and provided to the system as a set of imported libraries. The rules are domain specific and deterministic and the rule interpreter uses them to combine instances of context to infer new context information.

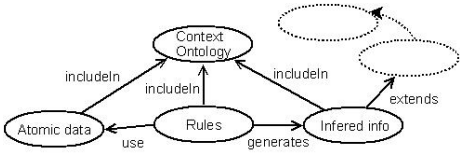



**Fig. 4.** Ontology extension based on Inferred context

### 5.1 Inference Rules and Context Reasoning

In context-aware ubiquitous systems where the emergence of increasing number of devices are used to perform desired services, we need to impose control constraining the participating devices' behavior. Rules can define how to react when a phone rings during a meeting; how to handle multiple requests to play different channels on a single TV at the same time; how to infer the user's activity using the active context information from multiple KPs. The inference rules, based on logic programming, allow context information origination from the provided set of ontologies. Its evolution/adaptation is caused by KPs taking part in the application. Figure 4 shows the ontological model after inferring context information. Dotted ellipses represent the application specific part of the context ontology.

We have defined an inference rule using a 3-clauses pattern:

*With clause* models declarations and assertions.

*When clause* contains events that trigger the rule.

*Then clause* includes conclusions representing inferred information after the rule is triggered.

Following there are few examples which illustrate our approach to define inference rules. The first rule states that if the user is in room B4050 at a specific time 13:20, and the room is occupied between 13:00 to 15:00 having more than one person there, then the user's activity is inferred as *busy in a meeting.*

```
with U:- User(id="1", role="Student", name="Mohsin"),
     R:- Room(room No.="B4050", location= "ICT"),
     P:- projector (id="101", type="ProModel")
when U.locatedIn (R, atTime"13:20"),
     R.occupied("13:00-15:00"),     P.locatedIn:- R,
     R.number of people > 1, P.statusON
then U.busyInMeeting
```

The inferred context information can then be used by another rule to infer other level information or to perform a task when some event is triggered. For example, the following rule states that when the user is in the meeting then forward incoming calls to his voice mail without interrupting him.

```
with U:- User(id="1", role="Student", name="Mohsin"),
     Ph:- phone(id="10", type="Iphone", model="4G")
when Ph.incomingCall, U.busyInMeeting, Ph.owner:- U
then Ph.activeVoicemail
```

There might be some emergency cases when the user does not want to ignore incoming calls. The following rule overwrites the result of the previous one when the calling person is the user's wife giving a beep to the user's phone. The user's relationship with the caller can be obtained from existing ontologies given to the system, such as Friend of a Friend (Foaf) ontology in this case.

```
with U:- User(id="1", role="Student", name="Mohsin"),
     Ph:- phone(id="10", type="Iphone", model="4G"),
     C:- Caller(name="Samra", association="wife")
when Ph.incomingCall,  U.busyInMeeting,
     Ph.owner:- U,  U.relation:- C
then Ph.beepOnce
```

The following rule infers based on the user's context information that the user is out for lunch.

```
with   U:- User(id="1", role="Student", name="Mohsin"),
     Res:- Restaurant(name="Unica", location= "ICT")
when ResOpen:- true, U.locatedIn(Res, atTime"12:15"),
     timeInterval > 5 min
then   U.havingLunch
```

A set of rules are to be specified for each domain. The rules above represent office domain. Based on the context information from multiple context dimensions, the system triggers, when certain changes happen, the activation of the associated rules.

## 6 Implementation

We use Python's meta-programming features to enable writing Python code which includes logic programming statements representing inference rules. These are inspired from the event-condition-action (ECA) rules model which is a technology from active databases for supporting dynamic functionality [9]. The interpreter for the inference rules is in its early stage of development.

### 6.1 Development Framework

The first task is defining a script language to use OWL2 allowing the user to express rules as section V showed. The second task is the integration of those logic expressions to work with ontologies into a functional Object-Oriented language. Because of its versatility, metaprogramming opportunities and ease of prototyping (easy to learn and use) we chose Python. Thus, given a context, the programmer could define in a simple way and beforehand the underlying rules that pervasively help the user daily in his Smart Space.

The third task is the integration of first and second approaches with the *Smart-M3 Ontology to Python API Generator* framework [3], which makes more intuitive to the programmer the definition of pervasive applications. This tool

provides automatic generation of a Python API for every OWL class as well as setters and getters among other methods to interact effortlessly with the common SIB through which all KPs communicate with each other.

In this section we focus on the second and third tasks and its motivation. Given the functionality provided by the Smart-M3 Ontology-Python framework, there is a need for designing a rule syntax language that allows users -with basic programming skills- easy definition of rules to model DIEM applications. In this way the need for learning OWL or query languages is minimized or null.

### 6.2 Programming Knowledge Processors in Python

The main feature of the Python Rules Module is to encapsulate, acting like a bind, the SIB interface. Our implementation approach is inspired by *Pythologic, Prolog syntax in Python* [2].

A Rule is structured as follows:

```
With() |= When() >> Then()
```

- **With()** Clause represents *assumptions* about existence of individuals.
- **When()** Clause represents *conditions*, when the KP must execute.
- **Then()** Clause represents *actions* to trigger.

In this way, the application programmer does not deal with RDF Triples directly but mainly with logic Python expressions. Therefore, the programmer could embed into Python code expressions like:

```
1  condition1 = lambda: user.isBusy()
2  condition2 = lambda: room.getOccupied()
3  conditions = [condition1, condition2]
4  action =  lambda: user.setVoiceMail(True)
5  myRule = With([user, room]) // When(conditions) >> Then(
       action)
6  diem.addRule(myRule)
```

**Listing 1.1.** Rule definition with Python Rules Module

The underlying implementation of the Python Rules Module translates Python logic expressions to the SIB API main interface: *Query, Subscribe, Insert, Remove, Update*. Thus, the Python Rules Module just needs to be imported to be used with the KP class where the DIEM application is coded:

- **With()**: If instances in *With()* exist in the SIB (SIB-Query), proceeds to evaluate *When()*. The check includes the ontology's Python object declaration, this is, other KPs know about it.
- **When()**: If *When()* is true (SIB-Query), executes *Then()*. If not, sets a SIB-Subscription to the attributes in *When* clause. The subscription capability provided by the Smart-M3 SIB allows knowing when the value of certain attribute has changed so that the rule can be evaluated again avoiding, in this way, unnecessary infinite query loops or traffic bottlenecks.

– **Then()**: If *With()* & *When()* satisfy, executes *Then()*, which translates into SIB-Update/ SIB-Add/ SIB-Remove/ SIB-Unsubscribe (results may update RDF Triples).

A Knowledge Processor can be located e.g. in any smart phone or device and can be for example a phone application for getting the local temperature from the Internet or a sauna/thermostat activator. All the KPs can be created and connected to the Smart Space (called 'x' in this example) in the following way:

```
7     def main(args):
8     app = QtGui.QApplication(sys.argv)
9     smartSpace = ('x', (TCPConnector, ('127.0.0.1', 10010))
          )
10    KP = PhoneKP.create(smartSpace)
11    # Definition of Rules
12    sys.exit(app.exec_())
```

**Listing 1.2.** KP Programming and Connection to the Smart Space 'x'

Straight after the KP is created, the user could define the Python rules related to the existing KPs. Then, connect the KPs to the Smart Space and run them is the only thing left.

If `EmptyKP.py` (provided by the Ontology-Python Generator) is used, instance declarations will automatically translate to insertions of Triples into the SIB. This allows other KP applications connected to the same Smart Space to know about those individuals' existence to interact with them. In the Python Rules Module, every KP application contains a `TripleStore` instance (produced by Ontology-Python Generator) representing the Smart Space' SIB. At last, the *With()*, *When()* and *Then()* clauses translate its Python statements to one of the implementation options given by the Ontology-Python Generator. These are SIB calls in *RDF* or *WQL* language. Our approach shows that learning OWL or query languages is not needed for interconnections with the SIB and interactions with other devices' KPs.

## 7 Related Work

The research in context-aware computing provides a wide number of context-aware systems and approaches for application development. Starting with context modeling, there are plenty of different points of view, but since the ontology model wins the rest regarding simplicity, flexibility, extensibility and expressiveness, we focus in comparison with ontology based systems and similar systems to ours. A good compendium of pros and cons in relation to design architectures and context models [5] shows that, already in 2004, similar advanced ontology-based context models were presented.

CoBrA and SOCAM are some of them using their own OWL-based approach for context processing while others like Context Managing Toolkit describe context in RDF. CoBrA [6] is proposed as an agent-based infrastructure for context modeling, context reasoning and knowledge sharing using context acquisition

components. The Soupa and CoBrA-Ont Ontologies are some of the related tools. They also provide techniques for user's privacy control. SOCAM [8] introduces another architecture for building context-aware services focused on information sensing and context providers. A central server or context interpreter gains context data through distributed context providers and processes it before sending it to the clients. These and other projects as [14] focus basically on creating ontologies for context-representation but they do not intend to build a framework for creating location- and context-aware development of services or applications based on the semantic back end.

In [13], the authors extend typical operating system concepts to include context-awareness. Gaia presents a similar representation to our RDF Triples with 4-ary predicates (the 4th one is context-type) and it does not use OWL but DAML + OIL. Gaia's MVC model also differs from our blackboard architecture. In [11] and [15], the authors presented a framework that targets only smart phone platform and technologies.

Context Toolkit [7] presents an approach to enable application development by using reusable components. However, its attribute-value tuples are not clearly meaningful enough making the application programming restricted. All these systems use SQL to access the central database. Contrasting to our RDF/WQL queries, we makes the queries to be restricted to a smaller set of statements.

With the purpose of facilitating the creation of services we can see that diverse technology-specific frameworks have been created but none of them outcomes with a clear functional programming tool. In comparison to the previous systems, our approach for modeling and processing context addresses the challenge of context-aware ubiquitous computing in smart environments using automated ontology code generation for Python. And our idea is expressive enough to represent domain specific context as abstract context data.

## 8 Conclusions and Future Work

In this paper, we expressed our ideas for context-aware applications for Smart Spaces. We presented our contextual ontology for modeling context information and the overall system architecture. The structure and syntax of inference rules are described with office domain scenario. We conclude that Smart Spaces are well suited for ambient applications to adapt to the user's preferences because they can provide information about the physical environment which can be shared and reused by many dynamic applications. In the future, we aim to implement a context manipulation library for our Smart-M3 tool to process contextual information. Next challenges to be tackled are e.g. consistency related issues as the insertion of individuals' properties into the SIB for the whole Smart Space could create ambiguity if not checked. Moreover, extra functionality to efficiently implement subscriptions to individuals' attributes is needed. After the basic functionality is consistent, new environments's use cases could be applied.

**Acknowledgment.** The research work presented in this paper is funded through the ICT-SHOCK DIEM project.

## References


1. OWL-S Services, http://www.daml.org/services/owl-s/
2. Pythologic, Prolog Syntax in Python, http://code.activestate.com/recipes/303057-pythologic-prolog-syntax-in-python/
3. Smart-M3 Ontology to Python API Generator, http://sourceforge.net/projects/smart-m3/files/smart-m3-ontology_to_python-api_generator_v0.9.1beta.tar.gz/
4. Smart-M3 software at sourceforge.net, release 0.9.4beta (May 2010), http://sourceforge.net/projects/smart-m3/
5. Baldauf, M., Dustdar, S., Rosenberg, F.: A Survey on Context-Aware systems. International Journal of Ad Hoc and Ubiquitous Computing 2 (2007)
6. Chen, H., Finin, T., Joshi, A.: An Ontology for context-aware pervasive computing environments. In: Proceedings of the Workshop on Ontologies in Agent Systems (2003)
7. Dey, A.K., Abowd, G.D.: The Context Toolkit: Aiding the development of context-aware applications. In: Workshop on Software Eng. for Wearable and Pervasive Computing (2000)
8. Gu, T., Pung, H.K., Zhang, D.Q.: A middleware for building context-aware mobile services. In: Proceedings of IEEE Vehicular Technology Conference (2004)
9. Bailey, P.T.W.J., Poulovassilis, A.: An event-condition-action language for XML. In: Proceedings of the 11th International Conference on World Wide Web (2002)
10. Kaustell, A., Saleemi, M.M., Rosqvist, T., Jokiniemi, J., Lilius, J., Porres, I.: Framework for Smart Space Application Development. In: Proceedings of the International Workshop on Semantic Interoperability, IWSI (2011)
11. Matetelki, P., Pataki, B., Kovacs, L.: Service-oriented context-aware framework. In: Young Researchers Workshop on Service-Oriented Computing (2009)
12. Oliver, I., Honkola, J.: Personal semantic web through a space based computing environment. In: Proceedings of the 2nd Interntional Conference on Semantic Computing (2008)
13. Roman, M., Hess, C., Cerqueira, R., Ranganathan, A.: A middleware infrastructure for active spaces. IEEE Pervasive Computing (2002)
14. Wang, X.H., Zhang, D.Q., Gu, T., Pung, H.K.: Ontology based context modeling and reasoning using OWL. In: Workshop Proceedings of the 2nd IEEE Conference on Pervasive Computing and Communications (2004)
15. Jin, Y., Her, Y., Kim, S.-K.: A context-aware framework using ontology for smart phone platform. International Journal of Digital Content Technology and its Applications 4(5) (2010)

# Analysis of the Energy Conservation Aspects of a Mobile Context Broker

Saad Liaquat Kiani[1], Boris Moltchanov[2], Michael Knappmeyer[1,3], and Nigel Baker[1]

[1] University of the West of England, Bristol, UK
[2] Telecom Italia Lab, Torino, Italy
[3] University of Applied Sciences Osnabrück, Osnabrück, Germany

**Abstract.** Mobile devices are increasingly becoming the main mean of interaction for inhabitants of ubiquitous computing environments. Pervasiveness of these devices as users' personal gadgets and their high-tech capabilities allow capturing of broad contextual information about the physical environment, users social profile and preferences. These modern roles of mobile devices facilitate a number of user and environment related context consuming and producing applications to be hosted on these devices. But without a coordinating component on the mobile device these context consumers and providers are a potential burden on device resources, specifically the effect of uncoordinated computation and communication shortens the battery life. In this paper we briefly describe the concept of a Mobile Context Broker and focus on the energy conservation benefits gained through the context coordination facilities provided by the Mobile Context Broker executing on a smart mobile device. The reported results signify reduction in energy consumption.

**Keywords:** Mobile Context Broker, Energy Conservation, Performance Evaluation.

## 1 Introduction

Ubiquitous computing is characterised by pervasively connected devices that unobtrusively become part of our daily experiences. These computing devices are not just personal computers but include wearable sensors, mobile devices and environmental sensors that project a digital snapshot of the environment, its inhabitants and their activities. Context-based applications and services based on these interconnected digital artefacts have the potential to enhance our experiences and interactions with the digital world. The complexity and dynamics exposed by the real world increase tremendously once the processing moves out from the well controlled and understood limits of the desktop environment. The challenge, and ultimate success, in this domain lies in application and service development and provisioning of contextual information that is readily usable in situations arising dynamically in day-to-day activities.

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 26–37, 2011.

The role of mobile devices has undergone marked transformation in the last decade, from the initial role of a mobile phone to the modern role of an information acquisition, processing and communication device. Moreover, our social life is increasingly effected through these devices as they follow us everywhere and provide us with immediate accessibility to our personal communication tools such as email, messaging and social networks. These additional roles are enabled by integrating various sensors into the device including GPS receivers, microphones, cameras, magnetometers, RFID readers, proximity and motion detectors in the form of gyroscopes and accelerometers. Combined with the increasing mobility of modern users, high-speed connectivity, a myriad of information and content services, availability of various communication technologies (Bluetooth, GPRS, WiFi) and presence of numerous digital artefacts embedded in the environment, mobile devices are increasingly becoming the primary tool of digital interaction for inhabitants of the digital world.

In order to enhance our digital experiences and provide useful context-based services, applications on our mobile devices take advantage of availability of information and the ability to interact with surrounding devices and embedded digital artefacts. These *context-consuming* applications utilise the knowledge created by *context providers* (services or applications) hosted either on the device itself or accessible over one of the communication interfaces. Examples of context acquisition carried out by *context providers* include interfacing with wearable sensors, location and proximity detection, gathering parameters for activity recognition etc. Context consumers, such as navigation software and other location-based applications, can use the information gathered by the providers to deliver useful services to the user. Even with the continuing increase in processing power and computing resources of mobile devices there is still reliance on network-based services and devices are mostly restricted to information acquisition and consumption. In addition to the sharing of information between local (device based) providers and consumers, significant contextual information exchange takes place between the device based consumers/providers and those deployed in the network infrastructure. These include network services that depend on contextual information collected at the user device (e.g. location, proximity) and device based information consumers that depend on context providers hosted in the network (e.g. social networks, weather service, online calendars, environmental sensors). Our earlier work [10] postulates how these enhanced capabilities and increased adoption of smart mobile devices and emerging context-awareness technologies will act as a driving force for introduction of intelligently personalised services in smart spaces and novel business models for enterprises.

In ubiquitous computing environments context consuming and providing applications/services on mobile devices are part of a complex distributed software system working towards a common goal of enhancing user experience unobtrusively. However, to be useful, context acquisition and processing is carried out unobtrusively and continually. This background processing is a constant drain on the battery life of the device. The scale, mobility and heterogeneity of devices and entities involved in context-awareness related functions pose further

challenges in coordination and communication of context across the distributed system. Without the presence of a coordination mechanism in the mobile device, each context consumer and provider has to manage communication and coordination with external services in the network individually, resulting in repetition of functionality that incurs a cost in terms of development time and resource usage. In this paper, we describe the concept and prototype of a Mobile Context Broker (Section 3) that provides this essential coordination facility to context consumers and providers on the device and those hosted on the network. Our focus is the analysis of its effect on energy consumption on the mobile device which is presented in Section 4. Before presenting the core discussion of this paper, it is essential to discuss the larger system, entitled the Context Provisioning Architecture, in which the Mobile Context Broker operates; this background is provided in Section 2.

## 2 Background

Brokering is a software architectural pattern [3] for distributed systems. The broker pattern is used to structure distributed software systems with decoupled, independent components that share information by message passing or service invocation. The use of brokers in information distribution systems is well established, e.g. CORBA [2], Web Services Brokered Notification [1]. In the domain of context-aware computing broker-based architectures have been developed for collecting, processing and distributing user and environmental context, e.g. Context Broker Architecture [4], MobiLife [11] and Context Casting [8]. These systems are based on a centralised broker architecture in which a single, central broker facilitates the flow of contextual information between context consuming and producing components of the system. We have improved upon this centralised architecture by extending the work carried out in the Context Casting project and designed a multi-broker context provisioning system. In our earlier work [6] we have presented the model for large-scale context provisioning system based on the federation of brokers for coordination and dissemination of contextual information. Applications that use context are labelled as Context Consumers (`CxC`), whereas Context Providers (`CxP`) are context producing applications/services and Context Brokers (`CxB`) act as facilitators that, at the most basic level, route context requests and responses between consumers and providers.

In the *Context Provisioning Architecture*, context providers and consumers register and communicate with only one of the brokers. Network brokers exchange routing tables to form an overlay network of brokers. Publish/subscribe communication semantics are used to issue context subscriptions and notifications between the clients (context providers and consumers). Simple routing with advertisements is used between the federated brokers for forwarding subscriptions and notifications across the broker network. The system uses an XML based scheme, entitled ContextML [9], to represent context information, subscriptions, notifications, client advertisements and other context management messages. Context information in the system is tagged under unique *context*

*scopes*, e.g. weather, position, deviceStatus, etc. A more detailed description of these components is provided in [8].

An area of concern in our architecture is the involvement of mobile devices. When a number of CxCs and CxPs are executing on a mobile device, each application is responsible for maintaining its registration with a CxB that is deployed on a network server, sending subscriptions and receiving notifications. This results in repetition of similar tasks amongst the set of context related applications executing in our system. Moreover, during periods of network disconnections the performance of a device based CxC/CxP depends on the behaviour implemented by the application developer, i.e. it may continue trying to access the network or it may move to a paused state until the network becomes available again. An inappropriate implementation of a CxC/CxP on the device may result in wastage of computing and communication resources on the device.

We have developed the Mobile Context Broker component to facilitate the execution of device based CxCs and CxPs. The conceptual basis of the utility of deploying a broker on mobile devices for coordinating contextual information has also been discussed in our earlier work [6]. One of the reasons why a coordinating entity on mobile devices, specifically for coordinating information between device-based and network-based CxCs and CxPs, has not attracted much attention in this domain is that the ability of mobile devices to host such software components has emerged only recently. Other factors include lack of context sources (e.g. sensors and services), data network connectivity available to mobile devices and usability scenarios for such applications and services.

## 3 Mobile Context Broker

The Mobile Context Broker (mCxB) is a software component designed to execute on a mobile device as a background service that brokers information exchange between CxCs and CxPs, hosted either on the device or in the network. The CxPs and CxCs register their presence and requirements during execution to this broker and do not have to lookup each other individually. Moreover, during periods of disconnected operation, which is still common in mobile devices and networks, the CxCs and CxPs do not have to monitor device connectivity individually; this task is delegated to the mCxB. Polling and waiting for events or information to become available by CxC components is improved by using the publish/subscribe communication paradigm and using the broker as an event service that manages notifications and subscriptions. These functions provided by the broker save valuable computation cycles and consequently reduce energy consumption. The main functional characteristics of the Mobile Context Broker that facilitate these advantages of the mobile broker are presented in the following sub-section.

### 3.1 Design and Functional Architecture

The design of the mobile broker is based on the set of functions it provides to the CxCs and CxPs. These functions are listed below:

**Registration and Lookup.** Each CxC and CxP register with the mCxB specifying their communication end point and the type of information they provide or require. This function in turn enables the brokering function in which the mobile broker can direct a CxC requesting a particular type of context (scope) to the correct CxP(s).

**Subscription Management.** A CxC subscribes with the mCxB specifying the type and instance of context it requires and the duration for which the subscription remains valid. The mCxB can forward the subscription to the appropriate CxP or filter information produced by a CxP in order to satisfy the subscription.

**Notification and Dissemination.** The mCxB, on availability of subscription satisfying information, notifies the CxC of the availability or the context is directly communicated to the CxC.

**Caching.** The mCxB can cache recently produced context in order to exploit the principle of *locality of reference*, as done routinely in internet communications, to improve overall performance.

**Querying.** The mCxB provides a query resolution service via which CxCs can request the mCxB to fetch information from the CxPs. The querying function is equivalent to a one-off subscription, valid only once – irrespective of whether it results in meaningful response from the provider or unavailability of information.

The mCxB offers these functions to CxCs and CxPs by exposing well defined interfaces, as shown in Figure 1. Externally, network based clients can communicate directly with the mCxB but the mobility causes changes in communication end points and hinders reachability of the mobile broker (and its clients). This issue can be addressed by updating all external clients whenever the communication end point of the mobile broker changes due to mobility. A better approach is to communicate with the external CxPs and CxCs via a network-based registration service or broker. Our prototype implementation uses the latter approach by federating the mobile and network brokers together into an overlay network of brokers. Clients of a broker only communicate with the local broker and queries, subscriptions and notifications are routed between brokers using a

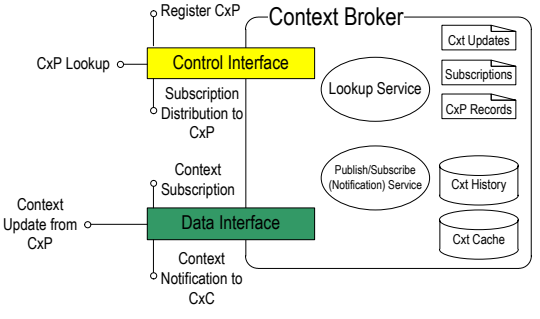



**Fig. 1.** Functional architecture of the mobile broker component

publish-subscribe communication paradigm. A detailed description of this model is provided in our earlier work [6].

### 3.2 Prototype Mobile Context Broker

Our prototype mobile context broker has been developed for the Android platform and runs as a background service. It provides the interfaces to `CxC` and `CxP` applications executing on smart mobile devices. Various mobile device based context-consuming applications that have been developed include context-based content providers (news, entertainment, etc.), location based gameplay using Layar augmented reality browser and context-based shopping recommendation applications. Mobile device based `CxP`s include *Location*, *Proximity* and *Activity CxPs*. Various `CxP`s are deployed on the network and include *Weather CxP*, *User Profile and Preferences CxP*, etc. In addition the mobile device acts as a gateway for nearby sensors (and sensor networks for dissemination of their sensed information). Development is underway to use Bluetooth enabled, wearable air quality sensors for participatory sensing and using the mobile broker to disseminate collected air quality information.

On the network side, the context broker is based on the JavaEE technology and exposes RESTful (Representational State Transfer [5]) interfaces to `CxC`s and `CxP`s in the network. Normally, communication between the brokers and their clients takes place over HTTP using ContextML [9] as the data model. The choice of these standardised technologies provides interoperability and allows for interaction between brokers and clients executing on heterogeneous hardware. The prototype mobile broker, `CxP`s and `CxC`s that have been developed for the Android platform are also able to utilise the Inter-Process Communication (IPC) facility of the Android platform in addition to HTTP based communication.

A significant feature of the broker is the caching facility. As discussed in [9], any instance of context information only remains valid for a certain amount of time. We utilise this temporal property of context information by annotating a validity period in context meta-data. The broker uses this validity period to cache recently retrieved information.

## 4 Mobile Context Broker and Energy Consumption

We have used the concept of the `mCxB` presented in the preceding section for coordination and dissemination of contextual information. Context information acquired at the mobile device by `CxP`s or requested by the mobile device based `CxC`s from local or external `CxP`s is brokered via the `mCxB`. In the following paragraphs, we describe the scenario used to evaluate the impact of the prototype `mCxB` on energy consumption.

### 4.1 Experiment Scenario

In order to experimentally analyse the benefits (or shortcomings) of using a `mCxB`, a scenario is designed to monitor various parameters with and without the use

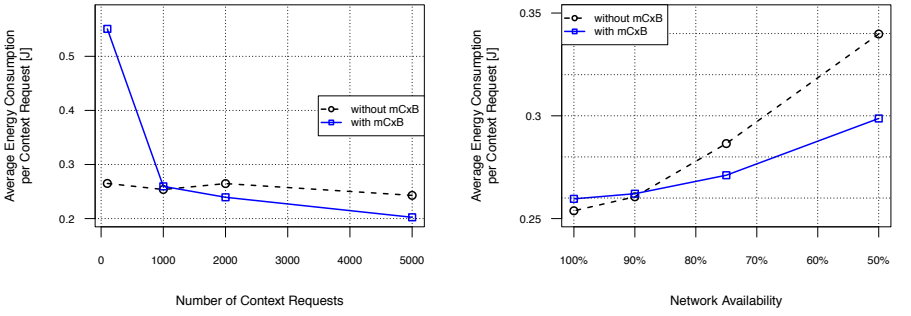
of the mobile broker. Five CxCs and five CxPs are deployed on the mobile device. Similarly, five CxCs and five CxPs are deployed on the network. For simplicity each CxP provides a unique type of context, i.e. context scope, and each CxC is only interested in one context scope available at one of the CxPs.

The experiment is carried out with and without the mCxB in separate runs. In absence of the mCxB, device based CxCs and CxPs register with and use the network CxB. Each CxC sends subscriptions and queries to at most one CxP. Each experiment is repeated on the same set of hardware (mobile device and network servers) and the WiFi interface is used on the server and mobile device within a single WLAN for minimising the effects of unpredictable network round-trip times. The mobile device is an Android 2.3 based Nexus One phone. PowerTutor [12] is used for calculating the energy used by individual applications (mCxB, CxCs and CxPs) on the device. Only the energy used by an application in utilising the CPU and WiFi is considered when calculating the energy consumption of an application. The reported results are averages of three repetitions of the experiments under identical conditions.

### 4.2 Results

The purpose of the experiment is to quantify the effects of using a broker on the mobile device for coordination and dissemination of contextual information collected at or required by the mobile device. The outcome of the experiments is focussed on measuring the average energy consumed by each context request (including associated response from a provider). We calculate this average by dividing the total energy consumed during an experiment run by the number of context requests made during that run.

**Effect on Battery Consumption.** Figure 2a shows the comparison of energy consumption on the mobile device with and without utilising the mCxB. The experiment is repeated with 100, 1000, 2000 and 5000 queries exchanged between all the device and network based CxCs and CxPs. All queries are satisfied with a valid response from the appropriate CxP. In this case, network availability is 100%, i.e. no disconnection takes place during the experiment. The results show that the use of mCxB only results in energy conservation after relatively extended periods of use. Initially, for a small number of queries, the broker results in increase in energy usage at the device. This increase is directly related to addition of a new process in the experiment (mCxB) whose advantages are not yet fully utilised. e.g. because the broker cache has not sufficiently filled up to provide locality of reference benefits. Additionally, with 100% network availability the CxCs and CxPs do not spend any idle or busy waiting for network connectivity to become available. With the increase in number of queries over time, the caching feature of the mCxB saves significant energy by satisfying a portion of queries locally from cache instead of initiating network communication. The cache hit rate in our experiments varied from 17-20%. This hit rate is marginally less than that of around 35% we have reported in our earlier work [7]. The reason for this is that our earlier experiments are based on a network CxB with greater resources and a larger number of CxPs and CxCs registered with it.




**(a)** 100% network availability **(b)** Varying levels of network availability

**Fig. 2.** Average energy consumption at the device per context request

**Effect on battery consumption with varying network availability.** The benefits of utilising a mobile broker become pronounced during scenarios where network connectivity is intermittent. The chart in Fig. 2b shows energy conservation in scenarios where network availability varies from 100% to 50%. This experiment is carried out with 1000 queries each from network and device based context consumers. Because `CxCs` and `CxPs` registered with the `mCxB` do not need to monitor network connectivity during periods of network unavailability, a portion of their execution cost is saved. The `mCxB`, which is responsible for routing queries and responses to and from the network for its clients, manages the network communication and combined with the local caching facility it provides significant energy conservation.

**Energy Conservation with IPC.** In this section we discuss the improvements in energy conservation during context exchange when device level components use IPC to communicate amongst each other rather than HTTP based communication. Our primary reason for providing native IPC based communication facility, in addition to HTTP based communication between device level components (`CxCs`, `CxPs` and `mCxB`), is efficiency. Efficient exchange of messages or processing of inter-process calls can be achieved by using a communication mechanism that is light-weight and uses least resources (e.g. creation of threads to process individual requests, allocation of buffers in the memory).

Our analysis (see Fig. 3) has revealed that on our reference implementation platform (Android SDK), native IPC calls between two processes take an order of magnitude less time to complete than HTTP and TCP/IP Sockets based calls. The difference in performance of IPC, HTTP and TCP/IP Sockets based communication mainly arises due to their different semantics and amount of implicit I/O operations, i.e. I/O operations not requested explicitly by the process using the facility, e.g. creation and allocation of memory buffers and creation of separate threads to process an invocation. The semantics of IPC, TCP/IP Sockets and HTTP are largely standardised and do not differ by much between different versions of the Android SDK, i.e. results of the same experiment on a Google

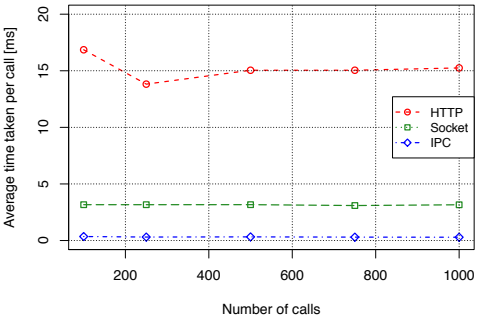
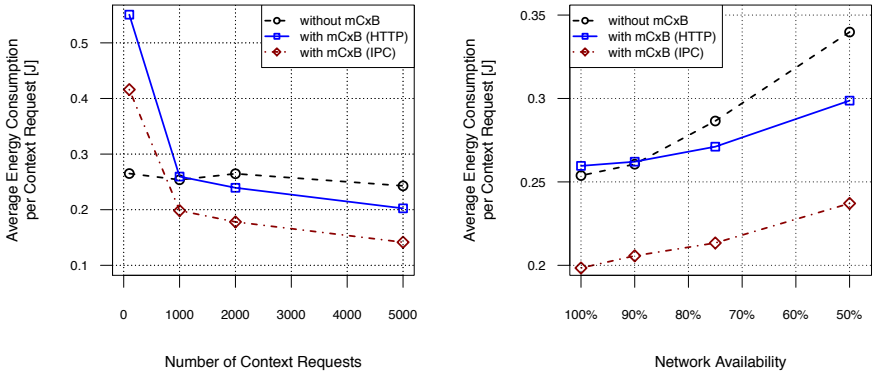



**Fig. 3.** Average completion time of calls between two processes with varying communication mechanisms



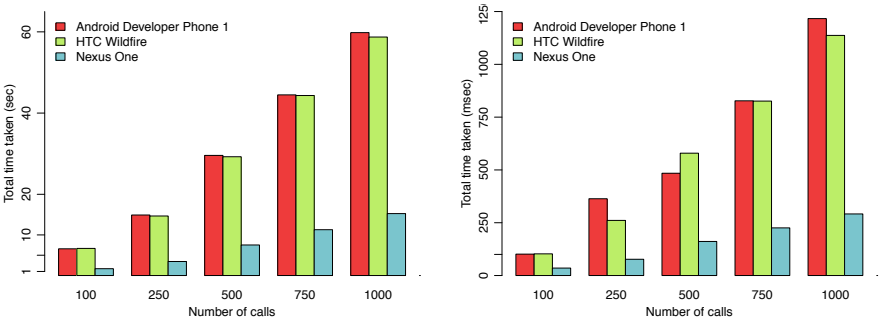

**(a)** 100% network availability **(b)** Varying network availability

**Fig. 4.** Average energy consumption comparison, without `mCxB`, with `mCxB` (HTTP) and with `mCxB` using IPC

Nexus One phone based on Android SDK version 2.1 only differ by a maximum +/-2% from those of the same device based on Android SDK version 2.3.

Based on this observation, we hypothesise that using IPC for communication between `CxCs`, `CxPs` and the `mCxB` executing on mobile devices will result in reduced energy consumption. In order to ascertain the validity of this hypothesis, we repeat the three experiments discussed in the previous section and enforce IPC based communication in between mobile device based `CxCs`, `CxPs` and the `mCxB`. Communication with network based `CxCs`, `CxPs` and the `mCxB` continues to take place over HTTP. Figure 4a shows the results of our experiments when using IPC based communication.

It is evident that using IPC as the communication mechanism between device based `CxCs`, `CxPs` and the `mCxB` further reduces the average energy consumption per context request-response call. These energy conservation benefits of the `mCxB` based interaction on the mobile device are pronounced in both scenarios, i.e. with




**(a)** Comparison of total completion times of HTTP based calls.

**(b)** Comparison of total completion times of IPC based calls.

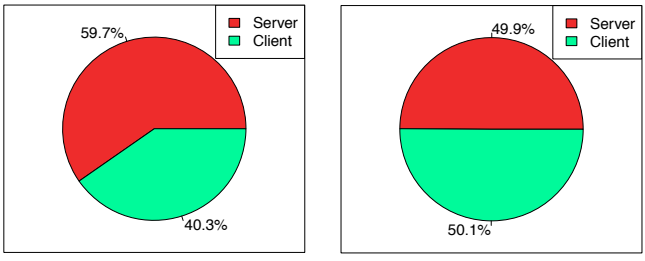
**Fig. 5.** HTTP vs. IPC comparison of total time taken to process a given number of calls between a client and a server process on different Android devices

full network availability (Fig. 4a) and varying network connectivity conditions (Fig. 4b). In addition to intrinsic differences in the operating mechanisms that influence the resource utilisation by IPC and HTTP based communication, one of the major factors that influence the overall reduction in energy consumption is that IPC calls take less time to complete on average. Hence the components on mobile devices that utilise IPC spend less waiting time and are able to carry out their application life cycles more efficiently. The performance difference between different communication mechanisms are also pronounced on different types of devices as well. The comparison between total call completion times of IPC, Sockets and HTTP based communication mechanism on Android based devices of varying capabilities is shown in Fig. 5. The effective difference between devices most relevant to the varying results is the processing power, e.g. Nexus One is based on a 1 GHz CPU while the other two devices have a 528 MHz CPU.

A notable observation in our analysis of HTTP vs. IPC energy utilisation is the different ratio of energy consumed by the server and client processes during the calls, which is plotted in Fig. 6. In case of IPC, a slight majority ($\approx 60\%$) of the energy consumption is due to the server process (Fig. 6a) while in case of HTTP it is almost evenly distributed between the server and client processes (Fig. 6b). This observation is significant as it reinforces our utilisation of IPC between device based components to reduce the processing burden on `CxC` and `CxP` (client processes) and shift it to `mCxB` (server process).

## 5 Summary and Future Work

In this paper we have presented the concept of a mobile broker for context coordination and communication. The main motivation for developing this components arises from the fact that if each device level `CxP` and `CxC` were to handle broker-bound communication itself, not only the computation cost will increase




**(a)** IPC calls client/server energy consumption

**(b)** HTTP calls client/server energy consumption

**Fig. 6.** Ratio of energy consumed by client/server during IPC and HTTP calls

but also the development cost for new `CxP`s and `CxC`s. Given the inherent mobility and somewhat intermittent connectivity of mobile devices, each `CxP` and `CxC` may have to carry out extra life-cycle management tasks as well. To mitigate the effects of these issues, a mobile broker is utilised that works in federation with the context brokers deployed in the network infrastructure.

Experiments have been carried out to analyse the effects of utilising a mobile broker on energy consumption on Android based smart phones. The results have shown that mobile broker helps in reducing energy consumption by providing a caching facility (reduces network usage) and reducing execution cost of `CxC`s and `CxP`s during periods of network unavailability. Utilising IPC based communication between device level components further reduces the energy consumption by the components of our system.

Applications/services involved in context-awareness related functions are likely to incur a significant energy cost as they utilise various resources on a mobile device (sensors, network interfaces, CPU, storage, etc.) and consequently the overall user experience may be impacted. In this paper we have identified possible approaches towards energy conservation for mobile devices hosting context consumers and providers, however we are exploring further approaches to supplement the ones implemented in our prototype e.g. a conditional push communication from mobile broker to an external broker that is based on more varied factors than used in these experiments. These factors may include current device battery level, quota restriction on context request and event generation for applications, etc. The experiment and results reported in this paper are a work in progress. The limited scenario presented in this paper provides a benchmark for further evaluation and tests to be carried out in a real world deployment. For immediate future, we are targeting an expansion and repetition of these experiments over a wider set of devices, networks and complex query patterns in order to infer more generalised results.

## References

1. Web Services Brokered Notification (WS-BrokeredNotification), `http://docs.oasis-open.org/wsn/wsn-ws_brokered_notification-1.3-spec-os.pdf`

2. Common Object Request Broker Architecture, Version 3.1 (January 2008), http://www.omg.org/spec/CORBA/3.1
3. Buschmann, F., Meunier, R., Rohnert, H., Sommerlad, P., Stal, M.: Pattern-Oriented Software Architecture: A System of Patterns, vol. 1. John Wiley & Sons Ltd., West Sussex (1995)
4. Chen, H., Finin, T., Joshi, A.: An intelligent broker for context-aware systems. In: Adjunct Proceedings of Ubicomp 2003, pp. 183–184 (October 2003) (poster paper)
5. Fielding, R.T.: Architectural Styles and the Design of Network-based Software Architectures. Ph.D. thesis, University of California (2000)
6. Kiani, S.L., Knappmeyer, M., Baker, N., Moltchanov, B.: A federated broker architecture for large scale context dissemination. In: Proceedings of the 10th International Conference on Scalable Computing and Communications, Bradford (2010)
7. Kiani, S.L., Knappmeyer, M., Reetz, E.S., Baker, N.: Effect of caching in a broker based context provisioning system. In: Lukowicz, P., Kunze, K., Kortuem, G. (eds.) EuroSSC 2010. LNCS, vol. 6446, pp. 108–121. Springer, Heidelberg (2010)
8. Knappmeyer, M., Tönjes, R., Baker, N.: Modular and extendible context provisioning for evolving mobile applications and services. In: 18th ICT Mobile Summit (2009)
9. Knappmeyer, M., Kiani, S.L., Frá, C., Moltchanov, B., Baker, N.: A Light-Weight Context Representation and Context Management Schema. In: Proceedings of IEEE International Symposium on Wireless Pervasive Computing (May 2010)
10. Moltchanov, B., Mannweiler, C., Simoes, J.: Context-awareness enabling new business models in smart spaces. In: Balandin, S., Dunaytsev, R., Koucheryavy, Y. (eds.) NEW2AN/ruSMART 2010. LNCS, vol. 6294, pp. 13–25. Springer, Heidelberg (2010)
11. Mrohs, B., Steglich, S., Aftelak, A., Klemettinen, M., Salo, J.T., Cordier, C., Carrez, F.: MobiLife Service Infrastructure and SPICE Architecture Principles, pp. 1–5 (September 2006)
12. Zhang, L., Tiwana, B., Qian, Z., Wang, Z., Dick, R.P., Mao, Z.M., Yang, L.: Accurate online power estimation and automatic battery behavior based power model generation for smartphones. In: Proceedings of the 8th IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis, CODES/ISSS 2010, pp. 105–114. ACM, New York (2010)

# Complex Activity Recognition Using Context Driven Activity Theory in Home Environments

Saguna[1,2], Arkady Zaslavsky[1,2], and Dipanjan Chakraborty[3]

[1] Monash University, Melbourne, Victoria, Australia
[2] Lulea University of Technology, SE-971 87, Lulea, Sweden
[3] IBM Research, India Research Lab, New Delhi, India
saguna.saguna@monash.edu, arkady.zaslavsky@ltu.se,
cdipanjan@in.ibm.com

**Abstract.** This paper proposes a context driven activity theory (CDAT) and reasoning approach for recognition of concurrent and interleaved complex activities of daily living (ADL) which involves no training and minimal annotation during the setup phase. We develop and validate our CDAT using the novel complex activity recognition algorithm on two users for three weeks. The algorithm accuracy reaches 88.5% for concurrent and interleaved activities. The inferencing of complex activities is performed online and mapped onto situations in near real-time mode. The developed systems performance is analyzed and its behavior evaluated.

**Keywords:** Activity recognition, context-awareness, situations.

## 1 Introduction

Activity recognition in smart home environments focuses on the activities of daily living (ADL) such as cooking, housework, eating, office related work, grooming, etc. ADLs are the focus of several research areas such as health care, aged care, emergency, security and comfort [1-3]. The activities performed by human users are highly complex and multi-tasking comes naturally. Each ADL has more than one sub activity. For example, eating breakfast may involve sitting on chair, picking a knife, picking a fork, etc. We call these sub activities as *atomic* activities and define them as those unit level activities which cannot be broken down further given application semantics. Also, these atomic activities are performed as operations which are routinized [4]. Figure 1 shows typical ADLs performed by human users. These ADLs may interleave or occur concurrently when human users multi-task. Atomic activities may not always follow the same sequence within an ADL.

All human activity recognition problems somewhat follow a common approach. This includes the use of sensors, multi-sensor fusion, the segmentation of data with relation to time and space, feature extraction and selection, classification of activities, and the use of activity models to infer activities [5]. These models can be based on a number of techniques [5] such as machine learning and string-matching algorithms, human activity language and use of semantics. Activity models built for recognition of sequential activities do not perform well when activities are interleaved, occur concurrently or when sequence of atomic activities changes [3, 6, 7]. Also, the

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 38–50, 2011.

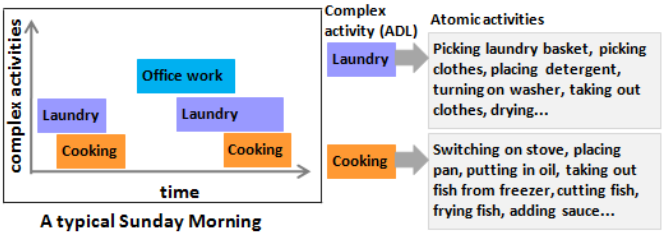



**Fig. 1.** ADLs performed by human users on a typical Sunday. Below we show atomic activities for two complex activities, Laundry and Cooking.

primary focus is on atomic activities. State of the art [3, 8] uses probabilistic models or data mining techniques to infer sequential ADLs. The inference of sequential and varied order activities which are concurrent and interleaved is also recently been addressed in [8, 9]. To summarize, these existing techniques suffer from several drawbacks. (1) Large amounts of data collection and data annotations are required for training of models, (2) Changes in sequence of ADLs affect activity recognition accuracy of trained models adversely, (3) Concurrent and interleaved ADLs are not accurately recognized as it is time consuming to train activity models for the different possible ways in which complex activities are interleaved or performed concurrently. In this paper, we focus on accurate and dependable recognition of complex ADLs[1] by addressing these issues and using context information. The use of context information has been emphasized in [2] for activity recognition. Our contributions are three-fold. Firstly, we propose and develop a context-driven activity theory (CDAT) which is used in defining atomic and complex activities. Secondly, we develop a novel context-driven algorithm to infer complex concurrent and interleaved activities without the need for training data and minimal annotation only during setup. Finally, we develop a novel situation- and context-aware activity recognition system (SACAAR) which can be used by ubiquitous applications. We validate our system using extensive experimentation.

This paper is organized as follows: Firstly, we present the SACAAR approach for activity recognition which gives the theoretical design of our system. This is followed by the SACAAR system architecture and its prototype implementation. In the next section, the experimental results are presented which validate our proposed approach. We, then compare our SACAAR system with related work on complex activity recognition. Lastly, we give the conclusion and future work.

## 2 SACAAR Approach to Activity Recognition

This section presents our proposed approach to infer complex ADLs. In [10], activity information is considered to be a context. But in activity analysis domain, our aim is to facilitate the inference of activities using sensors and other context. In figure 2, we show how activities and situations are related in the activity analysis domain. Firstly, context and sensors directly related to inferencing atomic activities are used, for

[1] In the rest of the paper, ADLs are referred to as complex activities.

example, one or more accelerometers are used for atomic activities like sitting and walking. Also, certain context information such as 'speed of user' may further validate whether user is walking or not. Further, a number of atomic activities together form a complex activity.

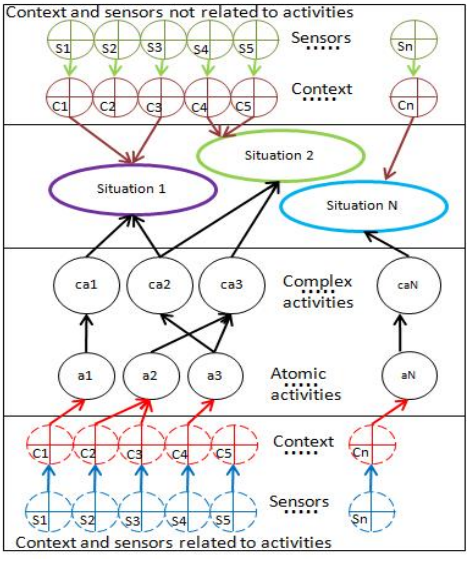



**Fig. 2.** Sensors(s), context(c), atomic activities(a), complex activities(ca), situations and their relationships

Also, of interest here is the concept of situations which can play an important role in activity inference. Situations are defined as "*a set of circumstances in which an entity may find itself*" [2] where an entity can be a human or object. Human users in a day are part of a number of situations. All activities that are performed by them are part of these situations. We are referring to situations which are inferred by context information not sourced by activity related sensor data but other context for example, location, time, temperature, weather conditions, light on/off. By inferring situations we can limit the number of activities to be recognized. In a situation hierarchy, situations and sub-situations exist and it is possible to identify situations with a limited set of sensor and context information. For example, a simple high level situation is "user is in office room" detected by user's location sensor. This can help in activity inference since the user will perform only those activities which are linked to office room and not the kitchen.

We use the context spaces theory [11] to infer situations and then use this knowledge to further infer activities. A complex activity can be linked to more than one situation, for example, "eating lunch" can be performed in the "office room" or on a day off from work in the "dining room at home". Also, a complex activity can occur in time such that it traverses through more than one situation during its completion, for example, "getting ready and going to work" can belong to situations "leaving home", "out on the road" and "near office". The situations can be inferred from spatio-temporal information as well as other context information.

### 2.1 Theoretical Design of SACAAR System

In this section, we propose and investigate a context-driven activity theory (CDAT) where activity could be complex or atomic. We further develop a context and atomic activity reasoning approach to infer and reason about complex activities.

[2] http://www.wordreference.com/definition/situation

**DEFINITION 1:** *Atomic activity, A is defined as activity which can be observed from a set of sensors,* $\Sigma_S=\{s_1,...,s_n\}$, where $n \geq 1$, where the level of granularity can vary based on sensor deployment scenarios. For example,

**Case 1:** $A_1$ = "user is walking" inferred by sensor $s_1$ which is an accelerometer placed on the user's waist. Case 2: $A_1$ = "user is walking" inferred by sensors $s_1$, $s_2$, $s_3$ and $s_4$,where $s_1$, $s_2$, $s_3$ are three accelerometers placed on different parts of the body and $s_4$ is a GPS sensor placed on the user which provides individual body part movements and velocity information.

**DEFINITION 2:** *Complex activity*, CA is a tuple CA=($\gamma A$, $\rho C$, $A_S \vee^3 C_S$, $A_E \vee C_E$, $T_S$, $T_E$, $T_L$) where $\gamma A$ is the sub-set of atomic activities from the complete set of atomic activities, $\Sigma_A=\{A_1,..., A_n\}$, $n \geq 1$ and $\rho C$ is the sub-set of context information from the complete set of context information, $\Sigma_C=\{C_1,...,C_n\}$, $n \geq 1$ which must occur for a complex activity CA to occur (where the order within $\gamma A$ is not important), $(A_S, A_E) \in \Sigma_A$ and are the start and end atomic activities respectively, $(C_S, C_E) \in \Sigma_C$ and are the start and end context information respectively. CA can have multiple start and end atomic activities as well as context information, $T_S$ and $T_E$ denote the start time and end time of a complex activity. $T_L = |T_E - T_S|$ is the complex activity lifespan and $T_{Lmin} < T_L < T_{Lmax}$ where $(T_{Lmin}, T_{Lmax})$ gives the time range for a complex activity, for example,

**Case 1:** $CA_1$ = "user is walking to bus stop" is inferred by atomic activity, $A_1$ = user is walking and context information, $C_2$ = user direction towards bus stop from home. Case 2: $CA_2$ = "user is working on presentation at home" is inferred by atomic activities, $A_2$ = user is sitting, $A_3$ = user is using study desk, $A_4$ = user is detected near laptop, $A_5$ = user is typing in Microsoft Powerpoint application file and context information, $C_3$ = user location is home, $C_4$ = study desk light is on. We further look at an example of concurrent and interleaved activities. Case 3: A user performs the following activities concurrently and interleaved in time. $CA_3$ = "user is reading/writing a document on his/her laptop in room at office", $CA_4$ = "user is browsing the Internet for research articles", $CA_5$ = "user is drinking coffee", $CA_6$ = "user is chatting on IM with friend". We can further perform operations on complex activities such as unions to create complex activities such as $(CA_3 \cup CA_4 \cup CA_5)$.

We do not consider order of atomic activities while performing complex activity inference though if required, it can be accommodated in Definition 2 by changing $\gamma A$ to a fixed-ordered list. Fixed order of atomic activities is important in industry environments or workshop settings, in real life it varies considerably.

*Context and Atomic Activity Reasoning to Infer Complex Activities*: To infer complex activities we propose reasoning of context and atomic activity information. Each complex activity has a list of atomic activities, $\gamma A$ and a list of context information, $\rho C$ as mentioned in Definition 2. Table 1 gives some examples to explain our reasoning approach. Each atomic activity, $A_i$ and context information, $C_i$ is assigned a particular weight, $w_{CA_k}^{A_i}$ and $w_{CA_k}^{C_i}$ respectively, corresponding to its importance in relation to the occurrence of a complex activity $CA_k$. The sum of all the weights, $\omega_{CA_k}$ for each $CA_k$ is 1. If $A_i$ or $C_i$ do not occur for a particular $CA_k$ then $w_{CA_k}^{A_i} = 0$ and

[3] $\vee$ implies the OR operation.

$w_{CA_k}^{C_i} = 0$. The sum of all the weights, $\omega_{CA_k}$ for all *occurring* atomic activities and context information needs to be above a threshold, $\omega_{CA_k}^{T}$ in order for $CA_k$ to occur successfully. If the sum of weights, $\omega_{CA_k}$ is less than the threshold, $\omega_{CA_k}^{T}$ then 1) it implies that the activity was started but abandoned in between, 2) it implies that the core set of atomic activities and context information for that particular $CA_k$ did not occur. Thus,

$$\omega_{CA_k} = \sum_{i=1}^{N} w_{CA_k}^{A_i} + \sum_{i=1}^{n} w_{CA_k}^{C_i} \quad (1)$$

where, $0 \leq \omega \leq 1$ and

$$\sum_{i=1}^{N} w_{CA_k}^{A_i} = (w_{CA_k}^{A_1} + \cdots + w_{CA_k}^{A_N}) \quad (2)$$

and

$$\sum_{i=1}^{N} w_{CA_k}^{C_i} = (w_{CA_k}^{C_1} + \cdots + w_{CA_k}^{C_N}) \quad (3)$$

and $w_{CA_k}^{A_i} = 0 \; and \; w_{CA_k}^{C_i} = 0$, if $A_i$ and $C_i$ do not occur for $CA_k$ and

$$\omega_{CA_k} \geq \omega_{CA_k}^{T} \quad (4)$$

for any $CA_k$ to have occurred successfully. We further demonstrate the use of weights which helps in complex activity recognition by checking the occurrence of key atomic activities using the following examples from table 1.

Example 1: $CA_1$ = Cooking omelette for breakfast in kitchen is a complex activity as shown in table I. We use our CDAT along with our reasoning approach to define and infer $CA_1$. We define the $CA_1$= ($\gamma$A, $\rho$C, $A_S \vee C_S$, $A_E \vee C_E$, $T_S$, $T_E$, $T_L$ ) tuple based on domain knowledge in a way that it is straightforward for users to define the tuple themselves by using the type of atomic activities and context information available.

**Table 1.** Complex Activity Examples

| $CA_k$ ($\omega_{CA_k}^{T}$) | $\gamma$A ($w_{CA_k}^{A_i}$) | $\rho$C ($w_{CA_k}^{C_i}$) | Core $\gamma$A and $\rho$C | $A_S$ ,$C_S$ | $A_E$, $C_E$ | $T_S$ | $T_E$ | Time taken (mins) | $T_L$ range (mins) |
|---|---|---|---|---|---|---|---|---|---|
| Cooking omelette for breakfast in kitchen (0.89) | $A_1$:standing(0.01), $A_2$:walking(0.01), $A_3$:fridge(0.02), $A_4$:plate(0.02), $A_5$:eggs(0.16), $A_6$:frypan(0.14), $A_7$:vegetable drawer(0.02), $A_8$:bowl(0.12), $A_9$:whisker(0.12) | $C_1$:in home kitchen (0.12), $C_2$:morning (0.12), $C_3$:kitchen light on(0.03), $C_4$:stove on(0.12) | Core $\gamma$A = ($A_5$, $A_6$, $A_8$, $A_9$) and Core $\rho$C = ($C_1$,$C_2$, $C_4$) | $C_1$, $A_5$, $A_3$, $A_6$, $A_{10}$, $A_9$ | $\neg C_3$, $A_4$, $\neg C_4$ | 07:06 | 07:22 | 16 | 10-20 |
| Drinking coffee in office room (0.68) | $A_1$:coffeemug (0.34), $A_2$:sitting(0.16) $A_2$:walking(0.16) | $C_1$:in office room(0.34) | Core $\gamma$A = ($A_1$) and Core $\rho$C = ($C_1$) | coffee mug | $\neg$coffee mug | 18:15 | 18:45 | 30 | 20-45 |
| Working on a doc in office room (0.68) | $A_1$:sitting(0.16), $A_2$:typing word doc(0.34) | $C_1$: in office room(0.16), $C_2$:usingdesk (0.34) | Core $\gamma$A = ($A_2$) and Core $\rho$C = ($C_2$) | word doc | $\neg$word doc | 10:10 | 11:30 | 80 | 10-60 |

The assignment of weights is based on the importance of each $A_i$ and $C_i$ for a corresponding $CA_1$, for example, here the highest weight is assigned to $A_5 = 0.16$ followed by $A_6 = 0.14$ which are very important atomic activities for cooking an omelette. Also, $C_1$, $C_2$ and $C_4$ are assigned weights of 0.12 as they are all of equal importance in the process of performing $CA_1$ while $C_3$ may or may not be switched on depending on time of day. In this case, $\omega_{CA_k}$ should be greater than the threshold $\omega^T_{CA_k} = 0.89$ for $CA_1$ to have successfully occurred. Also, temporal context is part of the complete set of context information, $\Sigma_C$. ¬A and ¬C represents non occurrence of atomic activity and context information.

Similarly, for complex activities $CA_2$ = Drinking coffee in lounge and $CA_3$ = Working on a document in office can be reasoned and inferred as shown in table 1. We highlight that uncertain context or sensor information can also be dealt with by taking the probabilities of occurrence of atomic activities and multiplying with their assigned weights in Equation 1.

## 3 SACAAR System Architecture

The SACAAR system architecture consists of two layers as shown in figure 3. It can be deployed in different application domains which require complex activity recognition. The sensory layer of the system can handle different types of sensor information which is required to infer atomic activities. For example, accelerometers to infer body motion such as sitting, walking, standing, jogging, RFID reader and tags for object interaction such as picking a cup, picking a pan, touch sensors for opening a cupboard, opening a door, physiological sensors for mood and stress levels. Virtual sensors can include modules of code which gather device activity, browser activity and music player activity from a user's devices such as computers, laptops and smart phones. We infer three types of context information. Firstly context information that helps in improving activity inference for example, if sensor 1 (accelerometer) returns atomic activity walking, the confidence in this atomic activity can be increased by context information from sensor 2 (GPS) which provides the speed of the user. Secondly, context information which helps in inferring complex activities for example, in the above case sensor 2(GPS) can also give direction of the user between 2 points A and B. This helps in inferring complex activity such as user walking towards point B. Lastly, context information can be used to infer situations. The activity inference layer is used to infer activities and it consists of the following components:

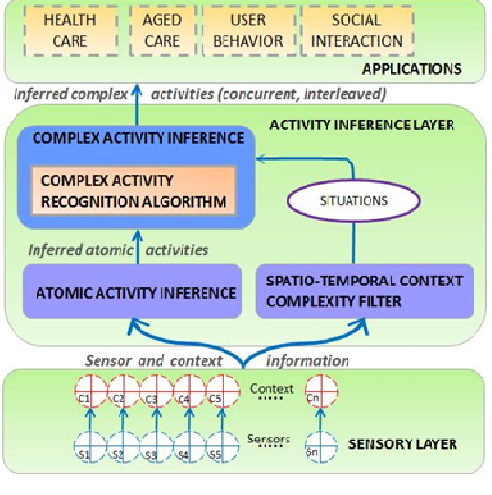



**Fig. 3.** SACAAR architecture

**Atomic Activity Inference Module**:
In AMOD, atomic activities are inferred. Body motion related atomic activities such as sitting, standing, walking, etc are inferred using decision trees [12]. A weighted voting sliding window mechanism is used to infer RFID object usage. A similar sliding window weighted voting mechanism is also applied to device and browser activity. Other context information such as speed and direction are used to ascertain activities as well.

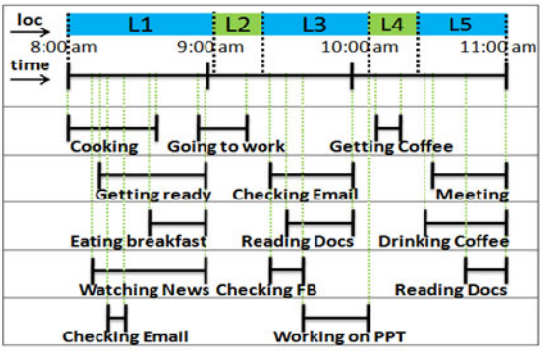



**Fig. 4.** Multiple complex activities inferred from 8:00 am to 11:00 am on a weekday (Day 12 for Subject 1 from our experiments) with locations L1=Home, L2=Outdoor/on the road, L3=Office room, L4=Office kitchen and L5=Meeting room. The labels for activities are generalized in some cases for brevity, for example cooking is a label for "cooking omelette for breakfast" and checking FB is social networking

**Spatio-temporal Context Complexity Filter:** Context information from the sensory layer is used by CCF for inferencing situations. This information is used in two ways by our SACAAR system architecture. Firstly, context information is used to infer situations. Activities always belong to some situation and linking them can help in inference of activities as well as enhance the richness of the inferred complex activity. These linkages can be created at run time and new activities can be added to situations in SACAAR system. We initially created situations and link our pre-defined complex activities to these situations during the setup phase. We then provided a mechanism to add activities to new situations at run time, if required. Secondly, when the links between activities and situations are known to be fixed, it can help to reduce the number of complex activities to look for during the inference process by inferring the situations first. Such situation inference can help in activity recognition by reducing battery consumption of certain sensors which are being polled to infer atomic activities which do not belong to the current situation.

**Complex Activity Inference Module:** The CAIM is used to fuse together different sources of atomic activities and combine them to infer complex activities with the help of the situation and activity linkages provided by the CCF. The complex activity recognition algorithm (CARALGO) is then used for inference of complex activities which can be concurrent and interleaved and is given in table 2. Figure 4 shows a snapshot of the inferred concurrent and interleaved complex activities from day 12 of our experiment 2 using subject 1 (see section 5). Some labels are generalized for brevity. It works by taking atomic activities, context information and situations as input and outputs the complex activities. The algorithm consists of finding the start atomic activity and then setting a time window of the size of the lifespan $T_L$ for each

matched $A_S \vee C_S$ belonging to a $CA_k$. We then look for matching γA, ρC and $A_E \vee C_E$ within the time window for each $CA_k$. We compute the $\omega_{CA_k}$ using equation (1) and then check against $\omega_{CA_k}^T$ as shown in equation (4). If the condition is matched $CA_k$ is inferred successfully.

**Table 2.** Complex Activity Recognition Algorithm (CARALGO)

**Algorithm 1:** Algorithm for complex activity recognition

**INPUT:** $A_i$, $C_i$, $S_i$ **; OUTPUT:** $CA_k$
**findStartAtomicActivity**($A_i$, $C_i$)
1.{ check for current $S_i$
2. **findComplexActivitiesList**($S_i$)
3. ***foreach*** ($CA_k$){
4. ***if***($A_i$== $A_S$)
5. add($CA_{list} \leftarrow CA_i$= (γA, ρC, $A_S$, $A_E$, $C_S$, $C_E$, $T_L$))
6. }//***foreach***
7. ***return*** $CA_{list}$. } **//end findStartAtomicActivity**
**findComplexActivity**($A_i$, $C_i$)
1. { ***foreach*** ($CA_{list} \leftarrow CA_k$) {
2. ***while***(time counter< $T_{Lmax}^{k}$){
3. ***if***($A_i$== element in γ$A_i$)
4. add $A_i \rightarrow \gamma A_i$ and recalculate $w_{CA_k}^{A_i}$
5. ***if***($C_i$== element in ρ$C_i$)
6. add $C_i \rightarrow \rho C_i$ and recalculate $w_{CA_k}^{C_i}$
7. ***if***(($A_E$,$C_E$ found for $CA_i$)&(ρ$C_i$ and γ$A_i$ are complete and $\omega_{CA_k} \geq \omega_{CA_k}^T$))
8. found $CA_k$ }//***while*** }//***foreach***
11. ***return*** $CA_k$. }//**end findComplexActivity**

All time windows run in parallel and all incoming $A_i$ and $C_i$ for each $CA_k$ are added to them till a successful match is found. The weights are added at runtime after each addition. We assign weights currently using domain knowledge but in the future we are looking at techniques to assign weights automatically.

## 4 SACAAR System Test-Bed and Prototype Implementation

We have built a test-bed comprising of several sensors. We place either Mulle v3 sensor[4] (accelerometer) or the Android Phone with its inbuilt accelerometer on the user's waist to detect body motion using decision trees. We chose the waist of the user as the most appropriate position [2]. A Bluetooth RFID reader was used. It was placed on the user's wrist to detect LF passive RFID tagged objects within a distance of 5-10 cm. Each tag is labeled in terms of an activity, and the RFID readings are inferred as atomic activities. The device and browser activity are inferred using a C++ TimeTracker[5] tool available openly on sourceforge. We extended this tool to incorporate it in our test-bed. The user's activity on the mobile phone is inferred using

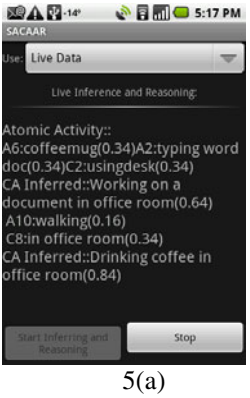
[4] http://www.eistec.se/
[5] http://ttracker.sourceforge.net/

a Java based code built by us specific to the Android platform. Location information is collected using GPS, Wi-fi positioning and RFID tags. We use the inbuilt GPS on an Android phone to gather the location, speed and weather information. We will provide the data and prototype online to share with those interested. Further details on test-bed can also be found in [13].

## 5 Experimental Results

We use our test-bed for testing and validating our proposed approach to recognize complex activities which are concurrent and interleaved. We conducted extensive experimentation and we present the results in this section. We perform experiments on two subjects for the duration of 21 days with an average of seven hours daily. The experiments were performed usually in the time range of 8:00 am to 12:00 pm and from 1:30 pm to 5:30 pm. We identified 16 complex activities (listed in table 3) and used our CDAT to define them. These definitions were stored in the SACAAR system. We gave our subjects an Android phone to record the activities manually which involved just adding a count for each occurrence of a complex activity in the corresponding hour. The user was asked to keep this record simply for measuring the accuracy of our algorithm. This information was not used for any annotation or training purpose. We show the accuracy of our algorithm for both subjects in table 3. Subject one performed all activities while subject two performed only 13 activities out of the 16. Our algorithm performed with an overall accuracy of 88.5%.

We performed online inferencing and users were given complete freedom to perform the previously defined complex activities in any interleaved and concurrent manner. We allowed users to change sequence of atomic activities as well. It is important to note that our approach does not require training data to create activity models. It only uses domain knowledge to define complex activities and create links to atomic activities. The activities in table 3 with format <activity@loc> always




5(a)

| CA(k) | 2 | 3 | 4 | 5 | 6 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 0 | 6 | 12 | 15 | 5 | 0 | 2 | 0 | 5 | 3 | 0 |
| 3 | 6 | 0 | 15 | 14 | 5 | 0 | 1 | 0 | 2 | 4 | 0 |
| 4 | 12 | 15 | 0 | 28 | 30 | 5 | 1 | 0 | 11 | 0 | 2 |
| 5 | 15 | 14 | 28 | 0 | 45 | 17 | 23 | 25 | 74 | 90 | 110 |
| 6 | 5 | 5 | 30 | 45 | 0 | 5 | 7 | 3 | 29 | 100 | 43 |
| 9 | 0 | 0 | 5 | 17 | 5 | 0 | 24 | 11 | 45 | 10 | 76 |
| 10 | 2 | 1 | 1 | 23 | 7 | 24 | 0 | 27 | 19 | 5 | 63 |
| 11 | 0 | 0 | 0 | 25 | 3 | 11 | 27 | 0 | 14 | 0 | 0 |
| 12 | 5 | 2 | 11 | 74 | 29 | 45 | 19 | 0 | 0 | 9 | 133 |
| 13 | 3 | 4 | 0 | 90 | 100 | 10 | 5 | 14 | 9 | 0 | 5 |
| 14 | 0 | 0 | 2 | 110 | 43 | 76 | 63 | 0 | 133 | 5 | 0 |

5(b)

**Fig. 5. (a)** SACAAR system running on Android Phone showing live inferencing of atomic and complex activities. (b) Matrix showing number of instances of interleaved and concurrent complex activities for 11 activities from table 3.

occurred at the same location while activities with format <activity> occurred at both home and office. Figure 5(a) shows SACAAR running on Android Phone with online inferencing. The snapshot shows atomic activities and context information with their corresponding weights which is followed by the inferred complex activity. To the best of our knowledge this is the first such work to infer complex activities using CDAT.

Figure 5(b) shows the activity pairs which had maximum concurrency and interleaving. We observe that ($CA_{14}$,$CA_{12}$) and ($CA_{14}$,$CA_5$) have the highest two values. $CA_{16}$ and $CA_{15}$ were the least interleaved activities with any other $CA_k$ (not shown in figure 5b). It is observed that mostly activities in the morning and in the

**Table 3.** Output instances of complex activity from subject one (U1) and subject two (U2) with the individual accuracy values. The complex activities, CA(k) comprised of the following: CA(1) = *"Getting ready @ home"*, CA(2) = *"Cooking omelette for breakfast @ home"*, CA(3) = *"Eating breakfast @ home"*, CA(4) = *"Watching news"*, CA(5) = *"Checking email"*; CA(6) = *"Social networking"*, CA(7) = *"Going to work @ outdoor"*, CA(8) = *"Getting coffee"*, CA(9) = *"Drinking coffee"*, CA(10) = *"Working on presentation"*; CA(11) = *"Meeting @ office"*, CA(12) = *"Read/Write docs"*, CA(13) = *"Watching videos online"*, CA(14) = *"Searching research articles"*, CA(15) = *"Jogging in the gym @ office"*, CA(16) = *"Walking in city @ mall"*.

| **Complex Activity $CA_k$** | **Correctly Inferred** | | **Incorrectly inferred or abandoned** | | **Accuracy (%)** | |
|---|---|---|---|---|---|---|
| | **U1** | **U2** | **U1** | **U2** | **U1** | **U2** |
| **1** | 16 | 12 | 3 | 2 | 84 | 86 |
| **2** | 8 | 10 | 3 | 1 | 73 | 91 |
| **3** | 12 | 12 | 2 | 2 | 86 | 86 |
| **4** | 18 | 16 | 2 | 4 | 90 | 80 |
| **5** | 100 | 74 | 29 | 3 | 78 | 96 |
| **6** | 60 | 45 | 10 | 3 | 86 | 94 |
| **7** | 12 | 14 | 6 | 1 | 67 | 93 |
| **8** | 30 | - | 6 | - | 83 | - |
| **9** | 25 | - | 8 | - | 76 | - |
| **10** | 20 | 33 | 3 | 3 | 87 | 92 |
| **11** | 16 | 22 | 2 | 9 | 89 | 71 |
| **12** | 75 | 85 | 7 | 6 | 91 | 93 |
| **13** | 70 | 60 | 6 | 4 | 92 | 94 |
| **14** | 98 | 100 | 4 | 11 | 96 | 90 |
| **15** | 8 | - | 0 | - | 100 | - |
| **16** | 5 | 4 | 0 | 1 | 100 | 80 |
| **Total** | 569 | 487 | 95 | 50 | 86 | 91 |

office are most interleaved and concurrent. We highlight that interleaving of complex activities is observed from the interleaving atomic activities belonging to different complex activities. Concurrency of complex activities is observed when complex activities occur within same time periods. Experimentation showed that the key to improving accuracy of inferring activities using CARALGO is the complex activity lifespan. We inferred situations linked to spatio-temporal context such as "at home in the morning", "outdoor early morning", "outdoor in the afternoon", "outdoor in the evening", "at office in the day", "in the kitchen in the morning", etc. Activities occurring in these situations were linked to them. This helped in the inference of activities as well as we ascertained which activities belonged to which situations. Any activity from our defined set when occurring in a situation not linked to it previously was recorded and the system was notified for later addition or deletion. We had 12 complex activities that were notified to be added to 5 situations which were not linked previously. We show in table 4, 5 such activities linking to 4 situations. We do not show all situations that were defined at setup and their activity links here due to lack of space. We plan to share the data collected as well as the activity model used for inferencing in this experiment with research community.

**Table 4.** Complex activity linked to situations ('X' denotes occurrence and '-' denotes not occurring) (S1 = Afternoon in office, S2 = Morning in city, S3 = Evening at home, S4 = Early morning at home).

| **Situations(S)→ Complex Activity↓** | **S1** | **S2** | **S3** | **S4** |
|---|---|---|---|---|
| Searching research articles | - | - | - | X |
| Watching online videos | X | - | - | - |
| Cooking omelette | - | - | X | - |
| Jogging in gym | X | - | - | - |
| Walking in city | - | X | - | - |

## 6 Related Work on Complex Activity Recognition

In [1, 2], activity recognition involves collecting and labeling large datasets along with training models using the collected data. SACAAR does not require training data for complex activity inference and only uses minimal domain knowledge to define activities. By using CDAT atomic and complex activities can be defined easily. Also, in existing research, once the activity models are created any addition or change in the steps of activities will require collecting data and training the model again while SACAAR helps in eliminating this tedious procedure. Another advantage of SACAAR is that it is used to infer complex activities which are interleaved and concurrent. Data mining techniques applied to concurrent and interleaved activities in [8] are interesting as they help in searching emerging patterns using frequent itemset mining and find abnormally growing patterns but we differ from this work fundamentally as we use our CDAT and additional context information as well as a reasoning approach to infer activities.

In [9], activities are discovered using a mining method called discontinuous varied-order mining and then clustering is performed to identify activities. For the inference of activities voted hidden Markov model is used and the relative frequencies from sample data are used for training. This would still require observing individuals for weeks to create clusters of activities and then human input to identify the clusters. We highlight that CDAT is still required to define activities. We differ from [9] as our approach uses additional context information as well as a reasoning technique which improves accuracy and helps in identifying key atomic activities for complex activities. Our work also differs from [8, 9]. We use wearable sensors which can facilitate the process of linking activities to users and in the future our SACAAR system can be used to recognize complex activities involving multiple users.

## 7 Conclusion and Future Work

In this paper, we propose, develop and evaluate SACAAR system which infers atomic and complex activities by mapping them onto situations. Firstly, we propose and develop a novel system which uses multiple sensors and context information to infer atomic and complex activities. Secondly, we presented our context-driven activity theory (CDAT) and complex activity recognition algorithm (CARALGO) which infers complex activities that are interleaved and concurrent in nature. Thirdly, we perform activity and context reasoning to infer complex activities. A test-bed and prototype are built to validate SACAAR and CARALGO. The experimental results show that activity inference with considerably high accuracy was achieved with minimal domain knowledge and without any training at complex activity level. Future work will extend our context-driven CDAT, reasoning approach and the SACAAR system to infer complex activities involving multiple users performing complex activities in same situations.

## References

[1] Philipose, M., Fishkin, K.P., Perkowitz, M., Patterson, D.J., Fox, D., Kautz, H., Hahnel, D.: Inferring activities from interactions with objects. IEEE Pervasive Computing 3, 50–57 (2004)

[2] Choudhury, T., Consolvo, S., Harrison, B., Hightower, J., LaMarca, A., LeGrand, L., Rahimi, A., Rea, A., Bordello, G., Hemingway, B., Klasnja, P., Koscher, K., Landay, J.A., Lester, J., Wyatt, D., Haehnel, D.: The Mobile Sensing Platform: An Embedded Activity Recognition System. IEEE Pervasive Computing 7, 32–41 (2008)

[3] Ferscha, A., Mattern, F., Tapia, E., Intille, S., Larson, K.: Activity Recognition in the Home Using Simple and Ubiquitous Sensors. In: Ferscha, A., Mattern, F. (eds.) PERVASIVE 2004. LNCS, vol. 3001, pp. 158–175. Springer, Heidelberg (2004)

[4] Kaptelinin, V., Nardi, B., Macaulay, C.: Methods & tools: The activity checklist: a tool for representing the "space" of context. Interactions 6, 27–39 (1999)

[5] Yang, G.-Z., Lo, B., Thiemjarus, S.: Body Sensor Networks. Springer, London (2006)

[6] Albinali, F., Davies, N., Friday, A.: Structural Learning of Activities from Sparse Datasets. In: Fifth Annual IEEE International Conference on Pervasive Computing and Communications, pp. 221–228 (2007)

[7] Davies, N., Siewiorek, D.P., Sukthankar, R.: Activity-Based Computing. IEEE Pervasive Computing 7, 20–21 (2008)
[8] Tao, G., Zhanqing, W., Xianping, T., Hung Keng, P., Jian, L.: epSICAR: An Emerging Patterns based approach to sequential, interleaved and Concurrent Activity Recognition. In: IEEE International Conference on Pervasive Computing and Communications, pp. 1–9 (2009)
[9] Rashidi, P., Cook, D., Holder, L., Schmitter-Edgecombe, M.: Discovering Activities to Recognize and Track in a Smart Environment. IEEE Transactions on Knowledge and Data Engineering 23(4), 527–539 (2011)
[10] Dey, A.K.: Providing architectural support for building context-aware applications, PhD Thesis, Georgia Institute of Technology, p. 240 (2000)
[11] Padovitz, A., Loke, S.W., Zaslavsky, A.: Towards a Theory of Context Spaces. In: Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops (2004)
[12] Bao, L., Intille, S.: Activity Recognition from User-Annotated Acceleration Data. In: Proc. Pervasive, Vienna, Austria, pp. 1–17 (2004)
[13] Saguna, S., Zaslavsky, A., Chakraborty, D.: CrysP: Multi-Faceted Activity-Infused Presence in Emerging Social Networks. In: Balandin, S., Dunaytsev, R., Koucheryavy, Y. (eds.) ruSMART 2010. LNCS, vol. 6294, pp. 50–61. Springer, Heidelberg (2010)

# Integration of Smart-M3 Applications: Blogging in Smart Conference

Dmitry G. Korzun[1,4], Ivan V. Galov[1], Alexey M. Kashevnik[2], Nikolay G. Shilov[2], Kirill Krinkin[3], and Yury Korolev[3]

[1] Department of Computer Science, Petrozavodsk State University (PetrSU)
33, Lenin Ave., Petrozavodsk, 185910, Russia
{dkorzun,galov}@cs.karelia.ru
[2] Computer-Aided Integrated Systems Lab,
St.-Petersburg Institute for Informatics and Automation of RAS (SPIIRAS)
39, 14-th Line, St.-Petersburg, 199178, Russia
{alexey,nick}@iias.spb.su
[3] St.-Petersburg Electrotechnical University (SPbETU)
5, Professor Popov St., St.-Petersburg, 197376, Russia
kirill.krinkin@fruct.org, yury.king@gmail.com
[4] Helsinki Institute for Information Technology (HIIT),
PO Box 19800, 00076 Aalto, Finland

**Abstract.** Smart spaces provide a shared view of dynamic resources and context-aware services within a distributed application. There is, however, no standard scheme for integration of several independent applications. In this paper we analyze two particular smart space-based applications: Smart Conference system that assists conferencing process online and SmartScribo system that provides advanced access to the blogosphere. For this reference use case we propose a scheme of their integration that employs an agent to share knowledge between the origin applications. The initial implementation was demonstrated in the 9th FRUCT Conference and indicates the applicability of our approach.

**Keywords:** Smart spaces, Ontology, Smart conference, Blogging.

## 1 Introduction

Smart spaces allow a multi-device system to use a shared view of dynamic resources and context-aware services [6]. It supports the vision of ubiquitous computing: computers seamlessly integrate into human lives and applications provide right services anywhere and anytime [8]. Smart-M3 is an open-source information sharing platform for smart space applications [1]. A Smart-M3 application consists of distributed agents (knowledge processors, KP) running on various computers and sharing the smart space. This paper considers the problem of integrating several Smart-M3 applications when they enhance own functions based on composition of their smart spaces. We focus on two independent systems: smart conference system (SCS) [3] and smart blogging system (SmartScribo) [9].

SCS intelligently assists complicated conference processes, automating the burdensome work of conference organizers. It maintains the visual content

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 51–62, 2011.

available for participants: a current presentation slide from the conference projector and up-to-date session program from the conference whiteboard. Presenter changes the slides directly from her/his mobile device. SCS supports automatic time management functions and some other useful online control functions.

SmartScribo provides access to the blogosphere through its shared presentation as a smart space. Blog content and some other knowledge of user's current interest are kept in the smart space, which is shared with other user's devices, applications, and friends. In parallel, other bloggers may access the blogosphere directly. The user can benefit from advanced functions, like searching relevant blogs; this processing can be delegated to dedicated machines.

Our reference use case is an extension of the SCS functionality with the blogging features of SmartScribo. It supports online discussions of conference events. Participants use mobile or other appropriate devices for clients to the smart spaces. The conference smart space keeps all knowledge related to the conference. The participants discuss the conference program, a talk, a slide of a talk, etc. Authors provide answers and further details based on the feedback.

For this feature we enhance SCS such that there is a post in the conference blog for each talk. A post starts a discussion (questions, opinions, answers from the participants including the authors). In the enhancement, SmartScribo shares blog content from the blog service into the blogosphere smart space. Blog discussions are accessible via SmartScribo clients. The key problem is coordination between the conference space, blog service, and blogosphere space.

The main contribution of the paper is the following.

1. Distributed multi-device multi-agent multi-space architecture for integrating SmartScribo blogging features into SCS. It supports discussions between the conference participants, even if the conference program is changing.

2. Ontological model. Content of each of the given two smart spaces is structured with own ontology. The overlay ontology defines the structure of data flows between the origin spaces.

3. Integration scheme. An additional SmartScribo KP blog processor becomes a reader of the conference space reflecting promptly the conference state in the blogosphere space.

The rest of the paper is organized as follows. Section 2 lists some existing systems that automate conferencing. Section 3 introduces our architecture of integrating SmartScribo into SCS. Section 4 describes our ontological model for the required composition of smart spaces. Section 5 provides the implementation details of our integration scheme. Finally, Section 6 summarizes the paper.

## 2 Related Work

Information visualization system on distributed displays[1] automates the control on displaying information during business meetings. Information from a selected source is forwarded to a given display. The system supports meeting

[1] http://www.polymedia.ru/ru/isl/26/

notes, audiovisual mode management (in real time, by a pre-arranged scenario), data representation in business graphic forms. Meetings are held in pre-designed rooms. It requires a large amount of expensive audio and video equipment; its users must be qualified to operate with many complicated functions. The system is for small "all-in-one room" meetings and there is no mechanism to automate online discussions between the participants.

AM-conference management system[2] is used for automating support of meetings carried out in factories, offices, or workgroups. Participants can physically reside in different rooms. The central server requires high computing capabilities and a lot of associated software. The system covers most of laborious work in meeting preparation and holding, especially for large enterprise-level meetings. Nevertheless, its complicated functionality requires much training for users. There is no mechanism for topic-based discussions.

The system of virtual meetings[3] is for organizing virtual meetings of geographically remote employees. It keeps user information and documents in a distributed storage, making the system more reliable and stable for network failures. System services are ontology-based, providing interoperability for software subsystems from participants and organizers. The system supports instant messaging between employees; more advanced kinds of discussion are not available.

Park Place Installations[4] focus on appropriate auxiliary equipment for conferencing. It includes video projector, flat panel television (for presentations or news), computer (interface for presentation content), DVD player, VHS player, speaker phone, video conference, electronic whiteboard (for capturing ideas), microphones, amplifiers and speakers, and lighting control.

TNN Telecom[5] is a solution for integrating equipment of a smart conference room. It includes projector (or projectors), LCD/Plasma, projection screen rolled (electric or manual), compatible sound and speakers system, audio and video systems, telephony networking, IP television, as required, conference sound systems, video conference systems, lighting, air condition, appropriate shading (curtains), switching and control equipment. Voice and video can be recorded with automated lights and curtains management. The solution does not automate the core conference organization processes.

Bell Conferencing Solutions[6] support teleconferences. For the audio conferencing only a phone line and a phone is required for each participant. In the conference call, all participants can join the discussion, just as if they were seated around a boardroom table. Web conferencing uses a browser and phone to talk, see, share information, and work collaboratively with others. Webcasting provides consistent messaging via the internet to shareholders, customers, employees, and other business partners. Video conferencing merges several meetings via the internet, approximating to the case of traditional face-to-face meetings.

[2] http://www.auto-management.ru/sector/management/management_21.html
[3] http://www.anbr.ru/page.php?name=sivis&lang=1
[4] http://www.parkplaceinstallations.com
[5] http://www.tnn.co.il
[6] http://www.conferencing.bell.ca

## 3 Architecture and Scenarios

A smart space can be thought as consisting of heterogeneous devices, services, and knowledge. In Smart-M3, services are implemented as KPs running on the devices. Knowledge is kept in an RDF triple store accessible via a semantic information broker (SIB). People access the knowledge via KPs running on end-user devices. This section describes the applications of our reference use case: SCS [3] and SmartScribo [9]. Figure 1 shows the architecture of their integration with the focus on participating devices and KPs.

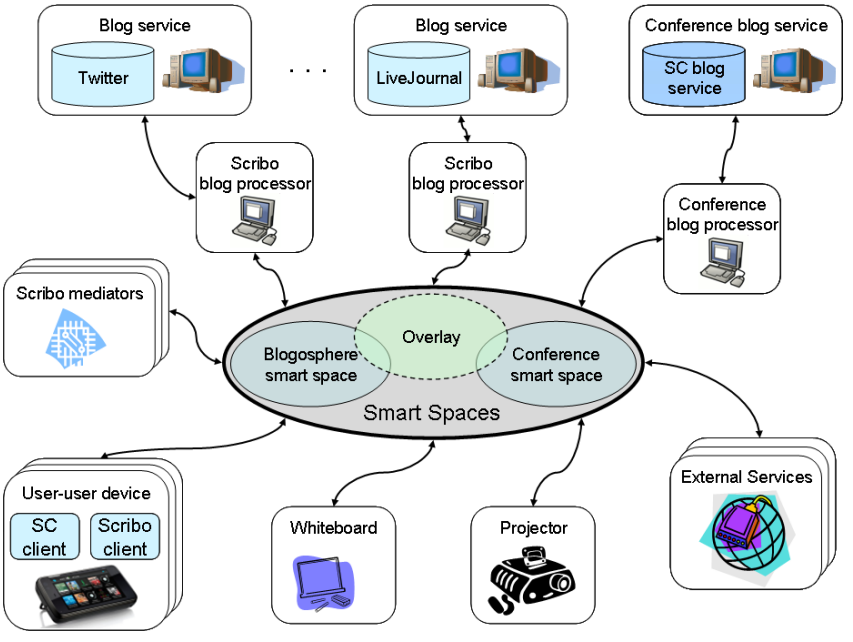



**Fig. 1.** Distributed multi-device multi-agent architecture of SCS and SmartScribo. Open source code is at `http://sourceforge.net/projects/smartconference/` and `http://gitorious.org/smart-scribo`

### 3.1 Smart Conference System

The conference smart space keeps information about the conference participants, their profiles, links to their presentations, conference schedule, and other conference attributes. The system consists of the following KPs.

*SC client:* 1) displays the conference schedule on the user device, 2) controls the slides during user's presentation, 3) shows information about other participants and their presentations, 4) shows and modifies own user profile. Each conference participant installs the KP on his/her device.

*Whiteboard:* 1) displays the schedule on a conference screen, 2) sends notifications to the presenter when the time of his/her presentation is finishing,

3) cancels presentation if the presenter is absent, 3) allows the conference chairman to change the order of participants.

*Projector:* displays the current presentation on the conference screen. When Whiteboard changes the current presentation, Projector displays it on the conference projector. KP also changes the current slide of the presentation when the SC client changes it in the conference smart space.

External service can be connected to the system via appropriate KPs. An example is a translation service for native languages of the participants.

### 3.2 SmartScribo

SmartScribo collects blog-related knowledge in the blogosphere smart space, keeping blog discussions the user is interested in. The space reflects only a part of the whole blogosphere; the primary storage is on the blog-service side (LiveJournal, Twitter, etc.). SmartScribo consists of the following KPs.

*Blog processor (BP):* 1) retrieves blog data from its blog-service and publishes them into the smart space and 2) discovers new data in the space (from clients) and sends them to the blog-service. Each blog service requires a dedicated BP, which serves many clients.

*Blog client (BC):* accesses blog data in the smart space and visualizes them to the user in a tree-like structure. When the user writes blog messages the KP publishes them in the smart space. BC runs on end-user mobile device.

*Blog mediator (BM):* produces "meta-information" for blogging. For instance, BM can implement a search or recommendation system for finding blog discussions of the user most current interest. The class of corresponding devices depends on the computational complexity of the BM function.

Blog content on the blog service side may be changed by non-SmartScribo clients. For example, some bloggers use the service directly via web-browsers.

### 3.3 Integration Idea

In the blogging extension, SCS deploys its own blog service or uses a public one. Our implementation supports both variants and is oriented to the LiveJournal blog engine. The conference blog service is a primary storage for all blog discussions related to each conference.

For each conference a separate blog must be created in advance. It consists of posts, one per a talk/paper. Additional posts are for other discussion threads, e.g., for a discussion on the conference organization. This initial phase defines a structure for discussions. Then the conference moves to its regular online phase. Participants start SC and SmartScribo clients. The use of SC client is as before. Switching to the SmartScribo client opens the access to the conference discussion.

To connect the conference program and its discussion threads we assign one SmartScribo BP to be a mediator. This conference BP keeps tracks of the SC schedule in the conference smart space. Whenever a new element or update is appearing the BP reflects it in the blog service. The connection is unidirectional; the BP reads data from the conference smart space and transforms the program

into the blog structure. To diminish improper intervention from external users to the conference blog service, any post (or comment) that is a part of the structure is published on behalf of SCS (an abstract user with admin rights). All other users, even they access the service directly, may not modify the structure.

The connection between the blog service and the smart space is implemented as it is in SmartScribo. The conference BP includes the functionality of a typical SmartScribo BP. Hence, the conference BP is the center in the Y-network of SCS, conference blog service, and SmartScribo. The connection "blog service $\leftrightarrow$ SmartScribo" is bidirectional. Note that our solution allows two separate KPs to implement the above two connections. The conference BP internals are considered in Section 5, see also Fig. 5 for the Y-network therein.

## 4 Ontology

This section describes the base ontologies inherited from our two origin applications. Although all knowledge is stored in a common RDF triple storage, they use independent ontologies. Moreover, the knowledge bases are separated on the application level, i.e., the RDF triple sets are disjoint. Cooperation needs sharing certain knowledge, and we suggest a kind of overlay ontology on top of the base ones to serve as a bridge for knowledge exchange. Its core is notification ontology for synchronization of changes happening on both sides.

### 4.1 Conference Smart Space

Figure 2 shows the conference smart space consisting of three parts: user information manager, event manager, and projector manager. User information manager deals with user profile, displays presenter's video and slide thumbnail, monitors time intervals of the conference, and displays conference schedule.

*User profile* consists of user information (e.g., name, photo, contacts, native language) and presentation (e.g., title, keywords, link to file with slides). It can also include links to video to be displayed during the presentation.

*Projector manager* generates thumbnail of the current slide. Then it is shared with end-user mobile devices. *Event manager* displays the conference schedule and manages time intervals. The schedule is created and dynamically adopted based on user profiles of available participants.

### 4.2 Blogosphere Smart Space

In general case, the blogosphere smart space is composed of personal spaces of bloggers, see details in [9]. This paper considers particularities for blogging in SCS. Its easiest variant employs a common SC blog account for every participant.

Figure 3 shows the division of knowledge in every personal space into the three subspaces: person (user profile), context, and blog data. *Person* describes user permanent or long-term data (e.g., name, age, e-mail, interests). They are not blog-specific and modeled with FOAF standard ontology and its extensions.

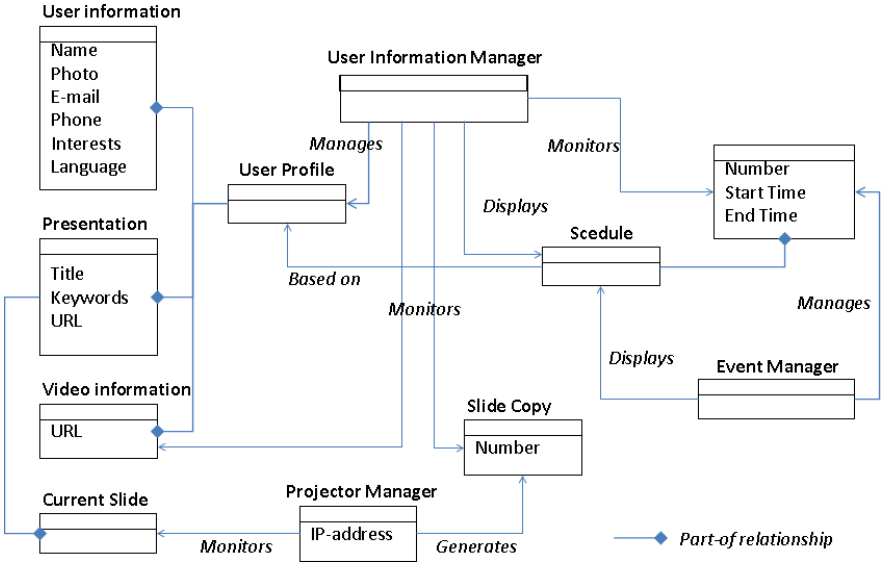



**Fig. 2.** Ontology for knowledge in the conference smart space

*Blog data* include service information and actual blog messages. In the general case, service information contains accounts from all services of the blogger. For SCS the only blog service is used. Person is linked with blog data using the FOAF concept "person has account". The blog consists of discussions structured with the SC program. Each discussion is a tree rooted with a post. Participants add comments to a post or to existing comments.

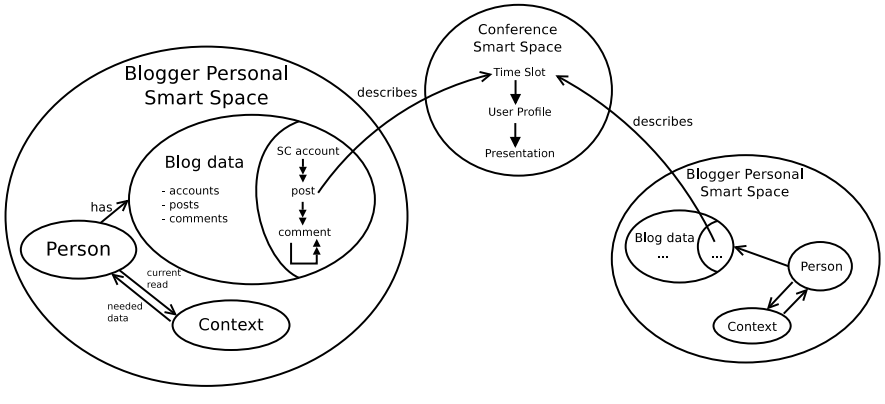



**Fig. 3.** SmartScribo blogger personal smart spaces and their relation with the conference smart space. There are many bloggers in SmartScribo; some of them can also become SC-participants

*Context* represents current and mutable characteristics of a person (e.g., current location, weather, music track, her mood). In the SC case, the context is for a common participant and can reflect the conference thematic and audience.

### 4.3 Notification Model and Overlay Ontology

Notifications initiate KP to execute some actions or to inform about execution result. A notification is implemented as a triple or triple set; KPs subscribe to notification triples. Whenever such a triple is changed the KP is notified. In the SC case, conference BP operates with several SmartScribo notifications, which are used in posts and comments manipulations.

This notification model supports proactive service discovery. When one KP (initiator) needs a service from another KP (responder) the former sends the notification triples. The responder KP activates the service due to the subscription and then publishes the result in the smart space. Since the initiator KP is subscribed to the triples with expected result, this KP immediately queries the data. Both KPs know the set of possible operations, expected results, and notification triples in advance.

Blog notifications are divided into the following groups: accounts, posts, and comments. An account notification requests the conference BP to perform session-based operations using account login and password, e.g., the BP logins the service on behalf of the blog client. The notification triple is "ServiceType—Operation—AccountInfo", where the object defines a type of the blog service (Live Journal, Twitter, WordPress, etc.), the predicate is the operation (login, refresh, etc.), and subject is an individual with account information. Posts and comment notifications are similar (send, receive, update, and delete operations).

Our notification model provides interaction between SmartScribo clients and conference blog service. Conference BP accesses two smart spaces using an overlay ontology. It provides "a bridge" that connects fragments from underlying ontologies, see Fig. 4. Conference BP tracks and synchronizes data using the one-to-one relation between *Post* class from SmartScribo and *TimeSlot* class from Smart Conference. There exists the only blog post for each time slot, see the link *describe* in Fig. 4.

## 5 KP-Based Integration Scheme

Conference blog processor deals with the problem of data integration from heterogeneous data sources: conference smart space, blogosphere smart space, and conference blog service. In this section we describe our solution to this integration problem. Its initial implementation was demonstrated and approved in real settings of the 9th FRUCT conference [4].

### 5.1 Smart Space Synchronization Approach

A sound approach for data integration is to employ a *domain mediator object* to synchronize data of different sources [2]. A *canonical information model* provides

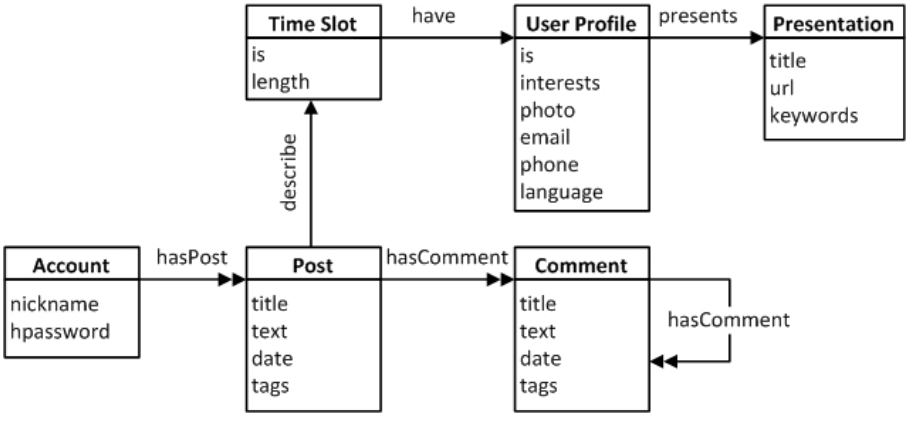



**Fig. 4.** Overlay ontology for the conference blog processor

a common language in semantic data presentation for different source information models. By means of the canonical model the mediator object requests and transforms data between different sources.

Our integration follows the domain mediator object approach: conference BP is an instance of such a mediator. Overlay ontology is a part a canonical information model. The latter deals with three data sources: conference smart space, blogosphere smart space and blog service. Our overlay ontology construction is based on the ontology matching technique, e.g., see [5]. An interesting problem is to apply other techniques like ontology unification; we leave this topic for our future research. A similar approach appeared in [7] for anonymous agent coordination in smart spaces.

Let $C$ be the canonical information model. We propose the following synchronization scheme. Smart spaces $S_1$ and $S_2$ are finite sets of knowledge objects $\{s_i^1\}$ and $\{s_j^2\}$, respectively. The synchronization rule is: $S_1$ and $S_2$ are synchronized if and only if $\exists c \in C$ such that $c \equiv s_i^1 \equiv s_j^2$. Hence $c$ is a canonical knowledge object. Note that we consider distinct spaces: each space uses own copy of $c$, leading to double store.

We assume that initially both spaces are synchronized. Then the simplest way is to start with empty smart spaces. Inserting a knowledge object is an atomic operation. A canonical object is created whenever its notification is received from one of the spaces. Before the object creation a check is performed weather the other space already keeps the pair object. If not, the object is created there. Note that this scheme requires data insertion only.

The scheme is applied for synchronizing the conference and blogosphere smart spaces with data structures on the conference blog service (LiveJournal). Conference BP manages two types of synchronizations: "SCS → LiveJournal" and "LiveJournal ↔ SmartScribo", see the Y network of the heterogeneous data sources and its mediator in Fig. 5.

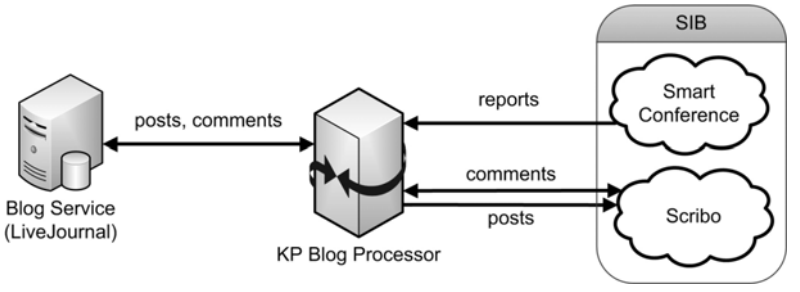



**Fig. 5.** Data flows of the conference blog processor

Synchronized knowledge objects are blog objects (posts and comments). A blog object has *text* and *title*, which are the same for both spaces, and unique universal identifier (UUID), which is space-dependent. The latter attribute needs special canonical presentation. Since UUID is a random numeric value we can use $\langle \text{LJ\_UID}, \text{SC\_UID} \rangle$ as the preimage for transformation function, where $\langle \text{LJ\_UID} \rangle$ is used to identify objects in the blogosphere smart space and $\langle \text{SC\_UID} \rangle$ is used to identify objects in the conference smart space.

Note that in our current design the conference space keeps presentations, which correspond to posts in the blogosphere smart space. Comments have no pair objects in the conference space.

### 5.2 Conference Blog Processor Implementation

Integration employs a special KP— a conference blog processor; it can access the smart spaces of Smart Conference and SmartScribo systems. In our initial implementation, conference participants have a common account for simplicity. The LiveJournal engine can be deployed on own conference server for privacy reasons and with the access for conference participants only.

Figure 5 shows the basic scheme that the blog processor follows to coordinate the integration. This KP performs the following actions: loading information about the conference schedule from the conference smart space; adding comments and new posts (report topics) into the blogosphere smart space; interacting with the conference blog service (LiveJournal) to keep posts and comment up to date; emitting (add/delete) notification records.

Table 1 summarizes the functional model. Currently the BP is implemented[7] in Qt (C++ interface) and contains the following main components: LiveJournalHandler, two SmartSpaceHandlers, and Synchronizer.

*LiveJournalHandler* interacts with the LiveJournal service by HTTP and XML-RPC protocols. It implements authorization and post&comments receive/send primitives. ConferenceHandler and ScriboHandler interact with SIB to access the two smart spaces.

[7] Open source code is available at `http://github.com/kua/scblog/`

**Table 1.** Functional model of a Smart Conference Blog Processor

| Data flow | Interaction with LiveJournal | Interaction with Smart Space | | Synchronization of different spaces |
|---|---|---|---|---|
| Module | LiveJournalHandler<br>Read / Write | ConferenceHandler<br>Read | ScriboHandler<br>Read / Write | Synchronizer<br>Local actions |
| Basic functions | 1. Receiving posts.<br>2. Sending post.<br>3. Receiving comments hierarchy of a post.<br>4. Sending comment. | 1. Loading report descriptions<br>2.Tracking changes in reports<br>3. Tracking changes in the schedule | 1. Loading information about comments<br>2. Tracking new comments from blog clients.<br>3. Tracking requests for data from blog clients<br>4.Sending notifications to blog clients | 1.Synchronization<br>2. Local storage of posts (reports) and comments based on common GUID<br>3.Store synchronized GUID map for items in different spaces |

*ConferenceHandler* interacts with the conference smart space. The latter is used as a knowledge source for information about reports; the KP does not modify the content of this space.

*ScriboHandler* interacts with the blogosphere smart space. This space is used (i) to publish posts and comments from the blog service, making them available to SmartScribo clients (conference participants) and (ii) as a source of comments from SmartScribo client, making them published in the blog service.

*Synchronizer* is responsible for synchronization between the conference blog service and the SIB of the two spaces. Objects for the synchronization are posts and comments (blog objects). Comments are structured hierarchically: each comment has a parent which is a post or previous comment. The blog processor uses unique identifiers for each of the two spaces.

The synchronization uses Map $\langle \texttt{ID}, \texttt{BlogObject} \rangle$, where ID is a structure that contains unique identifiers for each space in the form pair $\langle \texttt{LJ\_UID}, \texttt{SC\_UID} \rangle$. Based on this structure the synchronization algorithm determines what a particular blog object needs to be synchronized.

Post synchronization is initiated by receiving refreshPosts notification. Accordingly, the blog processor loads reports (time slot, author, topic and so on) from the SIB and converts the information into internal structures (in blog service format). The next step is coherence checking and adding absent topics. Synchronization of the blog service and SmartScribo is similar. When the operations have been completed, the blog processor creates refreshPosts notification.

Similarly, refreshComments notification is used for comment synchronization. In this process, the blog processor loads comments from the blogosphere smart space and from the Blog Service. Then BP adds absent pieces of data. Finally, BP writes refreshComments notification.

## 6 Conclusion

This paper considered the extension of smart conference functionality with blogging features for automating topic-based discussions between conference

participants. Our integration method consists of three key elements: 1) architecture, 2) ontological model, 3) integration scheme. The method is appropriate for integrating independent smart applications, when some features of one application can extend the functionality of another application. Based on the reference use case, our method can generalized to a class of integrations with more than two smart applications. Information sharing between their smart spaces is coordinated via a dedicated KP that applies an integration ontology on top of the ontologies from participating smart spaces.

**Acknowledgment.** We would like to thank Open Innovations Framework Program FRUCT for its support and R&D infrastructure. The Smart Conference System is a part of the project funded by grants 10-07-00368-a of the Russian Foundation for Basic Research: project #213 of the RAS research program "Intelligent information technologies, mathematical modeling, system analysis and automation") and contract #14.740.11.0357 of the Russian program "Research and Research-Human Resources for Innovating Russia in 2009–2013". SmartScribo is a part of the Karelia ENPI CBC Programme 2007–2013 project "Complex development of regional cooperation in the field of open ICT innovations".

## References

1. Honkola, J., Laine, H., Brown, R., Tyrkkö, O.: Smart-M3 information sharing platform. In: The 1st Int'l Workshop on Semantic Interoperability for Smart Spaces (SISS 2010) in Conjunction with IEEE ISCC 2010 (June 2010)
2. Kalinichenko, L.A.: Methodology of problem solving over multiple distributed heterogeneous information resources. In: Proc. Int'l Conf. Modern Technologies and IT Education. pp. 20–37. Moscow State University (September 2005) (in Russian)
3. Kashevnik, A., Valchenko, Y., Sitaev, M., Shilov, N.: Smart conference system. SPIIRAS Proceedings 14, 228–245 (2010) (in Russian)
4. Korzun, D., Galov, I., Kashevnik, A., Krinkin, K., Korolev, Y.: Blogging in the smart conference system. In: Proc. 9th Conf. of Open Innovations Framework Program FRUCT, pp. 63–73 (April 2011)
5. Masloboev, A., Lomov, P.: Common system thesaurus as the basis intellectual user interface for distributed semantic search system. Intellectual Systems 14(1-4), 4–10 (2010) (in Russian)
6. Oliver, I.: Information spaces as a basis for personalising the semantic web. In: Proc. 11th Int'l Conf. Enterprise Information Systems (ICEIS 2009). SAIC, pp. 179–184 (May 2009)
7. Smirnov, A., Kashnevik, A., Shilov, N., Oliver, I., Balandin, S., Boldyrev, S.: Anonymous agent coordination in smart spaces: State-of-the-art. In: Balandin, S., Moltchanov, D., Koucheryavy, Y. (eds.) ruSMART 2009. LNCS, vol. 5764, pp. 42–51. Springer, Heidelberg (2009)
8. Weiser, M.: The computer for the twenty-first century. Scientific American 265(3), 94–104 (1991)
9. Zaiceva, D., Galov, I., Korzun, D.: A blogging application for smart spaces. In: Proc. 9th Conf. of Open Innovations Framework Program FRUCT, pp. 154–163 (April 2011)

# Access Control at Triple Level: Specification and Enforcement of a Simple RDF Model to Support Concurrent Applications in Smart Environments

Alfredo D'Elia[1], Jukka Honkola[2], Daniele Manzaroli[1], and Tullio Salmon Cinotti[1]

[1] Arces, University of Bologna, Via Toffano 2/2, 40125, Bologna, Italy
{alfredo.delia4,daniele.manzaroli2,
tullio.salmoncinotti}@unibo.it
[2] Innorange Oy, 4 A 4, 00100, Helsinki, Finland
Apollonkatujukka@innorange.fi

**Abstract.** Smart environments support service innovation and in emerging approaches the information space involved is shared and accessible through simple primitives. Semantic web technologies play a crucial role in smart environments information representation, as they provide definitions allowing for interoperability at information level. The consistent interplay of multiple agents that concurrently access the knowledge base of an interoperable smart environment requires synchronization means like in traditional concurrent programming. This paper is focused on access control to synchronize concurrent access to shared resources of an RDF store in a multi-agent system. An RDF data model to semantically describe access rights at triple level is defined, an implementation to enforce this semantics on the RDF store is described and its performance are evaluated. Additional access control primitives can be implemented to support more complex behaviors.

## 1 Introduction

While the steady progress of miniaturization and networking capabilities of embedded systems contributes to the faster and faster implementation of Weiser's ubiquitous computing vision, lack of interoperability is still an obstacle to service innovation when the interaction between heterogeneous multivendor devices is required. Devices developed independently by different companies and for different purposes have different ways of representing dynamic information originated by their physical environment, and we may claim that multi-domain and multi-industry applications may only exist if information interoperability requirements similar to those assumed by the semantic web are met. This is also demonstrated by the fact that under the push of the semantic web [1], a growing number of new generation multi-agent platforms provide a shared knowledge base where information representation is machine interpretable and well defined from a semantic point of view [2]. Already at the beginning of the last decade early context management systems represented context information in terms of tuples, consisting of <subject, predicate, object> and possibly additional elements like timestamp, expiration time, privacy level indicator and so on

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 63–74, 2011.

[3, 4]. The agents had their internal logic and communication interface, and they interacted with the knowledge base by means of an implementation dependent set of primitives.

More modern platforms for smart environments applications adopt the same shared memory approach and enhance interoperability thanks to ontology driven, graph based semantic web data models. But, in order to seamlessly support evolving and situation-driven, highly dynamic applications, shared memory smart environments infrastructures need to include access control mechanisms at least to ensure exclusive access to selected pieces of information that need atomic updates.

Access control in digital systems has been deeply explored since the origin of computer science at all levels of abstraction of conventional computer architectures [10, 11]. Defining and enforcing a complete and general access control model optimized for the upcoming graph-based information stores is a complex task which goes beyond the scope of this paper.

Instead the attention here is focused on the definition of a simple extension of the semantic graph. Based on this extension, a powerful method to grant exclusive access to a small set of triples in an RDF store is proposed and its lightweight enforcement in a semantic platform for smart environment applications is described. The proposed method to grant exclusive access is based on a graph representation that is independent from the hosting platform or from the mechanism of enforcement so its ambition is to be general and flexible enough to be ready for further extensions in order to meet other challenging requirements including, for example, data confidentiality.

The need for the proposed semantic web "extension" is demonstrated in a mobile application taken by an industrial use case being developed within SOFIA[1].Here software agents run on maintenance operators personal devices and an activity has to be assigned to one of many competing operators. As it will be shown in section 5, this requires a locked transaction which includes several semantic transformations. The problem will be solved by temporarily granting to a single agent exclusive right to reassign several triples.

## 2 Motivation and Related Work

Access control mechanisms play a key role in many computer science areas. In concurrent programming they enable exclusive access to shared resources, as required by the consistent interplay of shared memory interacting agents, while in information security, for example, they provide solutions to handle "attributes", such as confidentiality, non-repudiability, integrity and availability [6].

Basically access control is required whenever access to specific resources needs to be restricted by a variety of rights. The desired level of protection may be achieved with appropriate models and primitives that always introduce a persistent performance penalty. Minimizing such penalty is a critical design requirement.

[1] The platform is being developed within the framework of a project of the European Joint Undertaking on Embedded Systems ARTEMIS. The project is called SOFIA (2009-11) and it is co-funded by the EU and by National Authorities including MIUR, the Italian Ministry for Education and Research. The platform interoperability component is open source, is called Smart-M3, it was originally proposed by NOKIA and it is described in Section 3.

In the area of security access control mechanisms have been investigated for over 30 years at all system architecture levels, from business application level to middleware, DBMS, operating systems and hardware. For example, most computer architectures enforce instruction level access control mechanisms to provide memory protection based on two or more privilege levels. Higher level layers may implement their access control and protection models relying on the mechanisms available at lower level.

At higher architectural level the discretionary models (DAC) [7] use a matrix relating access rights to the possible combinations of subjects and objects and the Bell LaPadula [8] model adds a mandatory check to partially solve the problem of Trojan horses. Lattice based access control (LBAC) allows the use of security labels organized in a lattice to enforce policies like the Chinese wall [9]. Many other important models were proposed and even if they share the same motivations, they differ in implementation and target domain.

In the area of concurrency management, the literature on access control to support interaction between concurrent agents goes back to the sixties and to the work of Prof. Djikstra at the University of Eindhoven, who set the foundations of operating Systems theory with his papers on concurrent programming and co-operating sequential processes [10, 11].

A comprehensive analysis of all access control models known in the literature is out of scope here, while it is more interesting to focus on the middleware level and, particularly, on the most recent ideas and discussions about access control in triple spaces and semantic smart spaces. In [12] an access control policy for semantic wikis is described. A semantic wiki is a software system which is similar to a normal Wiki like the well known Wikipedia, but additionally it provides metadata to describe page content and the relationships between pages. The idea claimed about access control in this particular semantic and information centric architecture is that a set of access rights and rules – also represented with semantic technologies – can be used to provide different views of the total knowledge, according to the access rights of the different subjects. A similar solution is also provided by Semantic Views [13], where a semantic description of views stored together with the core information, is used to offer different virtual perspectives of the available knowledge.

Moving to the smart environments scenario, a broad analysis of security aspects in smart environments is made in [14] and an architecture to handle security related mechanisms like access control, authentication, and, authorization and to manage security attributes like confidentiality is presented. Furthermore, [15, 16] point out that in smart environments software agents and security policies are context dependent, therefore policies should become context-centric from the current more common subject-centric model. Context based policies are related to a context ontology. By combining security policy rules with ontological reasoning, both the advantages of logic programming and description logic may be exploited.

RDF triple stores are context management systems made to store and manage RDF graphs i.e. list of subject-predicate-object triples in which subject and object are the endpoints of an arc while the predicate names the arc [24]. Not much literature is available on Access Control to RDF Store at triple level. Particularly, to the best of our knowledge, access control for mutually exclusive access to RDF subgraph has not been considered so far.

The research done in [17] suggests that a triple based information space calls for an access control model consisting of security objects associated to triple patterns. When an operation (e.g. a query) matches a triple pattern, authorization is granted or denied based on the associated security objects, which may specify pattern specific policies and security labels [9].

In [23] Access to an RDF store is controlled by an Access-Control Policies framework through which all the transactions are routed. Here the policies defined in the framework are used to determine whether to permit or prohibit the action on the store requested by the agent.

In this paper we address access control to RDF stores designed to support smart environment applications. The proposed model is defined at graph level as an RDF extension, and it is therefore inherently portable. No external framework is required as it can be easily enforced at primitive level. As performance test show the impact of the proposed model on the reference architecture is not high. The model represents access control policies in RDF stores and its first implementation is targeted at the synchronization scenario but the plan is to extend the semantic model to support more complex policies for all access primitives.

## 3 Reference Architecture

We have used the Sofia (Smart Objects for Intelligent Applications) project Interoperability Platform (IOP) [19] as the reference architecture for this work. An overview of the IOP functional and logical architecture is presented in Fig. 1. Sofia IOP provides an RDF [20] based information sharing platform for agents.

The semantic information broker (SIB) is the core of the system in which all the information known is stored. The information is stored as a set of RDF triples which constitute a directed labeled graph. Software agents, called knowledge processors (KP), may contribute to the evolution of this graph or may utilize its information by accessing the SIB through the smart space access protocol (SSAP) [21, 22]. We will call "smart space" the set of all the triples hosted by the SIB.

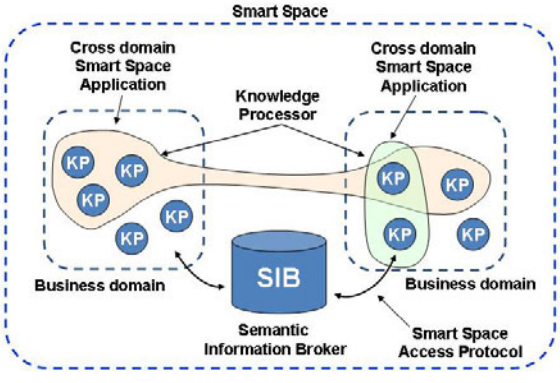



**Fig. 1.** Sofia IOP functional and logical architecture

The SSAP is the main integration point in the IOP. It defines the basic set of operations that KPs may perform on the triples. The operations and their informal descriptions are briefly the following:

- the *join* operation is necessary in order to be able to perform any other operation, its counterpart is the *leave primitive*.
- *insert*, *remove* and *update* operations modify the graph operating on its triples
- the *query* operation performs a query using any supported query language.
- The *subscri*be operation sets up a permanent query in the SIB, where the subscribing KP is notified about changes to the query results until the *unsubscribe operation is called*.

The Sofia IOP does not restrict the information stored in the SIB in any other way than requiring it to be syntactically valid RDF. However, it is obviously beneficial to store the information according to some ontology, described typically using OWL. This also provides a common structure for the information, namely that the information is represented as resources that are typed and have properties defined by the type(s) they are instances of. This work assumes that the information is modeled as resources with properties, but we do not necessitate the use of any ontology describing the stored information.

The nature of applications has been researched by doing several case studies. Combining information across several application domains has been investigated in [18]. The use of Sofia IOP as a coordinating layer for different services has been demonstrated in [5]. Other case studies have been described in [22].

The distributed nature of the applications also creates a need to coordinate access to the information when modifying it. This requirement has been encountered in several applications for example in service coordination [5] and in building maintenance case described in section 5, and solved with various mechanisms. In the next sections we present a generic mechanism that may be used in any application and provide building blocks for more complex synchronization protocols.

## 4 Data Model for Access Control Specification

This section describes the RDF data model to add access control information at triple level. Our objective was to be able to specify in a machine interpretable format the access restriction for an arbitrary set of triples. The resulting data model is represented in Fig.2.

The model provides a way to associate specific Access Restrictions to a triple pattern through an access restriction property named *AR_Property*. With reference to Fig.2, suppose that $KP^1$ wants to gain exclusive right to modify triples <*s, p,* *> where *s* = *I*, and *p* is one of the properties *P11* to *P1j*. In order to represent the access restriction in a machine readable format, a protection Entity *P* is attached to *I* through the predicate *AR_Property*. This is simply done with the insertion of the triple <*I, AR_property, P*>. Other statements having subject *P* and describing the access

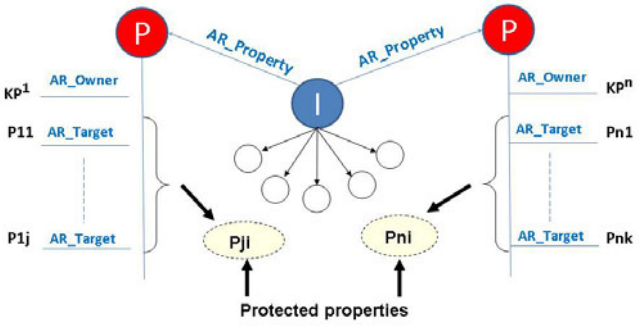



**Fig. 2.** RDF access control specification

restriction are inserted i.e. <*P, AR_Owner, KP*$^1$> will indicate which is the KP associated to that protection entity; statements <*P, AR_Target, P1i*> will indicate for which properties the protection is to be considered valid.

It should be noted that another KP (KP$^n$) may insert restrictions to other properties (say *Pn1* to *Pnk)* of the same entity *I*, that have not been protected by the KP$^1$. Fig. 2 shows the restrictions specified by KP1 (left) and by KPn right. In this case, according to the described semantics, only KP1 may update, remove or insert triples <*I, p11, some_object*> while only KPn has the right to modify remove or insert triples <*I, pn1, some object*>.

An arbitrary number of Protection Entities can be associated to each entity *I* in the Smart space: one for each different KP that wants to own a set of properties. The *P* entities are out of the domain of the *AR_Property* property so the access to them is managed according to ad hoc policies.

The proposed model presents both interesting properties and also some weak points. On one side, it is very simple as it was only introduced to solve some specific application problem in a precise scenario(see section V and VI), therefore it currently lacks the expressive power to address all possible policies of access restriction to RDF subgraphs. On the other side, thanks to RDF, it is inherently extensible and system agnostic; both of these qualities encourage its extension towards more powerful and expressive models.

Access control sub-graphs based on this model may be embedded in standard middleware data access primitives (insert, remove, update, query) and so hidden to the programmer, much in the same way as it occurs in conventional protected architectures at memory-read and memory-write instruction level. Similarly access control is carried out at every memory access event, but here a new target memory model is being considered, i.e. the graph, and access control information is stored in the graph itself and not in external structures as it happens in linear memory architectures. As it happens in conventional CPU architectures, embedding access control at primitive levels opens the way to performance optimized model implementations.

## 5 Synchronization of Interacting Knowledge Processors

In this section our access control data model is applied to synchronize the access of concurrent KPs in a real use case related to a building maintenance scenario. Concurrent KPs running on maintenance operators mobile devices are notified of a new maintenance intervention. All operators compete to get the job, but only one will get it, i.e. the first who accepts. At semantic level the winning KP has to modify the sub-graphs associated to the maintenance intervention instance, and this must be done with exclusive access otherwise smart space inconsistencies and unexpected behaviors may occur if other operators try to accept the job at the same time.

Fig. 3 shows graphically the RDF triples involved. The corrective intervention is an RDF-node created by some fault announcer KP as a reaction to an anomalous situation (fault). According to the announcer business logic, candidate operators with fault-specific skills are selected through the insertion of the triples with the *Sent_to* predicate. This insertion fires the notification to all candidates at the same time, thus they are given the opportunity to accept the job. When an operator accepts the job, the KP running on his device deletes all *Sent_to* predicates and inserts *Performed_by*. The deletion notifies all other operators that the job has been assigned to somebody else, and the insertion makes it possible for other actors of the maintenance chain to be informed and activated. This demonstrates how smoothly semantic platforms can support the interplay of heterogeneous interacting entities. But what happens in detail to the RDF knowledge base if two or more operators try to accept the job concurrently? If no exclusive access is granted to one KP at a time, there would be eventually more than one *Performed_by* predicate for the same *Corrective Intervention* instance, which is an inconsistent state. This would likely originate misbehaviors in other processes in the maintenance chain. This inconsistency occurs if a second operator accepts the job in the "critical" interval between the time when the first operator issues his acceptance and the time of the last notification that the job has already been assigned (i.e. last remove of the Sent_to property).

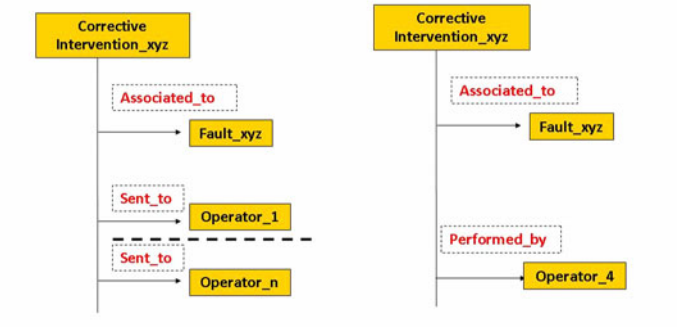



**Fig. 3.** Left: operators 1 to n compete for the job - Right: Operator 4 is the winner

If the proposed RDF data model is enforced, restricting access to the <*Corrective-Intervention-xyz, Performed_by,* * > solves the problem: only one KP at a time can have such restriction in place and during this time it can complete undisturbed its semantic transformations on the *Corrective_intervention-xyz* sub-graph. Then it will release the restriction.

A sequence diagram to provide a more general view on various possible situations with (c) or without (a,b) a mechanism for process synchronization is shown in Fig. 4 where all KPs are supposed to be subscribed to the triple pattern <s1,p1,*>. and when notified they may insert <s2,p2,o2> (in the maintenance scenario it is the *Performed_by*operation) which in turn may notify other KPs.

Three different situations are shown. In Fig. 4a, there is no access control enforced: KP1 is the only one process reacting to the notification; as no other process makes operations in the critical time, the overall behavior is correct but it is not safe. In fact, if two KPs perform the same operation during the critical time, unpredictable results may occur, as shown in Fig. 4b (KP1 and KP2, both receive a notification, but only one KP should perform the update).

Fig. 4c shows the behavior with access control in place: before updating the SIB, KP1 locks property *p2* (*Performed*-by it the maintenance scenario) and, when the concurrent process KP2 tries the same operation, it receives a Protection Fault due to the attempt to access to a locked pattern. Should KP2 try to do its sub-graph update without a prior protect request, it would still receive an access denied answer. KP1 releases its exclusive access right to the shared resource after receiving the notification of successful completion of the requested triple pattern update. It might be argued that the specific synchronization problem described above could also be solved without the use of the proposed access restriction facility. For example an Atomic Conditional Update (ACU) primitive would enable a straightforward solution. The ACU based solution solves efficiently only a small class of problems and its usage in more complicated scenarios might be difficult.

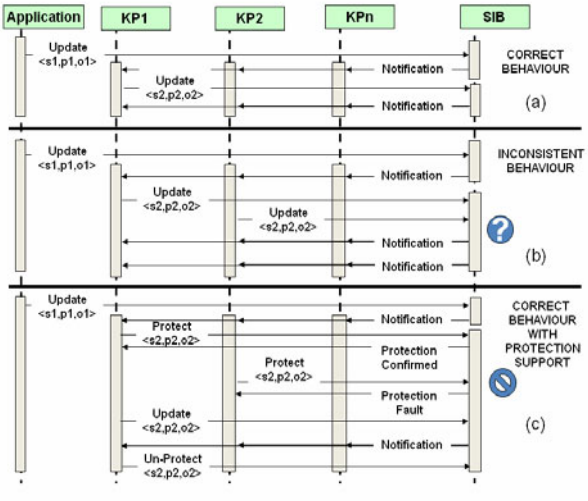



**Fig. 4.** Concurrent access to a sub-graph with (c) and without (b) access restriction in place

How could we deal for example with situations where more than one condition had to be verified in order to perform the update? In contrast, the proposed data model is capable of restricting access to an entire triple pattern, including triples that are still not explicitly stated, and can be simply extended to support more security attributes beside synchronization. For these reasons the ACU based approach was first considered and then abandoned.

## 6 Implementation and Performance

In this section the implementation in Smart-M3 of the access control policies enabled by our data model is discussed together with the resulting performance.

Only the SSAP primitives with write access are affected, i.e., with reference to Section 3: *Insert*, *Remove*, *Update*. The SSAP fragment processed by the policy enforcement algorithm is shown in Fig. 5: for each message the *Message-type* element tells if the request is in read or write mode, the *nodeID* element identifies the subject accessing the smart space and the *Triples* element represents the message content. The *Triples* elements and its children are processed and the request is accepted only if none of the *Triple* violates any of the access control policies. As we want to maintain the finest granularity level - i.e. that of the single triple - each *Triple* element has to pass the policy test. As shown in Fig. 5 the number of the *Triple* element can be high, moreover, depending from the query language used, more accesses to the SIB content may be required to be able to understand if a single triple violates the access control policies.

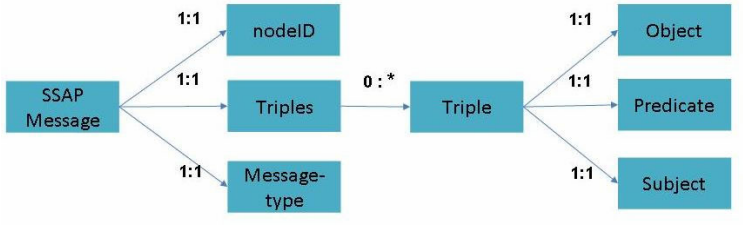



**Fig. 5.** SSAP fragment considered by the protection enforcement algorithm (schematic view)

Therefore the procedure to check a single triple should be fast in order to minimize the impact on system performance and should minimize the amount of accesses to the RDF store. The implementation is based on a dynamic table called Lock Cache Table (LCT), supporting the algorithm and evolving with the smart space access control related statements. The table has columns *I, P, Pi, KP*, which correspond to the nodes shown in Fig. 2. Each operation intended to add or remove protections updates the LCT (if allowed) while each triple of an SSAP message affected by the protection mechanisms is checked against each line of the table.

The software artifacts and the implemented algorithm provide Smart-M3 with basic access control functionality at triple pattern level with negligible impact on protocols and performance. Specifically, the platform access protocol (SSAP) runs unchanged because access restriction and release mechanisms are expressed in terms

of standard SSAP primitives (insert and remove). By not accessing the RDF store when permissions are checked, impact on response time is minimal as shown in Fig.6, where the top curve shows the insert time of fifty triples with a single primitive and the bottom curve shows the insert time of one triple. The offset between the two graphs is an indication of the insert time per triple, while their slope is an indication of the overhead introduced by the access control policy implementation.

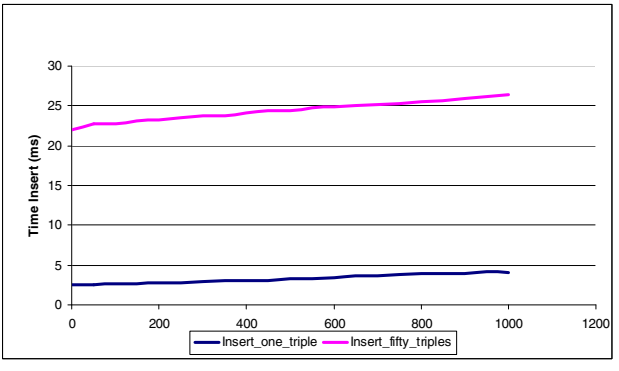



**Fig. 6.** Insert time for one and for fifty triples as a function of active protections

Each point of the graph was calculated over a mean of 50 samples and the test was run on Ubuntu Linux 10.4 with 1 GB of RAM and two P6 Intel processors at 3GHz. Even if the amount of comparisons required to check all the triples in the LCT is 50 times higher in the top graph with respect to the bottom one, still, the overhead added by our implementation is negligible for a small number of protections and it is still reasonble when the architecture is stressed by a large number of protections. The small slope of the graphs shows the scalability level of our implementation with respect to the number of active protections.

## 7 Conclusions

We have presented a data model to protect arbitrary properties of RDF instances in smart space RDF based triple stores. Naturally, the model adds some overhead when compared to a "plain" RDF, but the overhead is less than that incurred e.g. with a straightforward reification.

In this paper, we have described one use of our data model, which is for providing exclusive access to smart space resources, when multiple software agents try to access an RDF store simultaneously. The data model can as well be used to provide different security properties to the smart space information. However, this requires further research to analyze suitable policy description languages for this kind of setting.

One major benefit of using RDF for describing the protected properties is that the model is reflective, and the involved agents are able to see what has been protected. In security usage, this reflectivity may also be a drawback, but we believe that it is straightforward to utilize the current approach to hide the part of data model that

should not be public. Another benefit of our approach is that the model is general, and can be implemented on top of any software architecture that handles RDF. This is an important property for any system that is meant to be used in very different settings.

**Acknowledgments.** The authors thank the company CCC and particularly Valerio Nannini and Sandra Mattarozzi for providing the use case described in Section V.

The Interoperability Platform used has been developed within the framework of a project of the European Joint Undertaking on Embedded Systems ARTEMIS. The project is called SOFIA (2009-11), it is coordinated by NOKIA and it is co-funded by the EU and by National Authorities including the Finnish Public Authority TEKES and MIUR, the Italian Central Authority for Education and Research.

## References

1. Lee, T.B., Hendler, J., Lassila, O.: The semantic web. Scientific American (2001)
2. Context Broker Architecture, `http://cobra.umbc.edu/`
3. Smith, D., Ma, L., Ryan, N.: Acoustic environment as an indicator of social and physical context. Personal and Ubiquitous Computing 10, 241–254 (2006)
4. Ryan, N.: Smart environments for cultural heritage. In: Takao, U.N.O. (ed.) Takao UNO, Reading Historical Spatial Information From Around the World: Studies of Culture and Civilization Based on Geographic Information Systems Data, International Research Center for Japanese Studies, Kyoto (2005)
5. Luukkala, V., Binnema, D., Börzsei, M., Corongiu, A., Hyttinen, P.: Experiences in Implementing a Cross-Domain Use Case by Combining Semantic and Service Level Platforms. In: Proceedings of 1st Workshop on Semantic Interoperability in Smart Spaces (2010)
6. Savolainen, P., Niemelä, E., Savola, R.: A Taxonomy of Information Security for Service-Centric Systems. In: 33rd EUROMICRO Conference on Software Engineering and Advanced Applications, pp. 5–12 (2007)
7. Lampson, B.W.: Protection. In: Proc. Princeton Symposium on Information Sciences and Systems, pp. 437–443. Princeton University, Princeton (1971); reprinted in: Operating Systems Review 8(1), 18–24 (1974)
8. Bell, D.E., LaPadula, L.J.: Secure Computer Systems: Mathematical Founda-tions and Model. National Technical Information Service (Spring 1973)
9. Sandhu, R.: Lattice-based access control models. IEEE Computer 26(11), 9–19 (1993)
10. Dijkstra, E.W.: Solution of a problem in concurrent programming control. Communications of the ACM 8(9) (1965)
11. Dijkstra, E.W.: Co-operating sequential processes. In: Genuys, F. (ed.) Programming Languages, pp. 43–112 (1968)
12. Dietzold, S., Auer, S.: Access control on RDF triple stores from a semantic wiki perspective. In: Scripting for the Semantic Web Workshop at 3rd European Semantic Web Conference, ESWC (2006)
13. Manjunath, G., Sayers, C., Reynolds, D., Venugopal, K.S., Mohalik, S.K., Badrinath, R., Recker, J.L., Mesarina, M.: Semantic Views for Controlled Access to the Semantic Web. In: HP Technical Reports, Laboratories HPL (2008)

14. Suomalainen, J., Hyttinen, P., Tarvainen, P.: Secure information sharing between heterogeneous embedded devices. In: Proceedings of the Fourth European Conference on Software Architecture: Companion, ECSA (2010)
15. Montanari, R., Toninelli, A., Bradshaw, J.M.: Context-based security management for multi-agent systems. In: Proc. of the 2nd IEEE Sym. on Multi-Agent Security and Survivability (MAS&S 2005), pp. 75–84 (2005)
16. Toninelli, A., Kagal, L., Bradshaw, J.M., Montanari, R.: Rule-based and ontology-based policies: toward a hybrid approach to control agents in pervasive environments. In: Proc. of the Semantic Web and Policy Workshop (SWPW), in Conj. with ISWC 2005, Galway, Ireland (2005)
17. Jain, A., Farkas, C.: Secure resource description framework: An access control model. In: Proceedings of the Eleventh ACM Symposium on Access Control Models and Technologies (SACMAT 2006), pp. 121–129 (2006)
18. Honkola, J., Laine, H., Brown, R., Oliver, I.: Cross-Domain Interoperability: A Case Study. In: Balandin, S., Moltchanov, D., Koucheryavy, Y. (eds.) NEW2AN/ruSMART 2009. LNCS, vol. 5764, pp. 22–31. Springer, Heidelberg (2009)
19. SOFIA project – Smart Objects for Intelligent Applications, http://www.sofia-project.eu
20. RDF Semantics, http://www.w3.org/TR/rdf-mt/
21. Smart-M3 Open Source Project, http://sourceforge.net/projects/smart-m3
22. Honkola, J., Laine, H., Brown, R., Tyrkkö, O.: Smart-M3 Information Sharing Platform. In: Proceedings of 1st Workshop on Semantic Interoperability in Smart Spaces (2010)
23. Reddivari, P., Finin, T., Joshi, A.: Policy based Access Control for a RDF Store. In: Proceedings of the Policy Management for the Web Workshop, A WWW 2005 Workshop, W3C, pp. 78–83 (May 2005)
24. Lassila, O.: Programming Semantic Web Applications: A Synthesis of Knowledge Representation and Semi-Structured Data. Doctoral dissertation (October 2007) ISBN 978-951-22-8985-1

# Increasing Broker Performance in Smart-M3 Based Ridesharing System

Alexander Smirnov[1], Alexey M. Kashevnik[1], Nikolay G. Shilov[1], Harri Paloheimo[2], Heikki Waris[3], and Sergey Balandin[4]

[1] St.Petersburg Institute for Informatics and Automation, Russian Academy of Sciences, 14 Line, 39, 199178 St.Petersburg, Russia
{smir,alexey,nick}@iias.spb.su
[2] Nokia Corporation, Keilalahdentie 2-4, 02150 Espoo, Finland
Harri.Paloheimo@nokia.com
[3] Coreorient Oy, Kokkosaarenkatu 4 A 10, 00560 Helsinki, Finland
Heikki.Waris@coreorient.com
[4] FRUCT Oy, Kissankellontie 20, 00930 Helsinki, Finland
Sergey.Balandin@fruct.org

**Abstract.** Ridesharing is one of the most cost effective alternative transportation modes with high potential in reducing the greenhouse gas emissions and decreasing the amount of traffic in the streets. The paper extends the earlier presented approach to building sustainable logistics system for ridesharing support based on the idea of smart spaces, where various electronic devices can seamlessly access all required information distributed in the multi-device system from any of the devices. In particular, the paper describes broker as a main component of the system and algorithms used for its performance optimization. The Smart-M3 open source platform was chosen as a basis for the proposed approach.

**Keywords:** Smart Spaces, Smart-M3, ridesharing.

## 1 Introduction

Ridesharing (also known as carpooling, lift-sharing and covoiturage), is the shared use of a car by the driver and one or more passengers, usually for commuting. It is one of the most cost effective alternative transportation modes. Ridesharing has minimal incremental costs because it makes use of vehicle seats that would otherwise be not occupied. This approach allows getting lower costs per vehicle/mile even comparing to the public transport because it does not require a paid driver and avoids empty backhauls. However, currently Ridesharing is generally suitable only for trips with predictable schedules such as commuting or attending special events. Dynamic ridesharing (also known as instant ridesharing, ad-hoc ridesharing, real-time

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 75–86, 2011.

ridesharing or dynamic carpooling) is a special type of a ridesharing enabling formation of carpools on a nearly "real time" basis. Typical for this type of carpooling is: arrangement of one-time trips instead of recurrent appointments for commuters; the usage of mobile phones for placing carpooling requests and offers through a data service, automatic and instant matching of rides through a network service.

In accordance with Global GHG Abatement Cost Curve v 2.0 [1] in the travelling sector the carbon emission can be significantly decreased (10.5% of road transport[1]) via more efficient route planning, driving less, switching from car to rail, bus, cycle, etc. Additionally, the fact that the transport sector is 95% oil dependent makes it vulnerable to the expected rise in oil price during this decade [3, 4]. As a result, evolving of flexible, ecological and energy efficient logistics systems can be considered as one of the significant steps towards the knowledge-based Green ICT applications in low carbon economy. Regarding the above, a growing number of initiatives have been developed in this area (e.g. Fig. 1 represents activity of major vehicle manufacturers in the carsharing market). Carsharing assumes renting cars for one ride. This principle differs from the ridesharing, however, the figure clearly represents the growing demand and interest in this area, especially in Europe, where daily travel distances are relatively short.

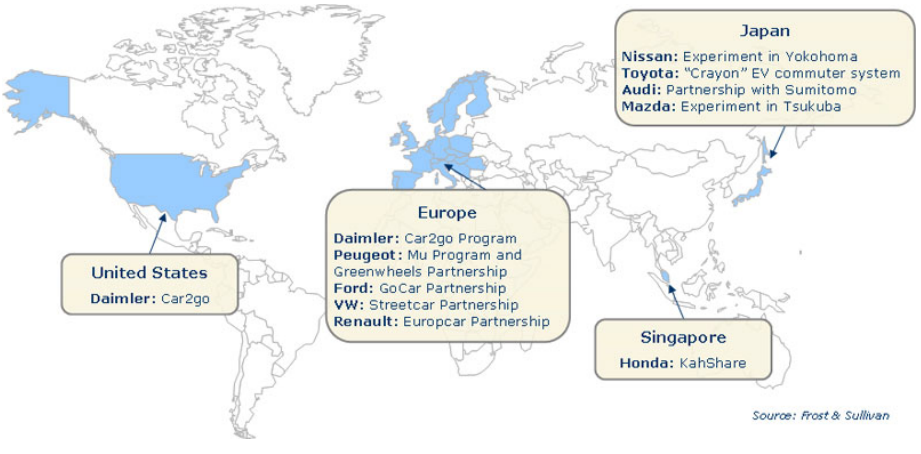



**Fig. 1.** Major vehicle manufacturers in the carsharing market (World), 2009

Modern ICT makes it possible to combine several ideas, which would result in a more flexible and efficient transportation systems. The main idea of the proposed approach is to develop models and methods that would enable configuration of resources for decision support in ad-hoc sustainable logistics. The paper extends the research results presented at AIS-IT'10 [5] presenting the developed ontology and major components of the dynamic ridesharing system and FRUCT 2010.

[1] By 0,5 $GtCO_2e$ (gigaton of carbon dioxide equivalent) [1]; combined with [2] results in 10.4% of road transport or 7.5% of all transport.

The paper is structured as follows. Section 2 describes related work. The technological framework is described in Section 3. Logic of the Broker as a main component of the ridesharing system is presented in Section 4. Its performance is discussed in Section 5, and some experimental results are given in Section 6. Major results are summarized in Conclusion.

## 2 Related Work

Several surveys have been proposed, which discuss such topic [e.g., 5-8]. They present a large amount of such kind of systems ranging from free local company or university ridesharing system to complex commercial non-free systems. The earlier ridesharing services existed at least in the 1970s when people could make appointments through a human operator.

Major advantages of ride sharing:

- Supposed the same amount of people travel in fewer cars, so fuel consumption, noise emissions, green house gas emissions and air pollution affecting the local environment decrease.
- In areas with little public transport an efficient ridesharing service can increase the speed of traveling.
- Congestion would be less frequent and so the macro economical costs of congestion would go down.
- As people travel together there's an increasing of social capital. On one hand it gives people the possibility to learn to know people who they would never have met otherwise.
- As less cars are on the road the risk for external people involved, for instant pedestrians of cyclists, in an accident goes down.
- The ameliorated connectivity, especially in rural areas, leads to economical growth as the markets become bigger.
- Because of ridesharing fewer people will buy new cars, the used cars will drive more kilometers per year and thus lead to a higher replacement rate resulting in more fuel-efficient and safer cars.

Major disadvantages of ride sharing:

- When carpooling, it becomes difficult to run errands on the way to and from the common locations.
- Tends to be complicated to reliably organize and is difficult to maintain, due to changing travel patterns and needs. Sign mark locations outside of their metro stops and large bus stations where drivers can share rides with other passengers in an orderly fashion.

To increase the efficiency of the ride sharing the following recommendations can be formulated:

- For a ride sharing system functioning it is needed to get a "critical mass" of the quantity of drivers, otherwise the efficiency will be extremely low. In order to solve this problem the proposed technology should be as easy to use and has to

provide high quality of service. The following recommendations related to this issue can be formulated:

- A ride query shouldn't take more than some seconds to enter.
- User's profiles are needed for the keeping rider's preferences.
- Current situation have to be taken into account during the matching process.
- For increasing ride matching probability the passengers can use two or more cars consequently during one trip.

## 3 Technological Framework

Fig. 2 represents the generic scheme of the approach. The main idea of the approach is to represent the logistics system components by sets of services provided by them. This makes it possible to replace the configuration of the logistics system with that of distributed services. For the purpose of interoperability the services are represented by services using the common notation described by the ontology.

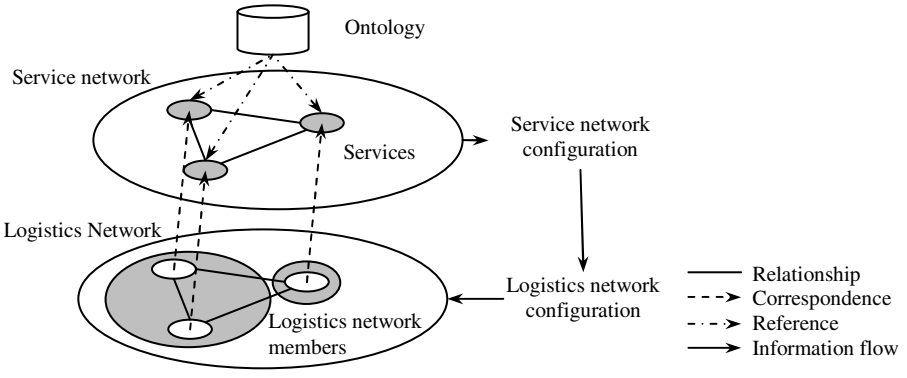



**Fig. 2.** Generic scheme of the approach

For the purpose of implementation the Smart-M3 platform is used [10, 11]. The key idea in Smart-M3 is that devices and software entities can publish their embedded information for other devices and software entities through simple, shared information brokers. The understandability of information is based on the usage of the common ontology models and common data formats. It is a free to use, open source solution available in BSD license [12]. The approach presented in the paper is aimed at solving the above mentioned problems via usage of the Smart-M3 platform and underlying technologies.

Fig. 3 represents the architecture of the prototype. Knowledge processors (KPs) represent participants of the system usually running on mobile devices of the users integrated into a temporary (ad hoc) wireless network via such technologies as GSM or WiFi. Places are defined via coordinates or address/intersection. "Car" actually stands for any means of transportation, including family car, small car, or bicycle. The division of functionalities between the car KP and driver KP are yet to be defined.

The broker produces possible matches between transportation service requesters (User1 KP, User2 KP) and providers (User3 KP, User4 KP). These matches are used then for direct negotiation between KPs concerning the transportation service.

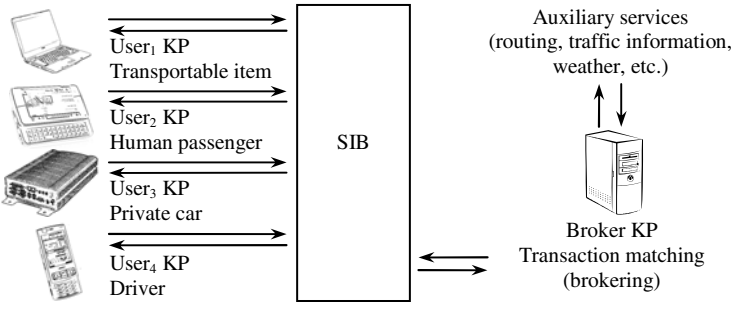



**Fig. 3.** Architecture of the sustainable smart logistics prototype

## 4 Broker Logic

The broker is the most loaded KP in the system. Below, the algorithm used inside the broker is considered in detail in order to estimate its computational complexity. Fig. 4 demonstrates the algorithm at the macro level. The numbered steps are described below in detail. The major idea of the algorithm is to reduce the search space step by step so that the less computationally intensive steps would be followed by more the more computationally intensive ones. Geometric distance is the least computational extensive – that is why it is used first. Steps 3-5 are buffer-based. We are currently working on vector-based heuristic, but this work is out of the scope of this paper.

Fig. 5 represents the input data for the algorithm (Fig. 4, step 1).

At step 2 of the algorithm the filtering by the geometric distance is performed. At this stage a significant number of possible meeting points (depending on the detour values for the passenger and the driver as well as the driver's path length) is eliminated. The computational complexity of this step is $O(m)$ (proportional to the number of points of the driver's path).

The idea of filtering by geometric distance is explained in Fig. 6. The passenger needs to get from point *PS* to point *PE*. The driver goes through points *D*1 … *D*9. The red circles' radius is the passenger's possible detour, the blue circles' radius is the driver's possible detour. For example, it can be seen that point *A* cannot be a meeting point since it is too far away from the passenger's start point, though it is close to driver's travel point *D*2. Points *B*, *C*, *D*, *E* could be the candidates for the meeting points for picking up the passenger and points *F* and *G* could be the candidates for the meeting points to drop off the passenger. The result of this step is the set of feasible meeting points (*MP*) and the appropriate points of the driver's route (*DP'*).

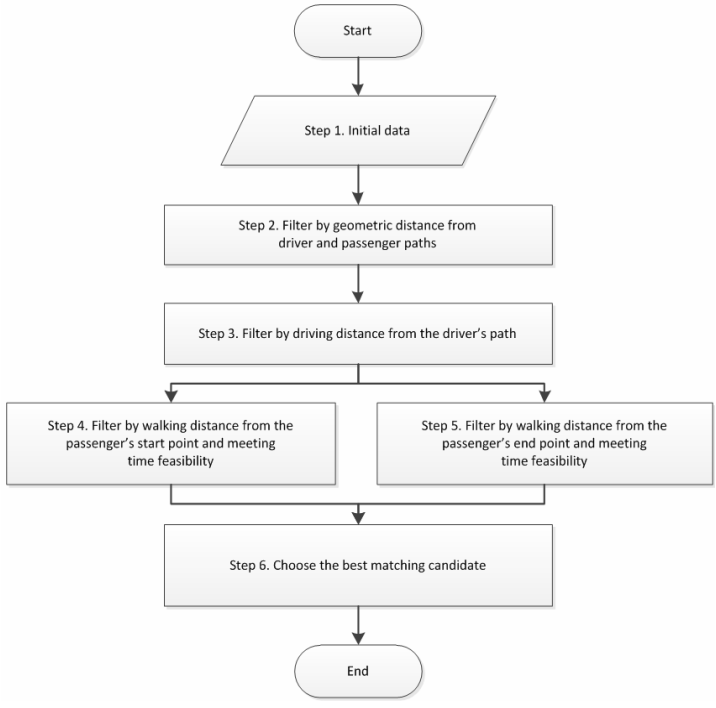



**Fig. 4.** The algorithm for finding matching points

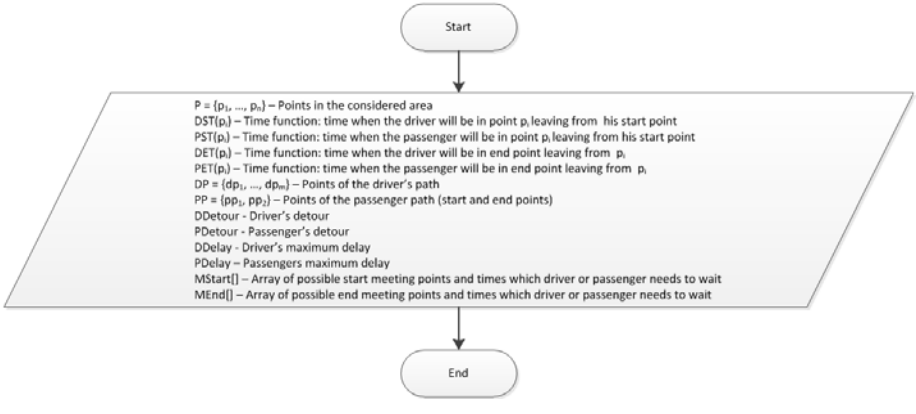



**Fig. 5.** Initial data

At step 3 (Fig. 7) a more precise estimation of the meeting points is performed taking into account possible driver's routes. The computational complexity of this step is $O(q*m)$ (proportional to the product of the number of possible meeting points ($MP$) and the number of the appropriate driver's route points ($DP'$) found at step 2).

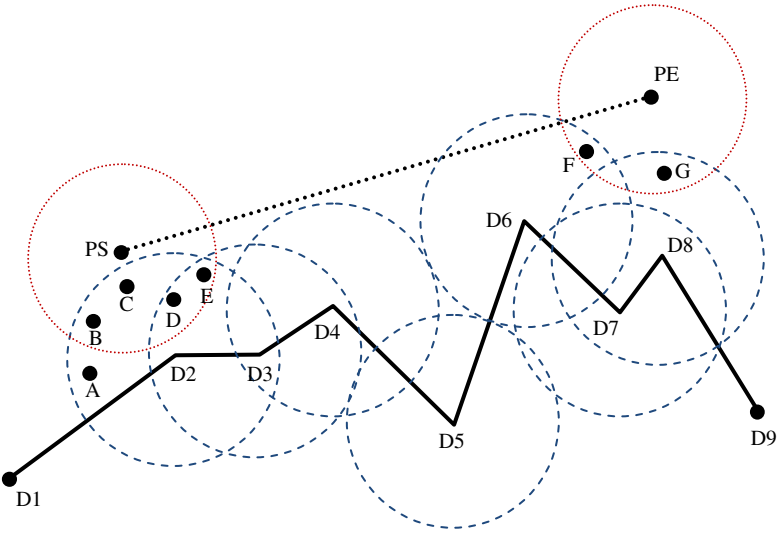



**Fig. 6.** Filtering by geometric distance from driver and passenger paths: scheme

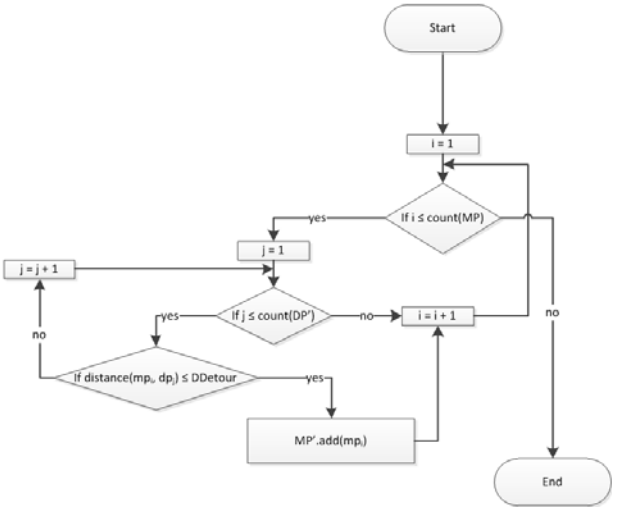



**Fig. 7.** Filtering by driving distance from driver's path

At steps 4-5 (Fig. 8, Fig. 9) a more precise estimation of the meeting points is performed taking into account possible passenger's routes as well as estimation based on the waiting times (defined as delays in the ontology/profile) for the passenger's start point (step 4, Fig. 8) and for the passenger's end point (step 5, Fig. 9). The computational complexity of each of these steps is $O(q')$ (proportional to the number of possible meeting points ($MP'$) found at step 3).

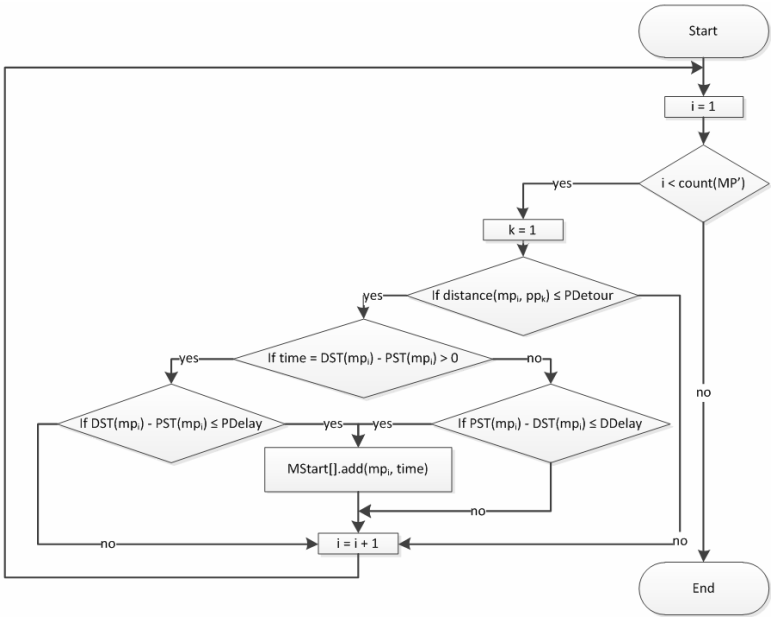



**Fig. 8.** Filtering by walking distance from passenger's start point

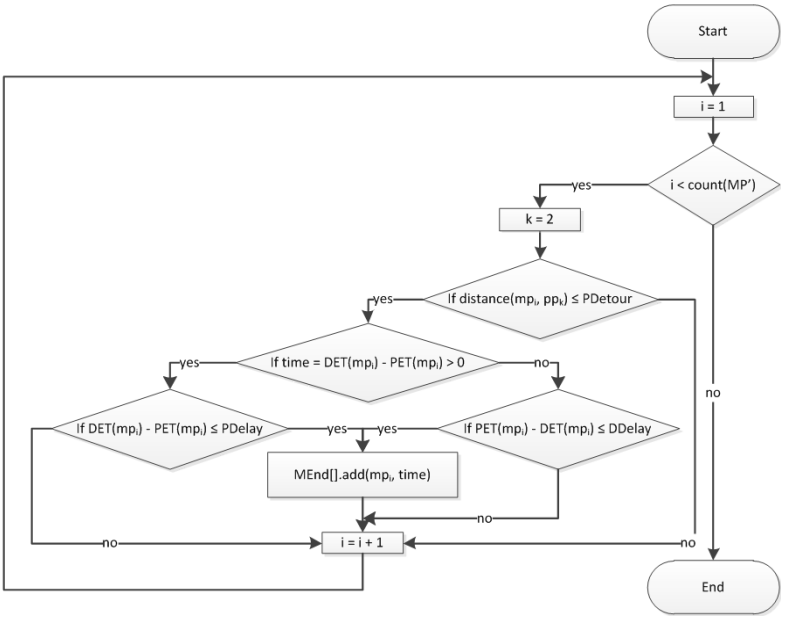



**Fig. 9.** Filtering by walking distance from passenger's end point

At step 6 (Fig. 10) an optimal meeting point is selected among the feasible meeting points defined at the previous steps (arrays *MStart*[] and *MEnd*[]). The criterion of optimality is the minimal total waiting time of both the driver and the passenger. The computational complexity of this step is $O(count(MStart[])*count(Mend[]))$ (proportional to the product of the number of possible meeting points for the passenger pick up and passenger drop off found at steps 4 and 5).

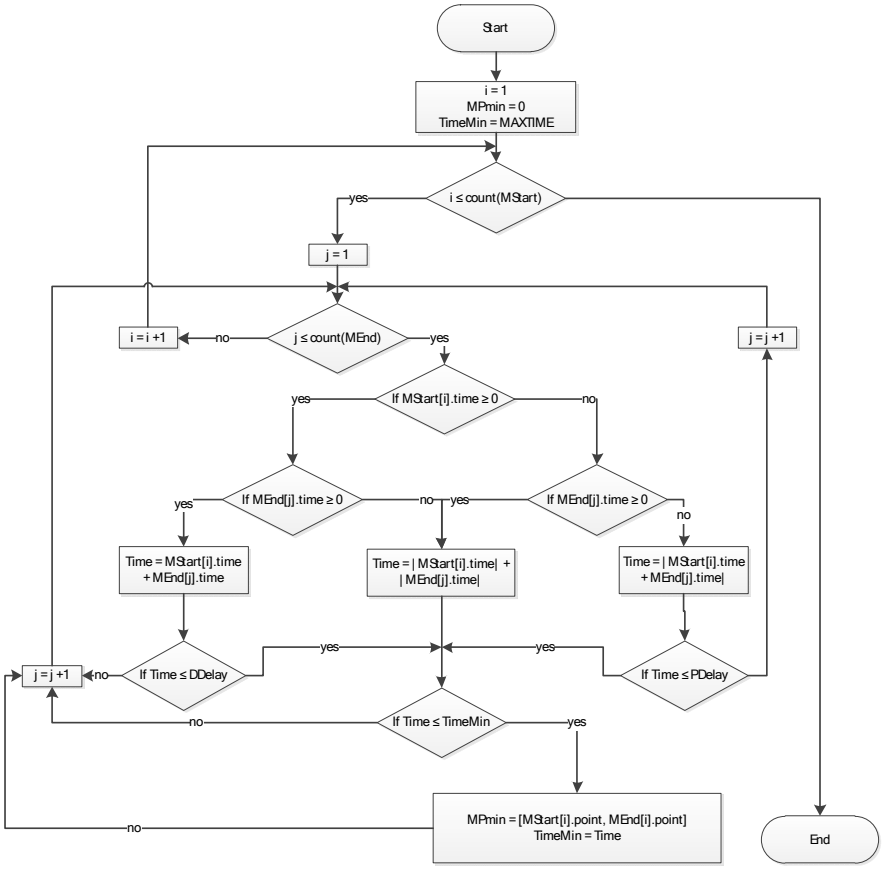



**Fig. 10.** Best point definition

## 5 Broker Performance Estimation

The total complexity of the algorithm can be estimated as the sum of complexities of its each step: $Complexity = = O(m) + O(q*m') + O(q') + O(q') + O(count(MStart[])*count(MEnd[]))$

With assumptions that (1) *q'* is smaller than *q* (*O*(*q'*) can be dropped), and (2) driver's path is much longer than the detour; consequently, *count*(*MStart*[]) and *count*(*MEnd*[]) are much smaller than *q* (*O*(*count*(*MStart*[])**count*(*MEnd*[])) can be dropped), the overall complexity would be as follows: *Complexity* = *O*(*m*) + *O*(*q***m'*), where *m* is the number of the driver's route points and *q* is the number of meeting points found after step 2, and *m'* is the number of driver's route points corresponding to the found meeting points (found at step 2).

The computational time strongly depends on the runtime environment and hardware. However, it can be said that linear complexity dependence on the number of the driver's route points and quadratic complexity dependence on the reduced numbers of points are reasonable.

Since this algorithm have to be run for each passenger and each driver, the total complexity would be: *Number_of_passengers* * *Number_of_drivers* * *Complexity.*

## 6 Experimentation Results

The experiments were performed with the Broker was running on the computer with Pentium M 1.86 GHz processor and 1024 Mb of RAM.

The first set of experiments was performed without filtering for the reduction of the computational complexity. Instead an exhaustive search for matching requests and offers. The approximating equation is quadratic for the total amount of users, which corresponds to the found earlier complexity equation.

The second set of experiments was performed using filtering for the reduction of the computational complexity. The approximating equation is also quadratic for the total amount of users, which corresponds to the found earlier complexity equation.

It can be seen that usage of proposed filtering techniques significantly reduces the calculation time (about 4 times). The results can be seen in Table 1 and Fig. 11.

**Table 1.** Experiment results

| Drivers | Passengers | Total Participants | Matching without using filtering, sec. | Matching using filtering, sec. |
|---|---|---|---|---|
| 10 | 10 | 20 | 14.17 | 4.85 |
| 10 | 20 | 30 | 26.70 | 9.12 |
| 20 | 20 | 40 | 54.83 | 17.51 |
| 30 | 30 | 60 | 139.81 | 37.93 |
| 40 | 40 | 80 | 274.03 | 66.13 |
| 50 | 50 | 100 | 431.51 | 101.29 |
| 100 | 100 | 200 | >1800.00 | 398.71 |

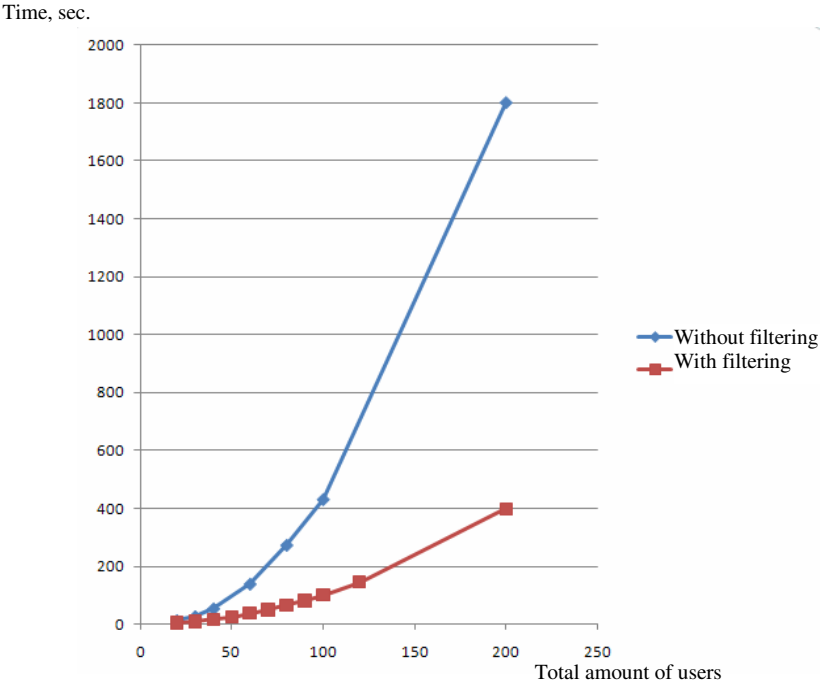



**Fig. 11.** Comparison graph

## 7 Conclusion

The paper extends the proposed earlier approach to sustainable logistics based on the usage of Smart-M3 platform for supporting a ridesharing system. Unlike similar systems (e.g., [13], where drivers set the list of possible pick up and drop off points), in the presented approach the possible pick up and drop off points are calculated automatically and optimized, what causes additional load for the broker. To increase the broker's performance the matching procedure within the broker has been split into several steps, where each step significantly reduces the search step at minimal computational expenses with no information losses. However, the performance can be further improved using heuristics with some information loss (this is one of the aims of the future research). The experimentation showed that broker already takes a reasonable time for result generation. Presented results are based on the usage of a research prototype written in Python and running on a desktop PC. In a production environment the broker is aimed to be run on a dedicated server and it is expected to be responsive enough to handle a large amount of queries daily. The future development of Smart-M3 up to the production level with a higher capacity (e.g., currently it slows down significantly if the number of subscribe queries reaches about 300) could also contribute to the system performance.

**Acknowledgments.** The presented work is a result of the joint project "Smart Logistics" between SPIIRAS and Nokia Research Center. Part of the implementation work has been also supported by Open Innovations Framework Program FRUCT – www.fruct.org. Some parts of the research were carried out under projects funded by grants # 09-07-00436-a and # 11-07-00045-a of the Russian Foundation for Basic Research, and project # 213 of the research program "Intelligent information technologies, mathematical modeling, system analysis and automation" of the Russian Academy of Sciences.

## References

1. Global GHG Abatement Cost Curve v 2.0 (2009), https://solutions.mckinsey.com/ClimateDesk/default.aspx
2. $CO_2$ Emissions from Fuel Combustion: Highlights. IEA Statistics, pp. 121 (2010), http://www.iea.org/co2highlights/co2highlights.pdf
3. UK Industry Task Force for Peak Oil and Energy Security, The oil crunch. Final report (2010), http://peakoiltaskforce.net/wp-content/uploads/2010/02/final-report-uk-itpoes_report_the-oil-crunch_feb20101.pdf
4. US Joint Forces Command, The Joint Operation Environment (JOE) (2010), http://www.jfcom.mil/newslink/storyarchive/2010/JOE_2010_o.pdf
5. Paloheimo, H., Waris, H., Balandin, S., Smirnov, A., Kashevnik, A., Shilov, N.: Ad-Hoc Mobile Networks for Sustainable Smart Logistics. In: Proceedings of the Congress on Intelligent Systems and Information Technologies, AIS-IT 2010, Physmatlit, vol. 4, pp. 38–44 (2010)
6. Hartwig, S., Buchmann, M.: Empty Seats Traveling. In: Next-generation Ridesharing and Its Potential to Migrate Traffic- and Emission Problems in the 21-st Century, February 14. Nokia Research Center Bochum (2007)
7. Zimmermann, H., Stempfel, Y.: Current Trends in Dynamic Ridesharing, Identification of Bottleneck Problems and Propositions of Solutions. Institute of Technology, Delhi(2009)
8. Commuter Financial Incentives. Parking Cash Out, Travel Allowance, Transit and Rideshare Benefits. TDM Encyclopedia Victoria Transport Policy Institute, http://www.vtpi.org/tdm/tdm8.htm
9. Ferreira J., Trigo P., Filipe P.: Collaborative Car Pooling System. World Academy of Science, Engineering and Technology 54 (2009)
10. Honkola, J., Laine, H., Brown, R., Tyrkkö, O.: Smart-M3 information sharing platform. In: The 1st Int'l Workshop on Semantic Interoperability for Smart Spaces (SISS 2010) in Conjunction with IEEE ISCC 2010 (2010)
11. Smart-M3 at Wikipedia, http://en.wikipedia.org/wiki/Smart-M3
12. Smart-M3 at Sourceforge, http://sourceforge.net/projects/smart-m3
13. Avego, http://www.avego.com

# Distributed Deadlock Handling for Resource Allocation in Smart Spaces

Rehan Abdul Aziz[1], Tomi Janhunen[1], and Vesa Luukkala[2]

[1] Aalto University School of Science
Department of Information and Computer Science
rabdulaz@cc.hut.fi,Tomi.Janhunen@aalto.fi
[2] Nokia Research Center
Vesa.Luukkala@nokia.com

**Abstract.** In a ubiquitous system, there are several interacting computational objects which use each others' resources. As the number of resources and their consumers grow in such systems, the delay that the consumers experience for obtaining control over resources increases with an existing rule-based resource allocation technique. Distributing the resource allocation, however, complicates the nature of deadlocks that may arise and requires more sophisticated techniques as compared to a setup with central control. The goal of this paper is to generalize the current resource allocation method to a distributed setting and, in particular, to propose an approach for handling deadlocks in this case.

**Keywords:** semantic web, resource allocation, distributed deadlock, answer set programming.

## 1 Introduction

The amount of artifacts with computing and digital communication facilities embedded in everyday environment will continue its growth. These objects contain information and functionality which can then be made available to benefit users. From the users' point of view, the distinction between these embedded devices and computing devices, such as personal computers or mobile appliances, will blur. Together these devices form a *smart space*—an abstract entity which makes services and information available for users in a seamless way using the most suitable available resources. This is a realization of the *ubiquitous computing* vision [24].

In this work, we assume that the issues of interoperability are resolved using semantic web standards. The Resource Description Framework (RDF) [18] is used to present the atomic elements of information as a set of triples. The structures which are built on top of RDF can be specified by knowledge representation languages such as RDF-Schema (RDFS) [19] or web ontology language (OWL) [17]. In this approach, the building blocks of the information are being standardized, making the resulting information partially self-describing. We assume that objects present in a smart space publish and consume information about their offered or used services or capabilities minimally in RDF format. One of the key objectives of a smart space is that a user

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 87–98, 2011.

always has the best possible resources available. The resources, their number and availability may vary with respect to time, location, or other parameters. Thus the smart space is an inherently distributed system with multiple independent actors that use and share their resources according to their own priorities.

In order to address the interoperability of the physical world and the information world, a middleware platform [16] has been implemented for sharing RDF information in smart spaces [22]. Based on the experiences obtained from implementing several use cases [5,6,21], a mechanism for a simple resource management framework using shared RDF structures [10] has been proposed as a necessary component for this environment. The underlying problem of resource allocation under preferences requires implementing a search to choose from different options and also detection and removal of deadlocks that may arise in the system.

In previous work [12], a methodology for handling the resource allocation task is introduced and an implementation of a resource allocator based on *answer set programming* (ASP) [9,13,15] is presented. A technical report [1] measuring the timings for the implementation acts as motivation for this work. The methodology described in [12] allocates resources using one centralized ASP rule engine. However, in a smart space environment it cannot be expected that only a single computing entity is able to carry out resource allocation since the participants of the smart spaces may change dynamically and each entity may have to manage its own resources. In these cases there may be several independent instances making the allocation decisions, suggesting a decentralized version of the resource allocation framework. Another problem arising from the distributed nature of the resource allocation is the possibility of a distributed deadlock. Deadlocks may arise from decisions made independently by the distributed resource allocators, so our methodology needs to be able to manage deadlocks which may potentially concern all resources in a smart space. Also, as shown by [1] the effort of deadlock detection tends to grow nonlinearly with the number of resources and users, so we hope that distributed resource allocation could alleviate this problem as well.

The goal of this paper is to create a distributed resource allocation framework with the ability to detect deadlocks and propose an approach compatible with the existing ASP-based implementation. We also compare our approach with related work in the field of distributed database systems. The main contribution of this paper is to tailor existing resource allocation and deadlock management frameworks for smart spaces. The rest of this paper is structured as follows. Section 2 describes the particular smart space environment used in this work and introduces the relevant resource allocation concepts. Section 3 formalizes the research problem that this paper aims to solve. Section 4 describes the centralized approach from [12] as a prelude to Section 5 where a solution for the problem is described in a distributed setting. Section 6 discusses the relation of our approach to distributed databases. Finally, we present our conclusions and lay out some future research directions in Section 7.

## 2 The Smart Space Infrastructure

According to the original semantic web vision [2] information is closely tied to the web infrastructure. However, we assume that web may not be the most suitable mechanism for privacy and efficiency reasons. This has been the motivation for implementing

Smart-M3 [16], an interoperability platform which allows devices to share and access local semantic information, while also allowing the more global semantic information to be available. Logically, Smart-M3 has two kinds of elements: a single *semantic information broker* (SIB), which is a blackboard like RDF store, and *nodes* that can connect to the SIB in order to exchange information. A logical SIB may span over several devices, internally handling the required synchronization. Smart-M3 provides a set of primitives for manipulating the RDF content and ontologies can be used to define more complex structures as part of information published at a SIB.

The available primitives for nodes to manipulate RDF triples on the SIB include an atomic *update* (first performing deletes and then inserts) and both single-shot and persistent queries expressed in languages such as SPARQL [23] and WQL [8]. It is guaranteed that for a single node, the operations are done in the same order as they were performed by the node. The definition of a Smart-M3 application is very loose: it is the result of the combined actions of the participating nodes, which may appear and disappear spontaneously. The implementations of the nodes themselves are not limited to any particular system or runtime, as long as the platform has an implementation of the Smart Space Access (SSA) protocol that offers the basic operations described above.

Specifically, we are interested in scenarios involving potentially large numbers of independent computing devices. Such resources can be used by the participants of a specific smart space to realize more abstract functionalities also represented in the smart space. The resource usage and availability may change dynamically and there may be preferences according to which the resources are allocated in a competitive setting. An example use case is the "music follows user" scenario, where a user can start playing music on her mobile device, but when she enters a car, she can use the car's loudspeakers for listening and the steering wheel buttons for controlling the playing. Likewise, if she visits a friend's home, she can use the multimedia system there, given permission. The idea is that the same implementation of the use case applies to many environments.

When an entity joins the smart space, it publishes information about its capabilities. When it needs to use capabilities of others or of its own, it publishes an activity. They are published on the SIB as RDF instances, which are sets of triples with uniquely named URIs in the first field (the subject field). It is possible that an ontology defines a class, whose instance may consist of several triples, which are properties of the class. Below we call the instances of classes `Activity` and `Capability` as *activities* and *capabilities*, respectively. A capability may have a *capacity*, i.e., a limit on the number of simultaneous users. At this time the object also indicates if particular capability or activity should be managed by the resource allocation framework. For the purposes of this paper, we abstract away connectivity issues by assuming that only the entities whose information is present at SIB are part of the smart space.

When a node wishes to use the capabilities of an object, it publishes an activity, using the `uses` property to target the desired capability. An activity has a property, `active`, which has a value "yes" or "no". It is expected that the creator of the activity monitors the `active` property and honors it so that when it has value "no", the activity is in a paused state. A capability which is targeted by the `uses` relation may commit to an activity by publishing a corresponding `commits` relation. Once all `uses` relations have been committed to, the activity is ready to operate and its `active` property

can be changed to "yes". We call this model an *activity-capability model*. The potential transient nature, large number, and ownership of capabilities implies that it is not reasonable to have a single global arbiter which would manage capabilities and activities. We expect that each capability manages itself, i.e., decides independently whether it commits itself to an activity. At this point policies that restrict what kind of entities may use the capability may be in effect.

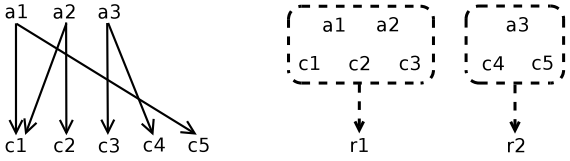
The resource allocation mechanism is also a node in the system. It monitors changes to relevant types of information and queries for more information in the system if needed. Based on this snapshot of the world, it decides the allocation of capabilities to activities and publishes its decisions. A particular node only manages those capabilities and activities which have registered themselves to be managed by that instance of the resource allocator node. Later on we will introduce another entity, which deals with the issues that arise globally across all activities and capabilities in the smart space. The implementation and technical details of combining Smart-M3 with a rule-based ASP implementation of the resource allocator are described in [11,12]. In ASP, it is customary to make *a closed world assumption* about any logical statement, i.e., if it is not explicitly mentioned to be true, its negation is interpreted to be true. In other words, logical statements are false by default. Following the terminology of [12], we call a single node implementing a localized resource allocation a *rule engine* (RE). Additionally in this paper, the entity managing multiple REs is referred to as the *main manager* (MM).

## 3 Resource Allocation and Deadlock Detection Problems

In this section, we provide a formal account of the resource allocation problem and the subsequent deadlock detection problem in a smart space. To this end, we introduce some mathematical notation and define the main concepts involved as follows. Referring to Section 2, we let $C$ and $A$ denote the respective sets of *capabilities* and *activities* in a smart space. Activities may request capabilities which is formalized by the *uses* relation $U \subseteq A \times C$, i.e., each pair $\langle a, c\rangle \in U$ indicates that an activity $a \in A$ requests to use a capability $c \in C$ in the space if permitted to do so. The sets $A$ and $C$ together with $U$ form a directed bipartite graph as illustrated in Figure 1a. The edges of the graph are determined by the relation $U$. Such a structure can be viewed as an instance $\langle C, A, U\rangle$ of the *resource allocation problem* (RAP). Given the requests depicted in Figure 1a, one is supposed to assign capabilities to activities. Thus, we formalize a solution of an RAP instance as a relation $S \subseteq U^{-1}$ where the inverse $U^{-1} = \{\langle c, a\rangle \mid \langle a, c\rangle \in U\}$.[1] Intuitively, a pair $\langle c, a\rangle \in S$ describes that a capability $c$ is *committed* to an activity $a$. Such relationships can be subject to change in a dynamic environment. However, given a fixed solution $S$ to an RAP instance $\langle C, A, U\rangle$, some activities may have to wait for capabilities which, in a circular setting, may give rise to deadlocks.

The *deadlock detection problem* (DDP) corresponding to the solution $S$ is formalized as follows. We say that an activity $a_1$ is *waiting for* an activity $a_2$ in $S$, denoted by $a_1 >_S a_2$, iff there is a capability $c \in C$ such that $\langle a_1, c\rangle \in U$, $\langle a_2, c\rangle \in U$, $\langle c, a_2\rangle \in S$, but $\langle c, a_1\rangle \notin S$. In other words, activities $a_1$ and $a_2$ compete over $c$ that is assigned to

[1] The use of a relation $S \subseteq C \times A$ rather than a partial function $S(c)$ allows for capabilities having a capacity greater than one which could be thereafter shared by several activities.




**(a)** A RAP instance **(b)** A splitting for two REs

**Fig. 1.** Input instance and its splitting

$a_2$ according to $S$. This makes $a_1$ dependent on $a_2$ (hence the notation $a_1 >_S a_2$) and $a_1$ is forced to wait until at least this critical capability is released for its use.

**Definition 1 (Deadlock).** *For an RAP instance* $\langle C, A, U\rangle$ *and its solution* $S$*, a* deadlock *is a sequence* $a_1, \ldots, a_k$ *of activities from* $A$ *such that* $a_1 >_S \ldots >_S a_k$ *and* $a_1 = a_k$.

The dependency relation $>_S$ gives rise to a *wait-for graph* (WG), formally a pair $\langle A, W\rangle$, where for any $a_1, a_2 \in A$, $\langle a_1, a_2\rangle \in W$ iff $a_1 >_S a_2$. Such a graph is similar to *transaction-wait-for graphs* used in deadlock detection algorithms for distributed databases [20]. It is clear by Definition 1 that deadlocks correspond to *cycles* in the wait-for graph $\langle W, G\rangle$ and, hence, DDP reduces to the problem of identifying cycles in a directed graph which can be efficiently dealt with in a non-distributed setting.

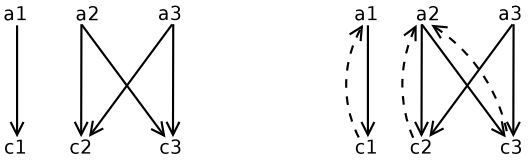
## 4 Centralized Approach

A way to solve the problem described in the last section using a single RE is proposed in [12]. The approach is formalized in ASP as a set of rules and constraints. The input of the problem is the same as described in Section 3 except that there is only a single RE and no MM. After the execution of the RE, the capabilities are allocated to activities in such a way that the rules and constraints described in [12] are satisfied. The output is a set of *commits* pairs. The activities observe these commitments and start to run once they have obtained commitments from all capabilities that they *use*.

*Example 1.* The problem instance shown in Figure 2 shows the function of a single centralized RE. Here $A = \{a_1, a_2, a_3\}$, $C = \{c_1, c_2, c_3\}$, and the set $U$ is given by the edges of the graph in Figure 2a. This graph goes as an input to the RE which then produces a solution as the set $S = \{\langle c_1, a_1\rangle, \langle c_2, a_2\rangle, \langle c_3, a_2\rangle\}$ of *commits* pairs (see Figure 2b). Note that $a_2$ and $a_3$ compete for the same resources but only $a_2$ gets them.

The rules for deriving a solution like the one described above are devised in [12]. They deal with the three tasks described below.

1. *AllocateResources*: These rules are provided by the owners of capabilities and determine how capabilities are committed to activities. They produce a solution which may still contain deadlocks that have to be addressed in the sequel.
2. *FindDeadlock*: To find a deadlock, these rules build a wait-for graph for the activities and detect a cycle in it. The cycle corresponds to a set of deadlocking activities.




**(a)** Input instance **(b)** Solution

**Fig. 2.** Centralized approach using a single RE (dotted arrows show commitments)

3. *ResolveDeadlock*: These rules resolve a deadlock by choosing a random *victim* from it and by removing its *uses* relations temporarily. The resource allocation rules are run again over the modified instance. This task is expressed as Algorithm 1.

The complete job of a single RE is summarized in Algorithm 2. It takes an instance of RAP as its input and finds a solution for it. Then it attempts to remove the deadlocks from the solution by modifying the instance and running resource allocation iteratively until the solution is free of deadlocks. Figure 3 shows an example of how a deadlocked solution is made deadlock free. In the end, $a_3$ waits for $a_2$ that took both $c_2$ and $c_3$.

**Algorithm 1.** $ResolveDeadlock(D, C, A, U)$

$a \leftarrow$ random element in $D$
$U' \leftarrow U \setminus \{\langle x, y\rangle \in U \mid x = a\}$
$S \leftarrow AllocateResources(C, A, U')$
**return** $S$

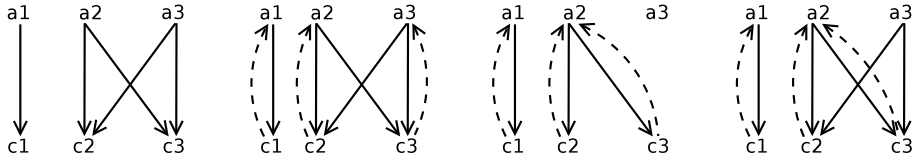
**Algorithm 2.** $CentralRuleEngine(C, A, U)$

$S \leftarrow AllocateResources(C, A, U)$
**while** $(D \leftarrow FindDeadlock(C, A, U, S)) \neq \emptyset$ **do**
  $S \leftarrow ResolveDeadlock(D, C, A, U)$
**end while**

## 5 Distributed Approach

In this section, we describe a distributed approach to solve resource allocation and deadlock detection problems using several REs. Our approach is a *hybrid* one as it combines a completely distributed approach with a partly centralized control. The respective non-hybrid approaches would have inherent shortcomings. In fully distributed setups, the localization of resource allocation is straightforward but the number of messages required for deadlock detection becomes large [20]. On the other hand, the single control site easily becomes a bottleneck in centralized approaches to solving RAPs and DDPs.

To alleviate these shortcomings, we suggest to enhance a distributed setup with a central site, referred to as the main manager in Section 2. The MM is responsible for breaking up an RAP instance into $n$ parts as illustrated in Figure 1b in the case $n = 2$. Then, the idea is to first solve RAPs locally using several REs without any communication. The next objective is to detect the resulting local deadlocks as well as to resolve




**(a)** Input instance **(b)** Solution having a deadlock **(c)** Activity $a_3$ banished and the solution recomputed **(d)** New solution in which $a_3$ restored

**Fig. 3.** Detailed example of centralized resource allocation

them locally. Then the globally relevant dependencies are extracted and conveyed to the MM which finally detects global deadlocks and resolves them. In what follows, we formalize local views for resource allocation in Section 5.1, and then address distributed deadlock detection and handling in Sections 5.2 and 5.3, respectively.

### 5.1 Local Views for Resource Allocation

To enable the distributed solution of an RAP $\langle C, A, U\rangle$, the sets $C$ and $A$ of capabilities and activities involved are split into $n$ disjoint parts—giving rise to the respective disjoint unions $C_1 \sqcup \ldots \sqcup C_n$ and $A_1 \sqcup \ldots \sqcup A_n$. Intuitively, the resulting sets will be assigned to particular REs for solving. The criteria imposed on such splittings is a topic of its own and will be addressed elsewhere in detail. For the purposes of this paper, however, it is sufficient to require that for each $1 \leq i \leq n$ and for each activity $a \in A_i$, there is at least one capability $c \in C_i$ such that $\langle a, c\rangle \in U$, i.e., $a$ requests to use $c$. In addition to $A_i$ and $C_i$, the *local view* over the instance $\langle C, A, U\rangle$ involves the following:

**Definition 2 (Local View).** *Given an RAP instance $\langle C, A, U\rangle$ and splittings $C = C_1 \sqcup \ldots \sqcup C_n$ and $A = A_1 \sqcup \ldots \sqcup A_n$, define for each $1 \leq i \leq n$,*

1. $A_i^{\text{out}}$ *as the set of activities $a \in A \setminus A_i$ having $\langle a, c\rangle \in U$ such that $c \in C_i$;*
2. $A_i^{\text{all}}$ *as the union $A_i \cup A_i^{\text{out}}$;*
3. $A_i^{\text{dst}}$ *as the set of activities $a \in A_i^{\text{all}}$ having $\langle a, c\rangle \in U$ such that $c \notin C_i$;*
4. $A_i^{\text{loc}}$ *as the difference $A_i \setminus A_i^{\text{dst}}$; and*
5. $U_i$ *as the set of pairs $\langle a, c\rangle \in U$ such that $a \in A_i^{\text{all}}$ and $c \in C_i$.*

Intuitively speaking, it is necessary to be aware of activities in $A_i^{\text{out}}$ which request capabilities from $C_i$ but reside *outside* $A_i$. Thus the set $A_i^{\text{all}}$ contains *all* activities having local relevance. Likewise, it is important to distinguish *distributed* and purely *local* activities in the respective sets $A_i^{\text{dst}}$ and $A_i^{\text{loc}}$. Last, $U_i$ localizes the *uses* relation $U$.

*Example 2.* Consider the splitting illustrated in Figure 1b. For the RE $r_1$, we obtain sets $A_1 = \{a_1, a_2\}$, $C_1 = \{c_1, c_2, c_3\}$, $A_1^{\text{out}} = \{a_3\}$, $A_1^{\text{all}} = \{a_1, a_2, a_3\}$, $A_1^{\text{dst}} = \{a_1, a_3\}$, $A_1^{\text{loc}} = \{a_2\}$, and the uses relation $U_1 = \{\langle a_1, c_1\rangle, \langle a_2, c_1\rangle, \langle a_2, c_2\rangle, \langle a_3, c_3\rangle\}$.

**Proposition 1.** *For the splitting $A = A_1 \sqcup \ldots \sqcup A_n$ and any $1 \leq i \leq n$, $A_i^{\text{out}} \subseteq A_i^{\text{dst}}$.*

*Proof.* Let $a \in A_i^{\text{out}}$ which implies that $a \in A \setminus A_i$, i.e., $a \in A_j$ for $j \neq i$. Then there is $\langle a, c\rangle \in U$ for $c \in C_j$ by our basic assumptions about splitting $A$. This implies that there is $\langle a, c\rangle \in U$ satisfying $c \notin C_i$ so that $a \in A_i^{\text{out}}$ and $a \in A_i^{\text{all}}$. Thus $a \in A_i^{\text{dst}}$. □

### 5.2 Distributing Deadlock Detection

Our next objective is to present a distributed method for detecting deadlocks and to establish the correctness of the method. Based on the local views defined above the partitions $C = C_1 \sqcup \ldots \sqcup C_n$ and $A = A_1 \sqcup \ldots \sqcup A_n$ give rise to local RAP instances $\langle C_i, A_i^{\text{all}}, U_i\rangle$ for $1 \leq i \leq n$. These instances can be solved locally and assuming that $S_1, \ldots, S_n$ are the respective solutions, then $S = \bigcup_{i=1}^{n} S_i$ is a global solution to the RAP $\langle C, A, U\rangle$ up to handling deadlocks. To this end, each set $A_i$ of activities with $1 \leq i \leq n$ induces a local WG $\langle A_i^{\text{all}}, W_i\rangle$ where $A_i^{\text{all}} = A_i \cup A_i^{\text{dst}}$ and $W_i$ is the restriction $W \cap (A_i^{\text{all}} \times A_i^{\text{all}})$ of $W$ on $A_i^{\text{all}}$. Such WGs are applicable to local deadlock detection but insufficient in view of detecting global ones. As a remedy, we introduce a global WG $\langle A^{\text{dst}}, W^*\rangle$ in order to capture mutual dependencies of distributed activities in $A^{\text{dst}} = \bigcup_{i=1}^{n} A_i^{\text{dst}}$. The set of edges $W^*$ in the global WG is defined as follows. A path $a_1, \ldots, a_k$ in $\langle A, W\rangle$ is called *local* if it consists of non-distributed activities only, i.e., $\{a_1, \ldots, a_k\} \cap A^{\text{dst}} = \emptyset$. Then, for any *distributed* activities $a_0 \in A^{\text{dst}}$ and $a_{k+1} \in A^{\text{dst}}$, the pair $\langle a_0, a_{k+1}\rangle \in W^*$ iff $\langle a_0, a_1\rangle \in W$, there is a local path $a_1, \ldots, a_k$ in $\langle A, W\rangle$, and $\langle a_k, a_{k+1}\rangle \in W$. If, in particular, $k = 0$, then $\langle a_0, a_1\rangle \in W^*$ iff $\langle a_0, a_1\rangle \in W$. The notion of a local path is justified by the following property.

**Proposition 2.** *If $a_1, \ldots, a_k$ is a local path in $\langle A, W\rangle$ partitioned by $A = A_1 \sqcup \ldots \sqcup A_n$, then $\{a_1, \ldots, a_k\} \subseteq A_i$ holds for some fixed $1 \leq i \leq n$.*

*Proof.* Let $a_1, \ldots, a_k$ be a local path in $\langle A, W\rangle$, i.e., $\{a_1, \ldots, a_k\} \cap A^{\text{dst}} = \emptyset$, and $1 \leq i \leq n$ the index of $A_i$ to which $a_1$ belongs. This is the base case for our inductive argument that $a_j \in A_i$ for each $j \geq 1$. Then consider any $a_j$ with $j > 1$. By the inductive hypothesis $a_{j-i} \in A_i$. Assuming $a_j \notin A_i$ implies that $a_j \in A_l$ for some $1 \leq l \leq n$ such that $l \neq i$. Since $a_{j-1} >_S a_j$ for the solution $S$ inducing $\langle A, W\rangle$, there is a capability $c$ such that $\langle a_{j-1}, c\rangle \in U$ and $\langle a_j, c\rangle \in U$, $\langle c, a_j\rangle \in S$, but $\langle c, a_{j-1}\rangle \notin S$. If $c \in C_i$, then $a_j \in A_i^{\text{out}}$. It follows by Proposition 1 that $a_j \in A_i^{\text{dst}} \subseteq A^{\text{dst}}$, a contradiction. Thus $c \notin C_i$ holds. Then $a_{j-1} \in A^{\text{out}}$ for some $1 \leq m \leq n$ with $m \neq i$. Again, by Proposition 1, $a_{j-1} \in A_m^{\text{dst}} \subseteq A^{\text{dst}}$, a contradiction. Hence $a_j \in A_i$. □

Let us then point out two corner cases of the definitions of local and global WGs.

1. If $n = 1$, then $A_1^{\text{dst}} = \emptyset$ by definition and $A_1^{\text{all}} = A_1$. It follows that the only local graph $\langle A_1^{\text{all}}, W_1\rangle = \langle A, W\rangle$ and the global graph collapses to $\langle \emptyset, \emptyset\rangle$.
2. If $n = |A|$, then each $A_i$ is a singleton. If, in addition, each activity is distributed, then the global graph $\langle A^{\text{dst}}, W^*\rangle$ coincides with $\langle A, W\rangle$.

**Theorem 1.** *Let $\langle C, A, U\rangle$ be an RAP instance and $A = A_1 \sqcup \ldots \sqcup A_n$ a splitting of $A$ inducing local RAPs $\langle C_i, A_i^{\text{all}}, U_i\rangle$ for $1 \leq i \leq n$. Let $S_1, \ldots, S_n$ be local solutions and $S$ their union. Then the WG $\langle A, W\rangle$ corresponding to $S$ has a deadlock iff*

**Algorithm 3.** $DistributedRuleEngine(C_i, A_i^{\mathrm{all}}, U_i)$

1: $S \leftarrow AllocateResources(C_i, A_i^{\mathrm{all}}, U_i)$
2: **while** $(D \leftarrow (FindDeadlock(C_i, A_i^{\mathrm{all}}, U_i, S) \cap A_i^{\mathrm{loc}})) \neq \emptyset$ **do**
3: $\quad ResolveDeadlock(D, C_i, A_i^{\mathrm{all}}, U_i)$
4: **end while**
5: $W_i^* \leftarrow GetDistributedDependencies(C_i, A_i^{\mathrm{all}}, U_i)$
6: $ReportToMainManager(W_i^*)$

1. *for some* $1 \leq i \leq n$*, the respective local WG* $\langle A_i^{\mathrm{all}}, W_i \rangle$ *has a deadlock, or*
2. *the global WG* $\langle A^{\mathrm{dst}}, W^* \rangle$ *has a deadlock.*

*Proof.* ( $\Longrightarrow$ ) Suppose that the WG $\langle A, W \rangle$ has a deadlock, i.e., there is a loop $a_1, \ldots, a_k$. (i) If none of the activities is distributed, then the path $a_1, \ldots, a_k$ is local and $\{a_1, \ldots, a_k\} \subseteq A_i$ for some $1 \leq i \leq n$ by Proposition 2. It follows that $a_1, \ldots, a_k$ is a loop in $\langle A_i^{\mathrm{all}}, W_i \rangle$ since $A_i \subseteq A_i^{\mathrm{all}}$ by definition. (ii) Otherwise, there is at least one distributed activity involved in the loop. Let $d_1, \ldots, d_l$ be the subsequence formed by the distributed activities in $a_1, \ldots, a_k$ such that $1 \leq l \leq k$ and $d_1 = d_l$ where $d_l$ can simply repeat $d_1$. For each pair $\langle d_j, d_{j+1} \rangle$ of activities with $1 \leq j < l$ there is a local path in $\langle A, W \rangle$ that consists of the intermediate nodes between $d_j$ and $d_{j+1}$ in $a_1, \ldots, a_k$. It follows that $\langle d_j, d_{j+1} \rangle$ is an edge in $\langle A^{\mathrm{dst}}, W^* \rangle$. Thus $d_1, \ldots, d_l$ forms a loop in $\langle A^{\mathrm{dst}}, W^* \rangle$ and a deadlock. The cases (i) and (ii) establish the claim.

( $\Longleftarrow$ ) Let us then assume that one of the local graphs $\langle A_i^{\mathrm{all}}, W_i \rangle$ has a deadlock actualized by a loop $a_1, \ldots, a_k$. Since $W_i$ is the restriction of $W$ on $A_i^{\mathrm{all}}$ by definition, this is also a loop in $\langle A, W \rangle$ that has a deadlock then. But what if the global graph $\langle A^{\mathrm{dst}}, W^* \rangle$ has a deadlock realized as a loop $d_1, \ldots, d_l$? Then, for each pair $\langle d_j, d_{j+1} \rangle$ of distributed activities with $1 \leq j < l$, there is a local path in some $\langle A_i^{\mathrm{dst}}, W_i \rangle$ with $1 \leq i \leq n$. Using these, the loop $d_1, \ldots, d_l$ can be expanded to a loop $a_1, \ldots, a_k$ of $\langle A, W \rangle$ such that $a_1 = d_1 = a_k = d_l$. Thus $\langle A, W \rangle$ has a deadlock as desired. □

### 5.3 Distributing Deadlock Handling

We are now fully prepared to present an algorithm, viz. Algorithm 3, for distributed resource allocation and deadlock handling in smart spaces. Once local resource allocation has been carried out (line 1), local deadlocks are detected one by one (line 2) and resolved by victimizing a random activity $a$ in a deadlock $D$ and by temporarily removing all requests $\langle a, c \rangle$ from $U$ (line 3). Finally, the dependencies among distributed activities are determined as a relation $W_i^* \subseteq (A_i^{\mathrm{dst}} \times A_i^{\mathrm{dst}})$. The set $W^*$ in the global WG can be formed as the union $W_1^* \cup \ldots \cup W_n^*$. If there is a cycle $a_1, \ldots, a_k$ in $\langle A^{\mathrm{dst}}, W^* \rangle$, a random victim $a_i$ can be chosen from the cycle and treated as in the case of a local deadlock. Steps 1–4 in Algorithm 3 are quite similar to Algorithm 2 except for the arguments. This highlights the fact that a distributed implementation is easy to obtain from the non-distributed one by changing the domains of the relevant logical rules. In case of ASP, such a modification is straightforward to carry out. For the rules used in resource allocation, the domain of activities is reduced to $A_i^{\mathrm{all}}$, the domain of capabilities to $C_i$, and the domain of edges to $U_i$. For deadlock detection, we use the same domains. However, for deadlock resolution, the domain of activities is $A_i^{\mathrm{loc}}$.

## 6 Relation to Distributed Database Systems

The purpose of this section is to highlight the connection between the two distinct fields of database systems and smart spaces from the point of view of resource allocation. The central difference between the architectures of distributed databases and smart spaces is the way in which information is exchanged between different nodes of the system. In a smart space, nodes communicate by publishing all the information to the SIB, while in distributed databases, sites exchange messages through peer-to-peer communication.

There are several models for distributed databases which describe the components of the system and the way they communicate. A brief survey of these models is given in [7]. The activity-capability model for smart spaces has many features in common with the *resource* model for databases. REs and capabilities correspond to *controllers* and *resources* respectively. Activities are quite similar to *transactions* in the sense that both consume resources. However, activities are created to perform a single task while a transaction is a sequence of operations, i.e., activities only make requests for capabilities once while a transaction may request for resources multiple times. Moreover, a transaction requests resources at different sites by using communication between *processes* [3] that represent the transaction at different *sites*. In contrast, there are no inter RE connections in smart spaces. The decision about the commitments from capabilities are directly published to the SIB. The responsibility of enabling and disabling a particular activity, however, rests with a single RE assigned to it. It is also important here to make the distinction between the *single-resource* model and the *AND* model as highlighted in [7]. Since an activity may make simultaneous requests to many capabilities, our setup is closer to the AND model.

The proposed distributed strategy for deadlock handling can be considered a special case of the hierarchical approach described in [14]. As described in Section 5, the task of resource allocation is restricted to REs while the deadlock detection and resolution is done in collaboration of REs and the MM. REs are analogous to *leaf controllers*, while the MM can be viewed as the only *non-leaf controller*. However, as opposed to [14], there is no distinction between the *input* and *output* nodes. The deadlock handling in our approach only distinguishes between *distributed* and *local* activities. A similar approach for deadlock detection as ours is given in [4] but the definitions for locality arise due to *resources* which are analogous to capabilities, as opposed to processes, which are analogous to activities. Moreover, the information reported to the MM is different from what is reported to the *Central Controller (CC)* in [4]. In our approach, only the dependencies that remain between distributed activities after local deadlock handling are reported to the MM. In contrast, in [4], the whole weakly connected component in the resource allocation graph (*connected set*) associated with a process that is denied a globally shared resource is reported to the CC.

## 7 Discussion and Future Work

In this paper, we present a distributed approach to resource allocation and deadlock handling that works by splitting a problem instance into several parts—each of which is assigned to a different RE that performs resource allocation and part of deadlock

handling locally. Information related to distributed activities is delivered to the MM, which can subsequently detect any remaining deadlocks. The details about the splitting strategy are abstracted away in this paper for two reasons. First, it is a different research problem to analyze different strategies for splitting the problem instance in such a way that the number of distributed activities is minimized which will result in minimum MM intervention for global deadlock handling. This can be done, for instance, by finding the strongly connected components of an RAP instance and then assigning each one of them to a different RE. Ideal parallelization can be achieved if all components are equally sized and distributed activities are completely absent. In such a case, no global deadlocks can arise. However, for large problem instances, it may be necessary to break the largest components into smaller ones by making some activities distributed. Second, the distributed approach to deadlock detection can be applied to the case where the objective is integration of numerous smart spaces already existing, each one with its own RE. In such a setting, splitting is not needed at all since the problem instance is divided by nature.

An advantage of decoupling resource allocation and deadlock handling is that the former can be done in real-time and the latter periodically. The frequency of initiating local as well as global deadlock detection can be varied according to the probability of deadlocks. Deadlock probability for database systems is briefly discussed in the concluding section of [20]. However, it would be an oversimplification to assume that the factors on which the deadlock probability depends in database systems such as resource request and release patterns, average number of resources needed by a transaction, and duration of one transaction are similar in smart spaces. Furthermore, smart spaces can be different from each other in terms of size, lifespan, and activity-capability density. Even activities within a single smart space can be very diverse in their time duration. Therefore, the probability of deadlocks is more unpredictable in the case of smart spaces. This also points to the incompatibility of the time-out deadlock resolution mechanism used in databases due to the broad spectrum of activity lifetimes.

There can be several directions for the future work. Implementation of the proposed distributed approach and its comparison with the centralized approach in terms of performance is our first target. Other directions are more research oriented. The proposed deadlock resolution suggested in this paper can fail for dense wait-for graphs since the resolution only terminates one victim *temporarily* as soon as a cycle is detected. More cycles may appear while others are being resolved. Therefore, we feel that it would be useful to apply the body of research work that exists for distributed databases in this domain. To this end, further unification of database techniques that deal with resource allocation and deadlock handling with our research is required.

*Acknowledgments.* This research has been funded by the European Institute of Technology (EIT) ICT Labs as part of its *Smart Spaces* thematic action line and the European Commission under the SOFIA (Smart Objects for Intelligent Applications) project.

## References

1. Aziz, R.A.: Testing scalability of SSLS rule engine, http://sourceforge.net/projects/ssls/files/raaziz11.pdf

2. Berners-Lee, T., Hendler, J., Lassila, O.: The semantic web. Scientific American (May 2001)
3. Chandy, K., Misra, J., Haas, L.: Distributed deadlock detection. ACM Trans. Comput. Syst. 1(2), 144–156 (1983)
4. Elmagarmid, A., Sheth, A., Liu, M.: A partially distributed deadlock detection algorithm. International Journal of Parallel Programming 14, 307–330 (1985)
5. Främling, K., Oliver, I., Honkola, J., Nyman, J.: Smart spaces for ubiquitously smart buildings. In: Proceedings of the 3rd International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM 2009) (October 2009)
6. Honkola, J., Laine, H., Brown, R., Oliver, I.: Cross-domain interoperability: A case study. In: Balandin, S., Moltchanov, D., Koucheryavy, Y. (eds.) ruSMART 2009. LNCS, vol. 5764, pp. 22–31. Springer, Heidelberg (2009)
7. Krivokapic, N., Kemper, A., Gudes, E.: Deadlock detection in distributed database systems: A new algorithm and a comparative performance analysis. VLDB J. 8(2), 79–100 (1999)
8. Lassila, O.: Programming Semantic Web Applications: A Synthesis of Knowledge Representation and Semi-Structured Data. PhD thesis, Helsinki University of Technology (November 2007)
9. Lifschitz, V.: Answer set planning. In: Proceedings of the 16th International Conference on Logic Programming, pp. 25–37. MIT Press, Cambridge (1999)
10. Luukkala, V., Binnema, D.-J., Borzsei, M., Corongiu, A., Hyttinen, P.: Experiences in implementing a cross-domain use case by combining semantic and service level platforms. In: Proceedings of the The IEEE symposium on Computers and Communications, ISCC 2010, pp. 1071–1076 (2010)
11. Luukkala, V., Honkola, J.: Integration of an answer set engine to SMART-M3. In: Proceedings of the Third Conference on Smart Spaces and Next Generation Wired, and 10th International Cconference on Wireless Networking, SMART/NEW2AN 2010, pp. 92–101. Springer, Heidelberg (2010)
12. Luukkala, V., Niemelä, I.: Enhancing a smart space with answer set programming. In: Dean, M., Hall, J., Rotolo, A., Tabet, S. (eds.) RuleML 2010. LNCS, vol. 6403, pp. 89–103. Springer, Heidelberg (2010)
13. Marek, W., Truszczyński, M.: Stable models and an alternative logic programming paradigm. In: The Logic Programming Paradigm: a 25-Year Perspective, pp. 375–398. Springer, Heidelberg (1999)
14. Menascé, D., Muntz, R.: Locking and deadlock detection in distributed data bases. IEEE Trans. Software Eng. 5(3), 195–202 (1979)
15. Niemelä, I.: Logic programs with stable model semantics as a constraint programming paradigm. Ann. Math. Artif. Intell. 25(3-4), 241–273 (1999)
16. OpenM3 release, http://sourceforge.net/projects/smart-m3/
17. Web ontology language, http://www.w3.org/2004/OWL/
18. Resource description framework, http://www.w3.org/RDF/
19. RDF vocabulary description language, http://www.w3.org/TR/rdf-schema
20. Singhal, M.: Deadlock detection in distributed systems. Computer 22(11), 37–48 (1989)
21. Smirnov, A., Kashevnik, A., Shilov, N., Oliver, I., Balandin, S., Boldyrev, S.: Anonymous agent coordination in smart spaces: State-of-the-art. In: Balandin, S., Moltchanov, D., Koucheryavy, Y. (eds.) ruSMART 2009. LNCS, vol. 5764, pp. 42–51. Springer, Heidelberg (2009)
22. SOFIA project, http://www.sofia-project.eu
23. W3C recommendation: SPARQL query language for RDF
24. Weiser, M.: The computer for the twenty-first century. Scientific American 265(3), 94–104 (1991)

# Inter-Agent Cooperative Communication Method Using TupleSpace for Guiding Users to Alternative Actions

Nobuo Sato and Kazumasa Takami

Graduate School of Engineering,
Soka University,
1-236 Tangi-cho, Hachioji-shi, Tokyo, 192-8577, Japan
k_takami@t.soka.ac.jp

**Abstract.** We propose an agent construction method that can identify action models from user action histories, propose alternative actions if users encounter situations different from their normal action models, and guide the users to those alternative actions. In addition, to realize an alternative action guidance service, we present a selection of terminal agents and network agents as well as an allocation of their functions and roles. Furthermore, we propose a steady-action model definition for realizing the service, a tuple classification and element constructions for TupleSpace-based inter-agent cooperative communication, and an alternative-action extraction algorithm. Next, we present a prototype system-based operation check of inter-agent cooperative communication and an evaluation of tuple-matching processing time.

**Keywords:** Action model, TupleSpace, Agent, Alternative-action, LifeLogs, Inter-agent cooperative communication, Ruby, Rinda.

## 1 Introduction

Recent cellular telephones have seen not only improvements in transmission speeds but also an increasing number of models that come standardly equipped with features such as WI-FI, GPS, and IC cards. Furthermore, end-user programming has also become possible on smartphones. Via these mobile terminals, user information such as position, purchase history, and transportation boarding history can be stored in servers, and users have become able to enjoy recommendation services and the like from that information [1]-[3]. Furthermore, there are expectations for not only a widening of the ubiquitous environment in which a diversity of IT devices such as sensors and information appliances connect to the network [4]-[6], but also a provision of advanced services using "LifeLogs" containing the detailed actions of individuals [7]-[14]. These services will be provided as agents loaded in mobile terminals cooperate with the network according to the situations of the users, but compliance with individual privacy protection is necessary.

On the other hand, users tend to become insensible to situations such as ordinary, constant practical problems like sudden delays when a train or bus accident occurs when they are making their regular commutes to and from work or school. Conventional services do not respond sufficiently to these sorts of situations. In

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 99–110, 2011.

particular, with regard to public transportation used daily in commuting to and from work or school in urban areas, a life based on the empirical assumption of such transportation running on time is natural, and transportation users do not frequently check the operation situations. In addition, according to press release materials of the Ministry of Land, Infrastructure, Transport and Tourism that were issued in December 2009, train service stoppages and train delays of more than 30 minutes have been increasing yearly, and, in fiscal year 2008, the number of such stoppages and delays occurring in Tokyo and three neighboring prefectures (Kanagawa, Saitama, and Chiba) was 40,600 [15]. It is inferred that there are many people whose action plans are affected by those train service stoppages and train delays. Therefore, in cases where public transportation operation is disrupted because of unexpected accidents or disasters, quick thinking to minimize the effects and support to determine alternative actions are important.

In this paper, we investigate an agent construction method that can ensure individual privacy protection, identify action models from user action histories, propose alternative actions when users encounter situations different from their normal action models, and guide users to the alternative actions. In Chapter 2, we present a brief overview of an alternative action guidance service and the agent construction method for that service, namely, the selection of agents and the allocation of their functions and roles. In Chapter 3, we propose a steady-action model definition for realizing that service, a tuple classification and element constructions for TupleSpace-based inter-agent cooperative communication, and an alternative-action extraction algorithm. In Chapter 4, we present a prototype system-based operation check of inter-agent cooperative communication and an evaluation of tuple-matching processing time. Finally, in Chapter 5, we present the conclusion and future challenges.

## 2 Alternative-Action Support Service and Agent Construction

We propose a group of agents that can respond to the situation and guide users if users that use public transportation regularly encounter a situation that occurs suddenly. The group of agents consists of a terminal agent (hereinafter referred to as "$A_T$") and a group of agents existing on the network (hereinafter referred to as "$A_N$"), and they cooperate over the network. $A_T$ holds a user's action model, but it is necessary to recognize where an exceptional event occurs in that action range. $A_T$ must obtain accident information from $A_N$, which holds accident information within the network, and the cooperative communication method is important. In inter-agent communication, an unspecified large number of agents must be able to communicate asynchronously if an unexpected event occurs. As a method suited to this kind of communication configuration, we used the inter-agent cooperative communication method over a TupleSpace [16]-[19].

Figure 1 shows the functional model of a service that supports alternative actions if users encounter train delay problems.

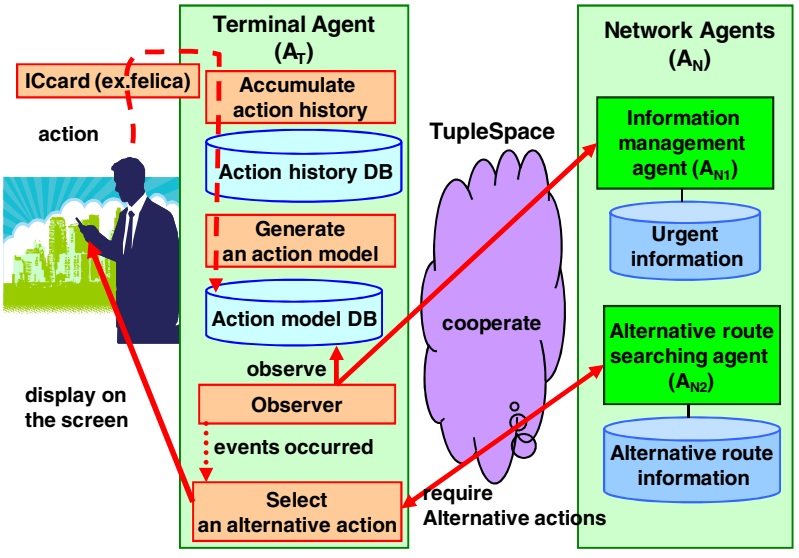



**Fig. 1.** Functional Model of an Alternative Action Support Service

$A_T$ holds the action model as a database, and it figures out the next action transition from that action model and the current time. In addition, $A_T$ is linked to the TupleSpace, and, to check accident information from the action model successively, it writes the next action into the TupleSpace as a question. If an answer is received, it implements processing such as an action model change.

$A_N$ is composed of two agents. One is $A_{N1}$, an accident information management agent for notifying the terminal agent of accident information. The other is $A_{N2}$, an alternative action derivation agent for deriving alternative actions. $A_{N1}$ holds the accident information database, is linked with the TupleSpace, and waits until a question is written into the TupleSpace by $A_T$. If there is a question, it cross-checks the contents of the question and the contents of the accident information database. If it judges that there is accident information, it writes the accident information into the TupleSpace. $A_{N2}$ receives a request from $A_T$ that is inserted into the TupleSpace, derives alternative actions from the information in the request, and inserts the results into the TupleSpace. $A_T$ proposes to the user the optimum alternative action plan from among the plans that it receives.

The functions that are necessary to realize the service shown in Fig.1 are action history collection analysis, action model management, current location detection, alternative action determination, alternative route search, train-delay certificate issuance, alternative-line transfer ticket issuance, and station floor-plan map management. In addition, some of the functions are related to user privacy, and others are held by the train operating companies. The functions are allocated by type to $A_T$ and $A_N$ respectively as shown in Table 1.

**Table 1.** Allocation of Necessary Functions

| Function | $A_T$ | $A_N$ | Reason |
|---|---|---|---|
| Action history collection analysis | ○ | | Involves the handling of private information |
| Action model management | ○ | | |
| Current location detection | ○ | | |
| Alternative action determination | ○ | | To obey the will of the user |
| Alternative route search | | ○ | Use of existing providers is efficient |
| Train-delay certificate issuance | | ○ | Information of train operating companies |
| Alternative-line transfer ticket issuance | | ○ | |
| Station floor-plan map management | | ○ | |

## 3 Steady-Action Model and Inter-agent Cooperative Communication Method

### 3.1 WTLM Action Model

To respond to an unexpected event that occurs in an individual's regular action, for example, commuting to and from work or school, it is necessary to define and obtain a steady-action model. By focusing attention on the bus stops, the train stations, and the times of day that he passed those bus stops and train stations, a college student collected an approximately three-month action history of his commuting pattern from home to college by bus and train. Figure 2 shows part of the action history that was obtained. From Fig. 2, it is was found that, although the time that the student left home differed each day of the week, his commuting pattern focusing on the route and the required time was about the same.

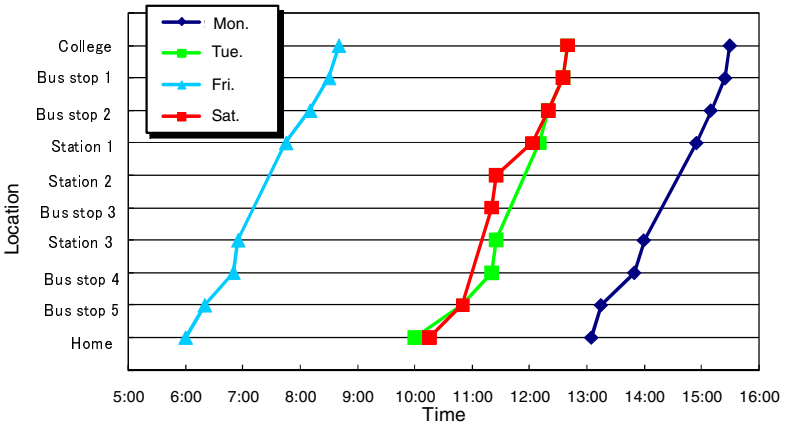



**Fig. 2.** Example of Change in Action and Time Period by Day of the Second Week in September

Therefore, since the action time period changes by the day of the week, we define as follows a WTLM action model that consists of the attributes of the steady-action model, i.e., day of the week, location, time, and means of transportation. Table 2 shows an example of the WTML action model.

$$WTLM\_AM(w, t_1, t_2, l_1, l_2, m)$$

Here, $w$ represents day of the week, $t_1$ and $t_2$ represent departure time and arrival time respectively, $l_1$ and $l_2$ represent departure location and arrival location respectively, and $m$ represents the means of transportation (bus, train, walking, other) from $l_1$ to $l_2$.

**Table 2.** Example of WTLM Action Model

| w | $t_1$ | $t_2$ | $l_1$ | $l_2$ | m |
|---|---|---|---|---|---|
| Monday | 7:00 | 7:30 | Home | Station1 | Walk |
| Monday | 7:30 | 8:00 | Station1 | Station2 | Train |
| Monday | 8:00 | 9:00 | Station2 | Station3 | Train |
| Monday | 9:00 | 9:15 | Station3 | BusStop1 | Walk |
| Monday | 9:15 | 10:00 | BusStop1 | BusStop2 | Bus |
| Monday | 10:00 | 10:15 | BusStop2 | University | Walk |

## 3.2 Inter-agent Cooperative Communication Method

### 3.2.1 Tuple Classes and Elements

TupleSpace-based communication can provide a flexible communication environment for agents, and it is implemented by defining tuples that can be shared freely between sending and receiving agents. In the tuple communication method, agents communicate with each other in the TupleSpace with one agent inserting a tuple that it generates and the other agent taking out that tuple. In addition, the agents can specify the respective survival times of a tuple's acquisition request and the tuple itself, thereby maintaining a space-saving space.

Figure 3 shows an example of TupleSpace-based inter-agent communication [16][17]. If Agent_A writes the information tuple [ "A", "B", "C", 123] into the TupleSpace by a write operation and Agent_B writes the tuple [ "A", "B", nil, nil ], which includes the elements of the desired information, by a take operation, matching processing is executed in the Tuple Space, and Agent_B can obtain tuple [ "A", "B", "C", 123], which includes the desired information, by a read operation.

In order to realize the service proposed in Fig. 1, we defined the three tuples shown in Fig. 4 as the tuples to be used in the inter-agent communication. In each tuple, we constructed as common elements an element for specifying network agent classification ($N_C$), the departure location ($l_1$) and the arrival location ($l_2$) included in the WTML action model, and the means of transportation from $l_1$ to $l_2$ (m). In order for TupleA: request_tuple to send a query from $A_T$ to the network agent, we added departure time ($t_1$) and arrival time ($t_2$) for a total of six elements. In TupleB: accident_info_tuple, in order for $A_{N1}$ to write the accident information obtained – based on TupleA's information – from the accident information database, we constructed a total of seven elements including accident location ($l_A$), accident time ($t_A$), and additional accident

information URL (f). In TupleC: alternative_route_tuple, by adding an element for $A_{N2}$ to write the alternative route set (R) calculated based on TupleA's information, we defined the tuple as five elements.

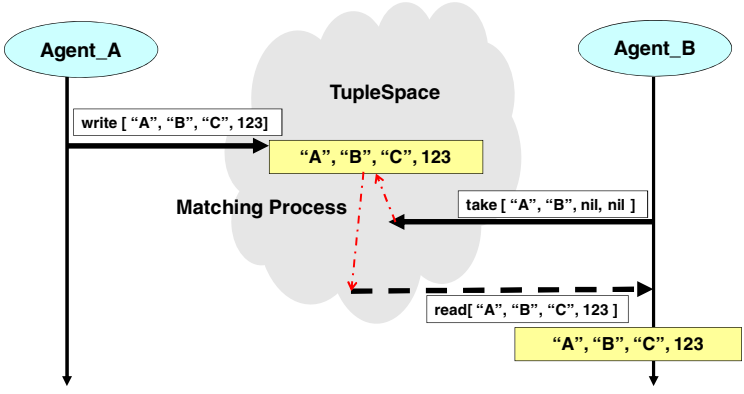



**Fig. 3.** TupleSpace-based Inter-agent Communication Method

TupleA : request_tuple[$N_C$, $l_1$, $l_2$, m, $t_1$, $t_2$]

TupleB : accident_info_tuple[$N_C$, $l_1$, $l_2$, m, $l_A$, $t_A$, f]

TupleC : alternative_route_tuple[$N_C$, $l_1$, $l_2$, m, R]

(note) $N_C$: $A_N$ classification, $l_1$ : Departure location, $l_2$ : Arrival location, m : Means of transportation from $l_1$ to $l_2$, $t_1$ : Departure time, $t_2$ : Arrival time, $l_A$ : Accident location, $t_A$ : Accident time, f : Supplementary information, R: Alternative route set

**Fig. 4.** Tuple Classification and Element Definition

From the viewpoint of privacy information protection, we allocated the functions as shown in Table 1, and it is also necessary to take into consideration the tuple elements. For the purpose of privacy information protection, information that identifies an individual (such as name, address, and user ID) must not be included in those elements. The device with which the terminal agent performs communication directly is the TupleSpace server. However, since information that identifies an individual is not included in the elements of each tuple shown in Fig. 4, if all of the databases that are distributedly managed by purpose are not analyzed, privacy will not be violated by means of only these elements.

### 3.2.2 Alternative Action Derivation Algorithm

Now, we explain the process to derive alternative actions through cooperative operations of the terminal agent and the network agents if an unexpected event occurs in the action model defined in section 3.1. Since WTLM_AM can be seen as a relational model, we also call that single line a record and express it as *C*. As an alternative action derivation method, we present below an algorithm to derive alternative actions taking into consideration the time to arrive from the current location to the destination. This time to arrive is from WTLM_AM record *C*, which

judged that, due to a train accident or other disruption, the commute would not proceed as scheduled.

Step 1: When the algorithm starts, the day of the week, the user's current location, and the time are expressed as $w$, $p$, and $t_x$ respectively.

Step 2: $A_T$ searches to see whether there is a record $C\ \{t_1>t_x)\&(|p-l_1|<d)\}$ that includes the day of the week $w$ from WTLM_AM and that can determine that $t_x<t_1$ and the distance between $p$ and $l_1$ is closer than $d$. If such a record exists, the algorithm proceeds to Step 3. If there is no such record, the algorithm returns to Step 1.

Step 3: $A_T$ extracts all the records for which $t_x<t_l$, and, to search to see whether an accident has occurred in the section indicated by $l_1$ and $l_2$ of each record, it edits the request_tuple corresponding to each record to specify $A_{N1}$ and sends the tuple to the TupleSpace. $A_T$ does this for all the records that it extracted.

Step 4: $A_{N1}$ reads the request_tuples from the TupleSpace and searches the accident information database to see whether there is an accident corresponding to each tuple. If there is, it edits the accident_info_tuple and sends it to the TupleSpace. If there is not, it does nothing. $A_{N1}$ does this for all of the request_tuples.

Step 5: $A_T$ determines from the existence or inexistence of an accident_info_tuple whether or not an accident has occurred. If there is an accident, it sets the accident section to $l_1$ and $l_2$ in the request_tuple. In addition, $A_T$ extracts m, $t_1$, and $t_2$ from the relevant record of WTML_AM, edits a request_tuple specifying $A_{N2}$, and sends it to the TupleSpace. If there is no accident, the algorithm returns to Step 1.

Step 6: $A_{N2}$ reads the request_tuples from the TupleSpace, edits an alternative_route_tuple specifying multiple alternative routes corresponding to each tuple, and writes the tuples into the TupleSpace.

Step 7: $A_T$ obtains alternative route information from the alternative_route_tuples, reconstructs WTML_AM, and presents the information to the user.

## 4 Prototype System and Evaluation

### 4.1 Prototype System

We equipped the evaluation system with Tuple communication functions in a distributed environment and made a prototype by using Ruby [17], which is suitable for prototype preparation. On a single PC, we implemented a distributed Ruby platform and MySQL on Cygwin, and, as shown in Fig. 5, produced each agent and the TupleSpace by programming in Ruby. Furthermore, Tuples A, B, and C indicate the classifications of the tuples in Fig. 4.

In the action model database, we collected the action history of a single examinee's commute from home to college, which he made by walking, riding the bus, and riding the train, and we registered 84 records of relational data conforming to the WTLM action model definition. We defined the following five attributes as the attributes of the accident information database held by one of the network agents: line name, starting and terminal stations of the accident section, accident occurrence time, and

supplementary information. In addition, for the line name and for the starting and terminal stations of the accident section, we set values by the station numbering system. Station numbering is a system for assigning station numbers to rail stations. Separate from the ordinary station names, these station numbers consist of the English letters and Arabic numerals of line symbols. We also applied these Arabic numerals to the elements ($l_1$, $l_2$) that indicate the locations of Tuples A, B, and C. In this way, the overlap of the accident section and the section defined in the action model can be determined easily by assessing only the relative sizes of the numerical values. The accident information database configured in the experiment was registered and run with three accidents: one delay-causing accident that includes a line related to the action model of the examinee and two accidents that do not include such a line. In addition, we constructed the alternative-action route database from five attributes: starting point, destination, means of transportation, alternative route, and arrival time. Then, after registering five alternative routes for the accident section envisioned on the examinee's action range, we conducted the experiment. In a normal situation, a method for successively obtaining alternative routes in cooperation with the Web service of an existing service provider is desirable, but we simplified that method in this experiment. Furthermore, we did not realize processing to determine whether the distance from the user's current location, p, to $l_1$ is within $d$, which is the distance condition of Step 2 mentioned in section 3.2.2.

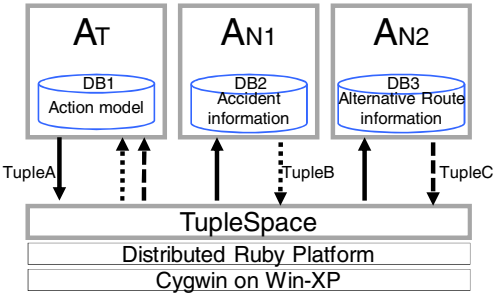



**Fig. 5.** Prototype System Configuration

The alternative action derivation sequence for the inter-agent cooperative communication that we confirmed in this experiment is shown in Fig. 6. We divide the sequence processing flow into "accident information acquisition process" and "alternative-action derivation process" and explain the processing flow as follows.

**(1) Accident information acquisition process**

$A_T$ extracts from the action model (DB1), in order of precedence, action model records for generating request_tuples. Then, for each of those records that it extracted, $A_T$ generates a request_tuple to check for accident information. Next, by a write operation, $A_T$ inserts into the TupleSpace the request_tuple that it generated (Fig. 6 ①).

$A_{N1}$ of the agent group that exists on the network sends to the TupleSpace a request to take out the tuple carrying "an1" and the six-element information (Fig. 6 ②). Matching processing is executed within the TupleSpace, and the six-element tuple

carrying "an1" information is sent to $A_{N1}$. From the information that it took out of the TupleSpace, $A_{N1}$ extracts information on time, line, and section, and it compares that information with the accident information database (DB2). If it judges that the section is a section on which an accident has occurred, it inserts the accident information into the TupleSpace (Fig. 6 ③).

In order to obtain the accident information, $A_T$ sends out to the TupleSpace a request with a time limit, and it obtains from the TupleSpace the accident information (Fig. 6 ④).

**(2) Alternative-action acquisition process**

After processing the acquisition of accident information, $A_T$ generates, from the action model record indicating that an accident has occurred, a request_tuple that includes "an2," and it inserts the tuple into the TupleSpace by a write operation.

Next, $A_{N2}$ requests to the TupleSpace the tuple holding "an2" and the six-element information (Fig. 6 ⑥). Matching processing is executed in the TupleSpace, and the matched tuple is sent to $A_{N2}$. $A_{N2}$, which has received the tuple, obtains from that request_tuple information on the action start time, the action end time, the action start point, and the destination. In addition, based on those pieces of information, $A_{N2}$ searches the alternative actions residing in the alternative action database (DB3). The retrieved alternative action is inserted into the TupleSpace as an alternative_route_tuple (Fig. 6 ⑦).

$A_T$ sends to the TupleSpace a request to obtain the alternative_route_tuple, and it thereby obtains alternative routes from the TupleSpace (Fig. 6 ⑧).

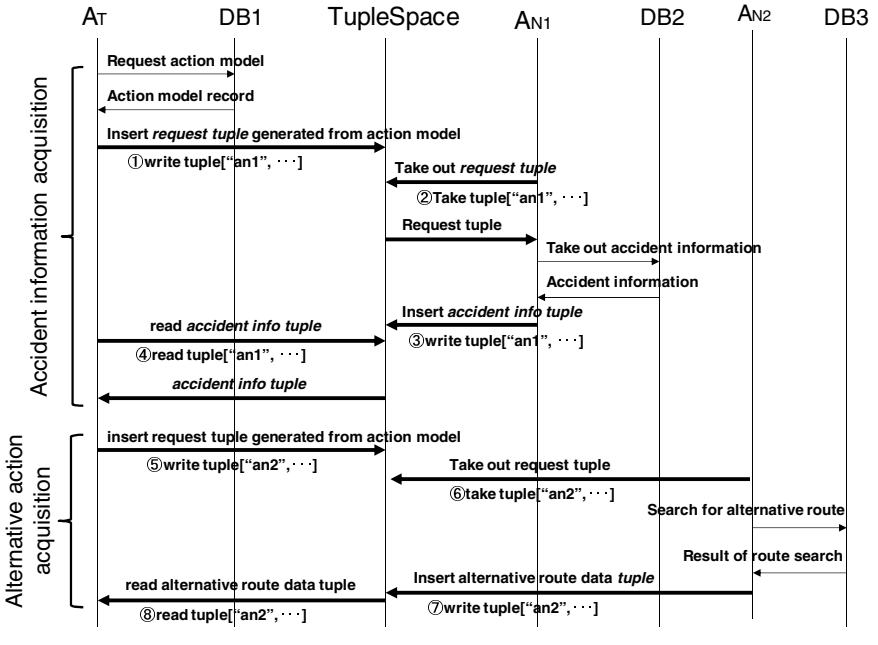



**Fig. 6.** Alternative Action Derivation Sequence in the Prototype System

### 4.2 Tuple-Matching Processing Time

In order to realize flexible communication between agents that take charge of different processes, the TupleSpace has a function to store tuple data. However, it does not hold advanced search functions such as those in MySQL. In the Tuple communication method, tuples are registered in the TupleSpace and communication is performed by matching. Therefore, if the communication between agents becomes more frequent, the number of tuples within the TupleSpace becomes large, and a delay occurs. Thus, on the prototype system, we assumed the case where multiple terminal agents share a single TupleSpace, and we verified the change in delay that occurs.

In the tuple communication in the service, since Tuple A that the terminal agent writes uses six elements – agent identification number, action start time, action end time, starting point, destination, and means of transportation – as information to perform matching, we defined a six-element tuple for the experiment and measured the matching processing time. Furthermore, a difference in matching processing time emerges between the case where matching is performed on the first element of a tuple and the case where it is performed on the sixth element. Matching performed on the first element is the fastest, and processing slows as the position of the element to be matched drops lower. Matching performed on the element in the lowest position, which, in this prototype, was the sixth element, produces the worst delay. Therefore, we conducted the following two experiments and measured the delays. We defined $n$ = 1~1400000, collected the measured value five times with $n$ set to a certain numerical value, and calculated the mean value of those five measurements.

[Experiment Condition 1] Fastest Matching

Within the TupleSpace, we registered the information [1,56,78,99,12,34] ～ [$n$,56,78,99,12,34] through a write operation, and, through a take operation on the [$n$,56,78,99,12,34] information that was inserted last, we collected the matching processing time.

[Experiment Condition 2] Worst-case Matching

Within the TupleSpace, we registered the information [34,56,78,99,12,1] ～ [34,56,78,99,12,$n$] through a write operation, and, through a take operation on the [34,56,78,99,12,$n$] information that was inserted last, we collected the matching processing time.

In the alternative-action support service, as shown in Fig. 6 ③ ⑦, two tuples are stored for each user. The reason why they are stored is that, to keep the agents on the network from performing redundant processing when they obtain the accident information and the alternative means of transportation, the accident information and the alternative means of transportation information are left in the TupleSpace. Every time the users increase, a target tuple will be searched from a TupleSpace where tuples equaling at most two times the number of users are stored. For this reason, we derived the relationship between the number of users and the matching processing delay, and we present the measurement results in Fig. 7.

From Fig. 7, it was found that the difference in the matching processing time between the two conditions became pronounced beyond 50,000 users, but that the

search processing could be done in less than a minute even when there were 55,000 users. If we consider the case of obtaining accident information at the train station with the most favorable conditions and transferring to the alternative route presented, since more than one minute is required for even an ordinary change of trains, it was found that application of TupleSpace-based inter-agent communication to the proposed alternative-action support service is possible. The prototype system was a single TupleSpace, but, by constructing a TupleSpace in multiple servers, that capacity may be expandable up to the number of users who use railways in a day.

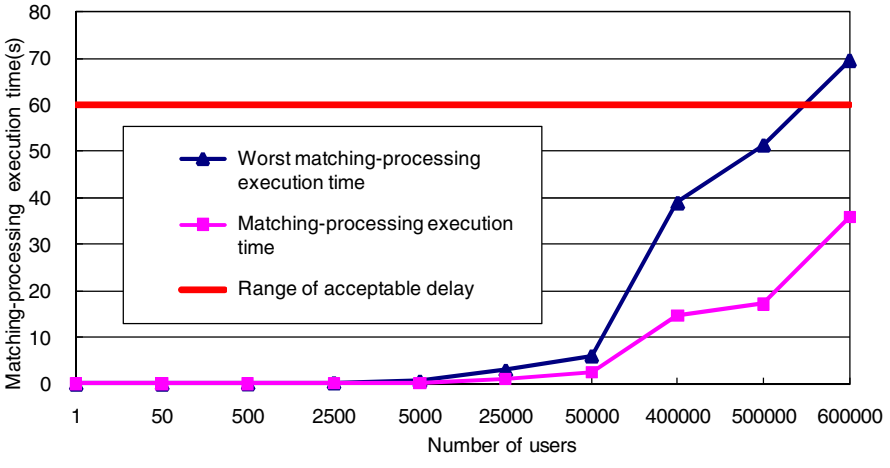



**Fig. 7.** Change in Number of Users and Processing Time in a Single TupleSpace

## 5 Conclusion and Future Challenges

In this article, we proposed an agent construction method and a service that can, by having agents operate cooperatively via a TupleSpace, propose alternative actions if users encounter situations different from their normal action models and guide the users to those alternative actions. In creating the prototype, we implemented each agent and database by using the distributed Ruby platform and MySQL, and we evaluated the feasibility of the service. In addition, we implemented an agent function allocation and a tuple definition that were designed for privacy protection. Furthermore, we measured matching processing time, which becomes a factor in the processing delay of the alternative-action support service, and we confirmed that TupleSpace-based inter-agent communication can be applied.

As a future challenge, it is necessary to investigate a high-precision algorithm for extracting action models form observable action histories. In addition, a method for collaboration with existing service providers that hold transfer information, and a method to determine the neighborhood between a user's current location and the departure location recorded in the action model, both remain as challenges for the future.

## References


1. Masatosi, M., Konomi, S., Ohnishi, K.: NAVITIME: Supporting Pedestrian Navigation in the Real World. Pervasive Computing, 21–29 (2007)
2. Anshin Gupass (March 2011), http://anshin-gp.jp/index.html
3. Rurubu mobile application (March 2011), http://www.rurubu.com/mobile/apli_1.aspx
4. Murata, M.: Towards Establishing Ambient Network Environment. IEICE Trains. Commun. E92-b(4), 1070–1076 (2009)
5. Arakawa, Y., Kashiwagi, K., Nakamura, T., Nakamura, M., Matsuo, M.: Dynamic Scaling Method of uTupleSpace Data-Sharing Mechanism for Wide Area Ubiquitous Network. In: 8th Asia-Pacific Symposium on Information and Telecommunication Technologies, APSITT 2010 (2010)
6. Lv, Q., Yang, F., Cao, Q., Li, S.: A Tuplespace-based Coordination Architecture for Service Composition in Pervasive Computing Environments. In: 12th International Conference on Computer Supported Cooperative Work in Design, CSCWD 2008 (2008), doi:10.1109/CSCWD.2008.4537012
7. Nishino, M., Yamada, T., Seko, S., Motegi, M., Muto, S., Abe, M.: Movement Prediction using Patterns of Stay with Time Infomation. In: IEICE General Conference, D-9-11 (March 2009)
8. Iwasaki, K., Niitsu, Y.: User Support System on Potential Demands by Focusing on Similar Events. In: IEICE General Conference, B-19-30 (March 2008)
9. Takata, K., Ma, J., Apduhan, B.O., Huang, R., Jin, Q.: Modeling and Analyzing Individual's Daily Activities using Lifelog. In: International Conference on Embedded Software and Systems, ICESS 2008 (2008), doi:10.1109/ICESS.2008.75
10. Kamisaka, D., Kobayashi, A., Nishiyama, S.: Proposal of Dynamic User-Behavior Modeling Method for Mobile. In: IEICE General Conference, B-19-10 (March 2007)
11. Prananto, B.H.: Ig-Jae Kim Hyoung-Gon Kim: Multi-level Experience Retrieval for the Personal Lifelog Media System. In: Third International IEEE Conference on Signal-Image Technologies and Internet-Based System, SITIS 2007 (2007), doi:10.1109/SITIS.2007.87
12. Yamaki, T., Hirano, Y., Kajita, S., Mase, K.: Modeling of Life Patterns with MuliSensor Lifelog. In: IEICE General Conference, D-12-106 (March 2005)
13. Harada, T., Kawano, Y., Otani, S., Mori, T., Sato, T.: Construction of Wireless Ad Hoc Network for Lifelog based Physical And Informational Support System. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2005 (2005), doi:10.1109/IROS.2005.1544964
14. Gotou, S., Ikeda, N., Suzuki, H.: A Proposal of the Personal Action Record Recycling System. In: IEICE General Conference, D-17-14 (March 2003)
15. Press Release of Land, Infrastructure, Transport and Tourism of Japan: About the holding result of Metropolitan Area Railway Transportation Trouble Strategy Meeting, http://www.mlit.go.jp/report/press/tetsudo08_hh_000008.html (Japanese March 2011)
16. Seki, M.: dRuby and Rinda: Implementation and Application of Distributed Ruby and its Parallel Coordination Mechanism. International Journal of Parallel Programming 37(1), 37–57 (2008), doi:10.1007/s10766-008-0086-1.
17. Seki, M.: Distributed and Web Programming with dRuby. In: Ohmsha, Ltd.
18. Well, G.C., Chalmers, A.G., Clayton, P.G.: Linda Implementation in Java for Concurrent Systems. In: Concurrency-Practice and Experience. John Wiley & Sons, Ltd., Chichester (2003)
19. Fongen, A., Larsen, C., Ghinea, G., Taylor, S.J.R.: Distributed Tuplespace and Location Management - an Integrated Perspective using Bluetooth. In: 2nd International Symposium on Wireless Pervasive Computing, ISWPC 2007 (2007), doi:10.1109/ISWPC.2007.342637

# Groups and Frequent Visitors Shaping the Space Dynamics

Karolina Baras[1,⋆] and Adriano Moreira[2]

[1] Exact Sciences and Engineering Competence Center,
University of Madeira, Campus da Penteada, 9000-390 Funchal, Portugal
kbaras@uma.pt
[2] Department of Information Systems, University of Minho, Guimarães, Portugal

**Abstract.** Our research is about a dynamic symbolic space model that is fed with data from the environment by a set of processing modules that receive raw data from sensor networks. For the conducted experiments we have been using data from a WiFi network as it is a widely available infrastructure in our campus. Here we propose two processing modules which will provide more information about the spaces described in the model. The first one tries to implement our human perception of the usual visitors of a place using two measures, the long term and the short term tenant level. The second one detects where groups of users emerge, how many there are and what are their dimensions. Based on this new perspective of the campus we intend to realize how the presence of people shapes the dynamics of a space.

**Keywords:** groups of users, space dynamics, symbolic space model, WiFi.

## 1 Introduction

Public places tend to be extremely dynamic. This means that their contexts change instantaneously as a result of different factors. These include, in the first place, the presence of people and their activities, but also changes of physical characteristics, schedules and so forth. In this paper we will focus on detecting groups of people and their relation with the space dynamics.

Our research is based on a dynamic symbolic world model which is integrated in a larger system that comprises sensor networks, data acquisition and processing modules and applications. The world model representing a particular area is fed by the results obtained from the processing of the data coming from the locally deployed sensor networks. It acts as a repository of rich context information for applications aimed at the usual users of a place or the newcomers. Our main source of context information about spaces is the wireless network which is widely deployed in the University area. Our previous results [2] have shown that the WiFi network data is quite rich in context and many interesting details about spaces can be extracted from there.

⋆ The first author would like to thank FCT for supporting this work through grant SFRH/BD/36046/2007.

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 111–122, 2011.

## 2 Related Work

During the last decade wireless LANs have increasingly become commonplace on many university campuses, in enterprises and at other public venues providing access to the Internet on the move. Every laptop, and more recently, every smartphone have also become WiFi enabled. Extended research has been made on the wireless networks usage, e.g., [4] and user mobility and traffic patterns, e.g., [3], aiming to provide useful elements that might help to improve network performance [1] and management and planning [6].

As WiFi networks spread over many public and private places, the potential for this technology to capture the dynamics of the space is enormous. We are interested in metrics that have been used for the characterization of the wireless network usage and user mobility, but our prime objective is to characterize space dynamics and to understand the usage patterns and the pulse of each observed space. For instance, the number of users per access point can provide us with some information about the popularity of an access point and, consequently, about the popularity of a space where it is deployed. We infer the way a space is used and whether it is, normally, one of the most popular areas in the campus, through the analysis of the access point tenant levels and group detection. Daily, weekly and seasonal trends give us a picture of the space pulse, contributing to its characterization. Although our approach is different and we have not found any similar work in the literature, we describe below some projects that are somehow related to ours. In [7], the authors observed a change of habits of the students concerning the choice of a place to study or socialize as a consequence of the implementation and expansion of a WiFi network. The work described in [8] links together data about the human presence in spaces based on a WiFi network data and a system that monitors HVAC levels in order to improve energy efficiency in buildings. An example of a dynamic system that is constantly and automatically updated as new people join in or new events are about to happen is the Just-for-us project [5] developed in Melbourne. This system aims to encourage new forms of social interaction in public spaces.

Although the data source is the same, our objectives are significantly different as we are using this data to populate a world model with dynamic data coming from the environment. In this paper we give two examples, the tenant levels which describe to what extent a space is visited always by the same users or by many different users, and the groups detection which allows to add more characteristics to a space description according to the number of groups, their dimension and the time they spend at each place.

## 3 A Dynamic Space Model

The world model we propose was inspired in the human mental models of the world. It is a symbolic model and it consists of *objects*, *object attributes*, *relations* and *relation attributes* as illustrated in Fig. 1. These elements are created, either manually by the user, or automatically by the processing modules. Each object

has a name, a URI (Uniform Resource Identifier) that uniquely identifies it (the object id, a host name or IP address of the server where the object is stored and a port from which it is accessible), a type, an author, a creation date, availability (*public* or *private*) and a status (*active*, *idle*, *inactive*). An object can have an arbitrary number of attributes and an arbitrary number of relations with other objects.

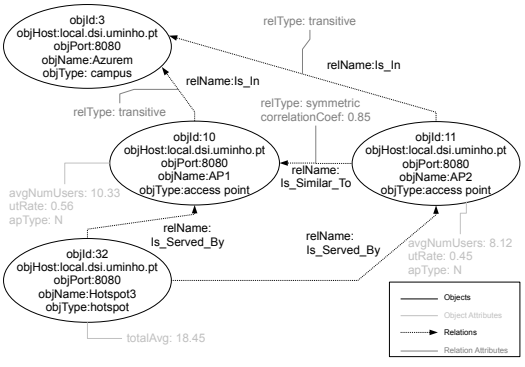



**Fig. 1.** A snapshot of the symbolic space model

***object*** = {*objid, URI, name, type, author, cdate, availability, status*}

An attribute is defined by the attribute id, the id of the described object, a timestamp and a name-value pair.

***objAttribute*** = {*attid, objid, timestamp, name, value*}

The semantics of relations is based on the way humans mentally relate objects or places. If we have a well known landmark, we will relate other objects to it, as in the following expression: *"the University is near the castle"*. In our model this is achieved through semantic relations that may be established between objects. A relation is defined by a relation id, a name, a type, a creation date, the URI of the object in the domain and the URI of the object in the co-domain. It is characterized by an arbitrary set of relation attributes. Relations support inference processes that extract implicit knowledge from the model. The way a relation is handled by the inference algorithms is based on its name and type.

***relation*** = {*relid, name, type, cdate, URIdom, URIcdom*}

Examples of the current relation names supported by our inference algorithms are: *Is_In*, which expresses spatial or administrative containment, *Is_Accessible_From*, which expresses accessibility or connectedness and *Is_Next_To* which expresses adjacency between two objects. Presently, the implemented inference algorithms support the *transitive* and the *symmetric* relation types. Other relation names and types are allowed but without any inference being performed.

A relation attribute is defined by its own id, the id of the relation to which it belongs, a timestamp and a name-value pair.

***relAttribute*** = {*relattid, relid, timestamp, name, value*}

Currently, the model is implemented on top of a relational database. There is a service, the Symbolic Contextualizer, that provides the interface to the model, and allows querying and editing the model data. The model is integrated in a bigger system, as shown in Fig. 2. The model is updated in real time by a set of processing modules that use sensor data and the model data to infer characteristics of the objects stored in the model.

In order to take advantage of the already existing infrastructures which are deployed in our University campus, we are using data from the WiFi network as our main source of context information about spaces within the campus. This option was also motivated by the results of other research projects which studied WiFi networks and found that they were a very useful source of information about space usage [7]. The Location Server in Fig. 2 stores data containing the number of users per access point and the corresponding lists of MAC addresses every five minutes. All the acquired data is accessible through a query interface provided by the Sensor Data Service.

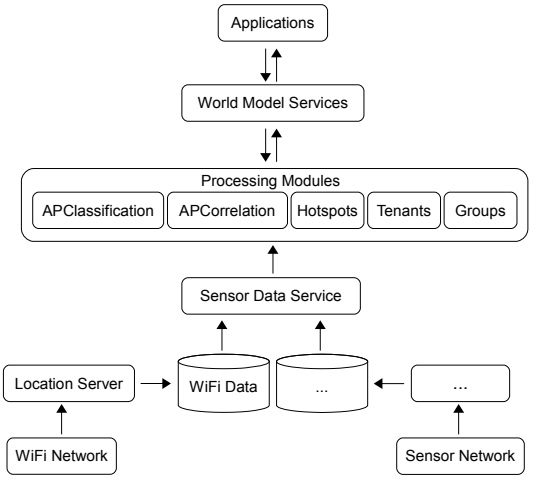



**Fig. 2.** Concise system architecture

In our previous work [2], we have analyzed data about the number of users per access point and identified useful information for creation and updating of the elements of our symbolic world model. This resulted in three processing modules, one for classification of access points according to their usage patterns, one for calculating correlation factor between access points and one for detecting hotspots that uses data produced by the two previous modules.

Next we explore the detection of frequent users and groups as two examples of processing modules that may feed the world model with data that contributes to space characterization. First, we explain the motivation for creating each of

these processing modules. Then, we identify the output that they are going to produce and, finally, the rules that should be followed for updating the model with these results. We assume that one MAC address corresponds to one person. Our objective is to observe if the users in a particular area of the campus are the usual visitors of that place or not by calculating the *tenant level* of a place as well as find and characterize the places in the campus where the groups emerge and spend their time.

## 4 Tenants, Visitors and Strangers

We propose a simple way of modeling our human perception of the usual users of a place. For instance, when we enter a familiar space, we easily notice whether the usual people are there or if there is someone new or someone missing. We propose a way of measuring this through the number of occurrences of each MAC address during a given time period. The process of calculating the *tenant level* of an access point, is the responsibility of one of the processing modules dedicated to this particular operation. Besides the calculations, it is also responsible for inserting the results in the model.

**Motivation.** Detecting a set of MAC addresses that repeatedly appear at some nodes of a network may allow for characterization of a place in terms of its users. Keeping track of the type of visitors of a particular place (place visitor profile) may let us discover whether a place is normally visited by its tenants, frequent or occasional visitors or complete strangers. This kind of knowledge may be used by context-aware applications running in the infrastructure.

**Output for the model.** Information about the frequent visitors of an access point may allow for the update of two object attributes. One describing the *long term tenant level* of a place and the other the *short term tenant level* as it will be explained in Sects. 4.1 and 4.2.

**The rules.** Update two attributes, one called *long term tenant level* and the other called *short term tenant level* with the values between 0 and 1.

For each access point, the list of connected devices' MAC addresses is recorded every five minutes. Frequencies of each detected MAC address are stored and updated. Fig. 3 shows an example in which data was stored during 24h resulting in 288 samples. In each sample the number of connected MACs per access point is recorded and the counter of occurrences for each present MAC is incremented. After the acquisition of samples during a time period, the available data consists of the frequencies of each detected MAC address. In our experiments the considered time periods were 20 days (corresponds to 5760 samples) for the long term and 5 minutes (a single sample) for the short term.

### 4.1 Long Term Tenant Level

The *long term tenant level* shows if a place is mostly visited by people who are always there, frequently dropping in or just occasionally passing by. It is

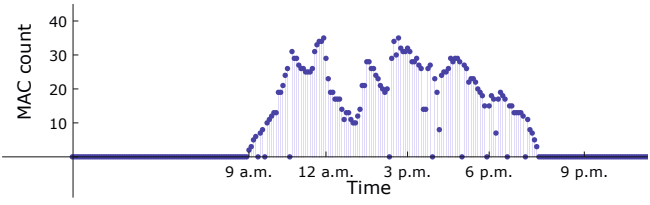



**Fig. 3.** 24h hours of MAC counts for an AP

calculated for the data collected during a fairly long time period, like several weeks, in the following way:

1. Calculate absolute frequencies for each observed MAC address $i$ ($f_{\text{abs},i}$);
2. Calculate normalized frequencies of observed MACs: $f_{\text{norm},i} = f_{\text{abs},i}/S$, where $S$ is the number of samples in which at least one MAC was detected (valid samples). This value is chosen instead of the total number of samples that are taken during the considered time interval in order to discard periods in which nobody connects, e.g. during the night or during the weekend. In places that never close and where there are always people, it may be more suitable to consider the total number of samples. In Fig.3, $S$ is given by the number of time intervals between 9 a.m. and 7.30 p.m.;
3. Classify MACs based on their normalized frequencies and according to the rules described bellow in Table1. The result is depicted in Fig.4 for a set of 19 MACs randomly chosen out of the total of 720 MACs detected during the observation period (20 days).

**Table 1.** Place visitor profile based on detected MACs

| Normalized frequency ($f_{\text{norm}}$) | MAC Class |
|---|---|
| $f_{\text{norm}} \geq 0.2$ | Tenants |
| $0.1 \leq f_{\text{norm}} < 0.2$ | Frequent Visitors |
| $0.01 \leq f_{\text{norm}} < 0.1$ | Visitors |
| $f_{\text{norm}} < 0.01$ | Strangers |

4. Count the total number of detected MACs ($n_{\text{MACs}}$).
5. Calculate the final value for the long term tenant level:

$$tenantLevel_{\text{LT}} = \frac{f_{\text{normT}} + f_{\text{normFV}} + f_{\text{normV}}}{n_{\text{MACs}}} \tag{1}$$

which is the sum of all the normalized frequencies higher or equal to 0.01 (all tenants, frequent visitors and visitors) divided by the total number of detected MACs. This results in a value between 0 and 1 for the *long term tenant level* of a place.

For the example illustrated in Fig.4, the calculated normalized frequencies reveal the detection of 6 tenants, 1 frequent visitor, 7 visitors, and 5 strangers. The resulting tenant level follows from (1): $tenantLevel_{\text{LT}} = 0.13$.

### 4.2 Short Term Tenant Level

The *short term tenant level* shows, in turn, if the considered area is being visited at a particular moment by its tenants, frequent visitors, visitors or strangers. The *short term tenant level* is calculated for a single sample as follows:

1. Count the number of tenants, frequent visitors, visitors and strangers in the current sample by comparison to their class, if any, in the long term place visitor profile. MACs appearing for the first time are considered strangers;
2. Calculate the value for the *short term tenant level* as:

$$tenantLevel_{\mathrm{ST}} = \frac{f_{\mathrm{normT}} + f_{\mathrm{normFV}} + f_{\mathrm{normV}}}{m_{\mathrm{MACs}}} \tag{2}$$

which is the sum of the normalized frequencies higher or equal to 0.01 divided by the total number of detected MACs (tenants, frequent visitors, visitors and strangers) in the current sample. This results in a value between 0 and 1 for the *short term tenant level* of a place.

For the same case from the Fig.4, where we calculated the *long term tenant level* for an access point with 19 detected devices, we will now calculate the *short term tenant level*. To do this, we will consider that, the following devices were detected in a single sample: A, D, J, K, L and N. According to the figure, we have 2 tenants (D, K), 1 frequent visitor (N), 1 visitor (A) and 1 stranger (L). If we use the same values for their normalized frequencies as those shown in the figure, (2) yields $tenantLevel_{\mathrm{ST}} = 0.21$. So it results in a slightly different value than the one obtained for the long term. Although a place may be mostly visited by occasional visitors, there may exist some time periods when it is visited by its tenants and frequent visitors.

For the following examples the *long term tenant level* was calculated for a 20 days long data set and the *short term tenant level* was calculated for each single sample during a 48h time period (from Sunday, 5 p.m. to Tuesday 5 p.m.). The first example (Fig.5(a)) is from one of the library access points which is usually visited by a very large number of users during the opening hours. The second example (Fig.5(b)) is from a rarely used access point, localized near the design rooms in the department of architecture. The third example (Fig.5(c)) shows the results for the set of four access points in the students' dormitories which

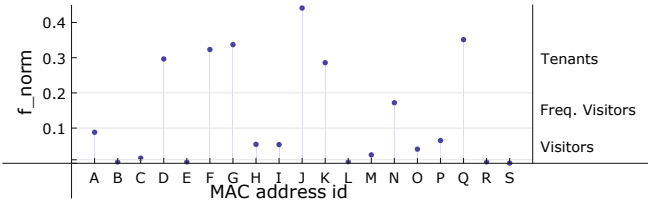
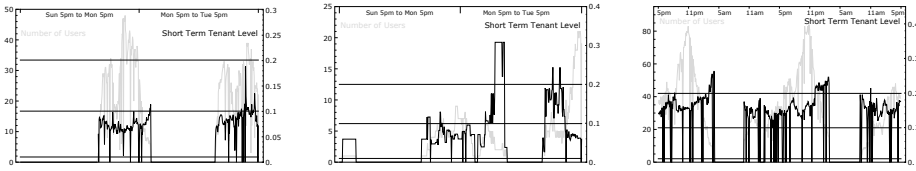



**Fig. 4.** An example of a place visitor profile based on a set of MACs detected at an access point during an observation period of 20 days




(a) The library is characterized by many occasional visitors.

(b) Design rooms are characterized by a few tenants when no one else is there.

(c) Dormitories are characterized by frequent visitors and tenants.

**Fig. 5.** Short term tenant level variation during two days time period. Left Y-axis represents the number of users and the right Y-axis, the short term tenant level.

are usually heavily used during all week. It shows strong frequent visitor, and sometimes, tenant profile.

It makes sense that in the library the visitor profile is characterized by many occasional visitors as it is a place where a large portion of the academic population spends some time during a week. However, the characterization of the design rooms profile is not as clear, because there are many fluctuations during the day and during the week. At some hours of the day only tenants are found, but as soon as the number of users increases, the tenant level value lowers. In the dormitories the profile is quite clear. Although there is a large number of users like in the library, this place is mostly used by its residents. Although it is not always clear whether the users connected at a particular access point are its tenants or occasional visitors, the results show a new perspective on the space usage. New knowledge about a space may be extracted from graphics like those shown in Fig. 5.

## 5 Groups of Users

In this Section we propose a processing module that is responsible for the detection of groups of users in a WiFi network coverage area. This is achieved based on the MAC address listings retrieved from the access points. This module also inserts the resulting data about groups into the space model.

**Definition 1.** *An iteration consists of six consecutive samples, taken every five minutes.*

This means that a single iteration is 30 minutes long. Due to overlapping (see Fig. 6) the duration of two consecutive iterations is 35 minutes.

**Definition 2.** *A group is defined as a set of two or more MAC addresses that are detected together at two different access points, during at least four iterations ($N_{\text{iter}} = 4$) at one of them.*

As explained above and shown in Fig. 6 the duration of this time interval may vary, depending on whether the MACs X and Y are found together in consecutive or non-consecutive iterations.

The minimum number of iterations for the creation of a group determines the number of detected groups. Four iterations for a week time is a reasonable value because it means that the members of the group are found together during at least 45 minutes, which is the minimum duration of a lecture. Also, the constraint of at least two different access points is important for it helps to eliminate laboratory equipment that is connected to the WiFi network unattended. An enhancement could be to establish a minimum distance between the two different access points in order to eliminate the "ping-pong" effect which may occur between nearby access points.

**Motivation.** The purpose of detecting groups of users is to extract more characteristics of the spaces where these groups are created, according to their number and their dimension. Identification of groups of devices, and to some extent, groups of users, allows for the identification of the areas in the campus where the groups of users work or meet. It also may add more dynamics to our model as it reflects their presence and movement within the area represented in the model.

**Output for the model.** When a group is identified, an object representing it may be created in the model. A relation with an access point may also be created every time a group connects to the network. There may be a list of all visited access points as an attribute of a group that may be used for inference of additional information about the group itself and the places it visits.

**The rules.** Two or more MAC addresses should be found connected to at least two access points during at least $N_{\text{iter}}$ iterations for an object of type *group* to be created. Once created, this object will have a relation *Is_In_Range_Of* established with an access point.

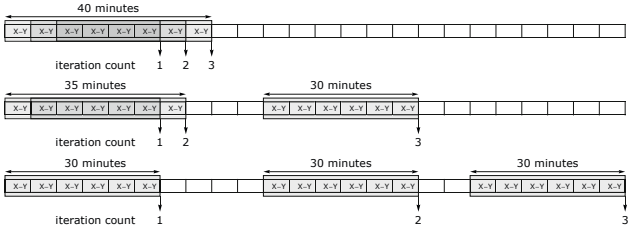
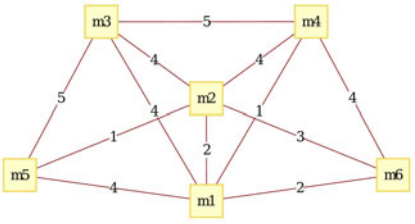
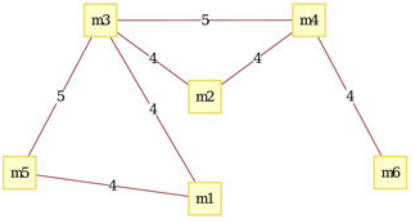



**Fig. 6.** Duration of the total number of counted iterations is variable. Three iterations may mean that the members of a group were found together during 40 min, 65 min or 90 min. The minimum of minutes, given by $30 + (N_{\text{iter}} - 1) \times 5$, is obtained if all the iterations are consecutive (partially overlapping) and the maximum, equal to $N_{\text{iter}} \times 30$ is obtained if there is no overlapping between iterations.

Fig. 7(a) shows a graph representation of a group in which vertices correspond to MAC addresses that were found connected to an access point during a considered time interval. The weight associated with each edge represents the number of iterations during which the pair of MACs was observed. If we set a threshold




(a) Edges between MAC addresses correspond to the number of iterations.




(b) When a threshold of 4 is set for the minimum number of iterations, edges with lower weights are removed.

**Fig. 7.** Graph representation of a group of users

to 4 iterations ($N_{\text{iter}} = 4$), as we did in our experiments, all the edges with weights lower than 4 are removed as shown in Fig.7(b).

In our experiments, we detected 50 groups on average during one day in the campus. During one week we detected 830 groups on average. We also identified places in the campus where the largest number of groups emerge and also where the largest groups appear. Each of the considered places is covered by at least two and at most four access points. As shown in Fig.8, the most frequent group dimension is 2. The largest group had 42 members and it was detected in the dormitories.

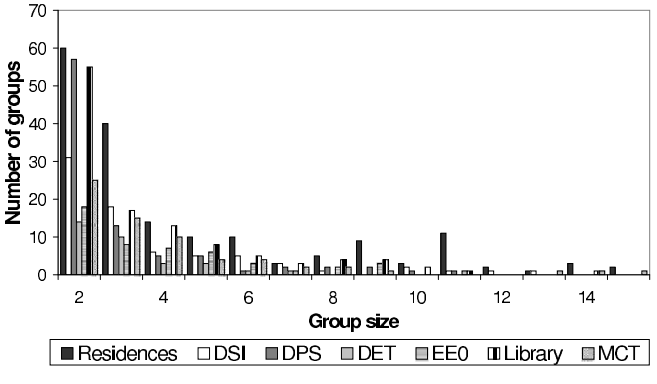



**Fig. 8.** Dimensions and number of groups for a week long data set. Larger groups were detected, 42 being the biggest. They are not shown here as they are not very frequent.

In comparison to the total number of unique MACs detected (3222), those participating in groups (945) represent nearly 30%. As expected, the most crowded places are the library and the dormitories and there appear the largest number of groups. We identified 7 places in the campus where 70% of the groups emerge (see Fig.9). If we do not consider the dormitories, than we have 50% of all groups emerging in 6 places.

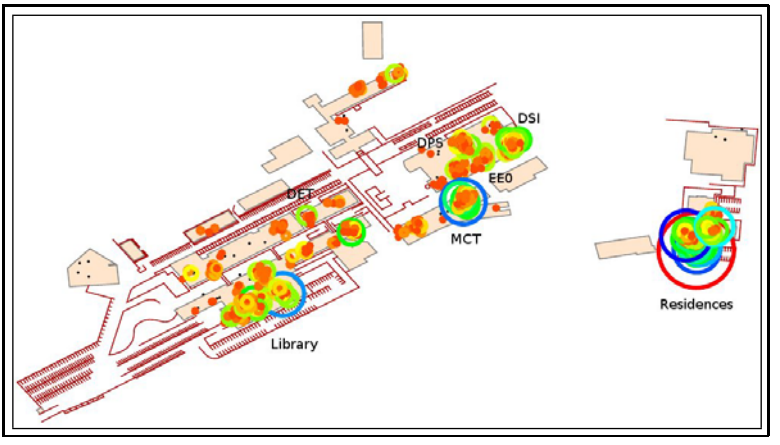



**Fig. 9.** Groups at the campus after analyzing a week long data set. The smallest groups have 2 members and the largest 42 (the biggest circle). The radius of the circle is proportional to the dimension of the group.

## 6 Conclusions and Future Work

Starting with a structure of a dynamic world model and the data about the utilization of an University WiFi network, we have identified two additional sources of information for the update and the expansion of the developed model. We have defined a new parameter for characterization of spaces, the *tenant level*. This parameter translates the human ability to distinguish whether a familiar place is occupied by the usual users or there are new people around. The second source of data for our model comes from the detection of groups. We were able to detect where the groups emerge and what are their dimensions. These two new aspects add a new perspective on spaces derived from data gathered from the WiFi network.

Further developments may be done based on the results from this paper. For example, we may study the movement of groups around the campus and identify the most frequent trajectories. Besides the individual group detection, we can also observe the interconnections between all the existent groups in the campus. This happens due to the fact that people participate in several social groups and so they act as links between these groups. So, if we observe a week long data in all access points at once, we obtain a large group containing the great majority of detected MACs and several smaller groups that remain separated, probably representing occasional visitors. The social hierarchy of the users of a WiFi network may be studied through the observation of the groups that are formed at different levels of detail, according to the value of $N_{\text{iter}}$ parameter. The largest group we found contains nearly all the users of the considered WiFi network. Inside the largest group of users, smaller groups consist of people belonging to

different courses and classes, inside the classes we can find groups of friends, and so on.

We have developed a prototype application that shows the current number of users for a set of APs through a color code in an attempt to illustrate the current state of the campus. For the future work, we intend to develop a new and more complete version of this application in order to allow for the visualization of the state of the campus in real time. A screen shot of that application will be something similar to Fig. 9. This application may be targeted at people coming to the campus for the first time to help them find the places of interest inside the campus. Further information about our ongoing research may be found at http://ubicomp.algoritmi.uminho.pt/symbolic/.

## References

1. Balachandran, A., Voelker, G.M., Bahl, P., Rangan, P.V.: Characterizing user behavior and network performance in a public wireless LAN. In: SIGMETRICS 2002: Proceedings of the 2002 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems, pp. 195–205. ACM, New York (2002)
2. Baras, K., Moreira, A.: Symbolic space modeling based on WiFi network data analysis. In: 2010 Seventh International Conference on Networked Sensing Systems, INSS (2010)
3. Henderson, T., Kotz, D., Abyzov, I.: The changing usage of a mature campus-wide wireless network. Computer Networks 52(14), 2690–2712 (2008)
4. Kim, M., Kotz, D.: Periodic properties of user mobility and access-point popularity. Personal Ubiquitous Computing 11(6), 465–479 (2007)
5. Kjeldskov, J., Paay, J.: Public pervasive computing: Making the invisible visible. IEEE Computer 39(9), 60–65 (2006)
6. Schwab, D., Bunt, R.: Characterising the use of a campus wireless network. In: Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM 2004, vol. 2, pp. 862–870 (March 2004)
7. Sevtsuk, A., Huang, S., Calabrese, F., Ratti, C.: Mapping the MIT campus in real time using WiFi. In: Foth, M. (ed.) Handbook of Research on Urban Informatics: The Practice and Promise of the Real-Time City, pp. 326–337. IGI Global, Hershey PA (2008)
8. Vaccari, A., Samouhos, S., Glicksman, L., Ratti, C.: MIT enernet: Correlating WiFi activity to human occupancy. In: Proceedings of Healthy Buildings 2009, the 9th International Conference and Exhibition (2009)

# Multi-sensor Data Fusion within the Belief Functions Framework

## Application to Smart Home Services

Bastien Pietropaoli[1], Michele Dominici[1], and Frédéric Weis[2,⋆]

[1] INRIA, Rennes-Bretagne Atlantique,
Campus Universitaire de Beaulieu
35042 Rennes Cedex, France
{Bastien.Pietropaoli,Michele.Dominici}@inria.fr
http://www.inria.fr

[2] IRISA, Université de Rennes 1,
Campus Universitaire de Beaulieu
35042 Rennes Cedex, France
Frederic.Weis@irisa.fr
http://www.irisa.fr

**Abstract.** In Smart Home, understanding the environment and what is going on is the basis of all adapted services. Unfortunately, inferring situations and activity recognition directly from raw data is way too complex to be applied. Firstly, we present a layered architecture we are building to process raw data into abstract situations and activities. Secondly, data fusion tools using the belief functions theory are introduced as a general framework to provide a first level of abstraction from raw data given by sensors to a more complex context model. Then a methodology to apply the model to our Smart Home within the belief functions framework, a first implementation and the encountered issues in modeling are discussed.

**Keywords:** Smart Home, Ubiquitous Computing, Data fusion, Belief Functions Theory.

## 1 Introduction and Motivation

A *smart home* is a residence equipped with information-and-communication-technology devices conceived to collaborate in order to anticipate and respond to the needs of the occupants, working to promote their comfort, convenience, security and entertainment while preserving their natural interaction with the environment [1]. When talking about natural interaction, one of the most precious resources to preserve is user attention: during their activities, users should be supported *invisibly*, reducing interruptions and explicit interactions with the system as much as possible. In order to achieve these goals, smart home systems

⋆ The authors would like to thank EDF that has funded this research.

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 123–134, 2011.

must be able to take into account the *context* to provide adapted functionalities. For example, under some conditions, knowing that an inhabitant is executing some long-lasting static activity in a room can suggest that the system should turn on the room's heating and turn off the other rooms' lights.

Existing solutions for human activity recognition often rely on data coming from wearable sensors or video cameras, technologies that are difficult to deploy and get accepted in real-world households. Furthermore, these solutions address the problem of activity pattern discovery directly on raw sensor data or video streams, exploiting data mining techniques to extract recurring patterns in the raw data and to predict or classify future observations, as explained in [2]. The resulting systems fail to provide adapted services to people in real-world scenarios, as the "gap" between the captured context and the complexity of human behavior is too large.

In this paper, we present how it is possible to reduce this gap by furnishing an intermediate level of abstraction between the sensors and the context model. To accomplish this, data fusion tools within the belief functions theory are exploited to convert raw data into a first-level abstraction such as a posture of someone or a presence in a room. In section 2, we present a functional architecture that is designed to extract high-level situations from low-level raw sensor data and a system architecture leveraging the ubiquitous computing principles, core of the prototype system that we are developing. In section 3, we highlight the main problems appearing when trying to fuse raw data from multiple heterogeneous sensors. The belief functions theory basics is also introduced in order to emphasize the advantages of this framework w.r.t. to our work. In section 4, an application of the belief functions theory is suggested as a methodology for data fusion. A first experimentation and some performance results are also presented. In section 5, the open issues remaining in our approach are discussed, in particular on how to build the belief functions and the real meaning of the obtained results in order to communicate with the higher-level layers. Then, section 6 concludes the paper.

## 2 Architecture

In this section, we present the architecture of a prototype system that we are developing. The aim of our system is to capture physical information from the environment, extract higher-level concepts and combine them to infer human situations and activities, with the ultimate goal of semi-automatically managing household appliances and provide additional functionalities that allow saving energy while preserving comfort.

To achieve the goals of our scenario, it is natural to adopt the human-computer interaction paradigm called *ubiquitous computing*. The aim of this paradigm is to seamlessly and invisibly integrate in the physical environment a multitude of digital devices that provide services to the users without asking for their attention [3]. To this end, ubiquitous computing applications are typically context-aware, where the word context is used to address any static or dynamic condition that

concerns the digital, physical and user-related environments in which a context-aware application is executed. In [4], a four-layer model is suggested to build context-aware applications, as showed in Fig. 1.

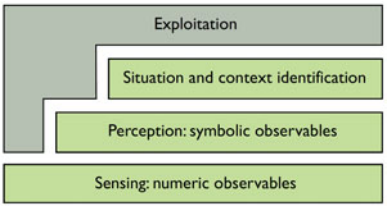



**Fig. 1.** The four-layer model for context-aware applications proposed by [4]

The first layer, *sensing*, corresponds to the raw data sensed from the environment. The second layer, called *perception*, can be interpreted as an abstraction of the raw data. *Situation and context identification*, the third layer, concerns the context itself and the situations occurring in the home. The top layer, called *exploitation*, provides contextual information to applications. Our work is partly based on this model.

Considering the aforementioned model, the first layer of our system should be simply composed of sensors [5], but some constraints have to be fitted. In order to reduce the global system cost and to protect the inhabitants' privacy, the number of sensors dispatched in the environment has to be reduced as much as possible. However, a huge number of different sensors are required to sense context pieces and redundancy can significantly increase the reliability of the sources. With this idea in mind, the sensors will be grouped in nodes, as showed in Fig. 2. These nodes are able to preprocess the data with simple computation such as minimum, maximum and average. They also enable the sensors to communicate, using, for instance, 6LowPAN (IPv6 over LoW Power wireless Area Networks), which is specifically designed for embedded systems [6]. Another benefit of using nodes is the optimization of energy consumption due to radio communications. This is not negligible as most of the nodes will be running on batteries.

In the second layer of Coutaz' model, the raw data are processed to obtain more abstract data about context and occurring situations. It could be, for instance, a presence in a room, the number of people in this room or the posture of someone. The aggregation of raw data is realized thanks to a data fusion algorithm. The one we adopted is called the *belief functions theory* or *theory of evidence*. Sections 3 and 4 explain with more details this theory and how we try to apply it to our problem.

The bridge between the second and the third layer is realized integrating the results of sensor data fusion into a context model called *Context Spaces*. This model uses geometrical metaphors to describe context and situations, relying on the following concepts [7]: the *context attributes*, the *application space*, the *situation spaces* and the *context state*. The *context attributes* are low-level contextual information types that are relevant and obtainable by the system, like a

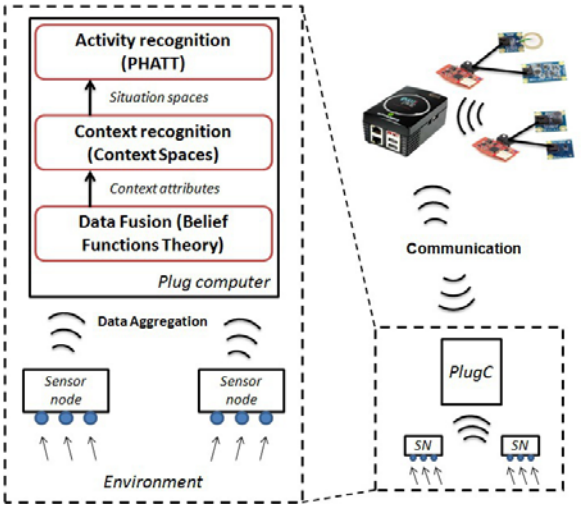



**Fig. 2.** The system architecture – The sensor nodes send aggregated data to the plug computers, which are in charge of performing sensor data fusion, to produce the context attributes, and context spaces reasoning, to infer the ongoing situation spaces and activities

posture; their values are provided by the perception layer, together with a degree of *confidence* on them, needed to cope with the intrinsic uncertainty of sensing systems in real world scenarios. The *application space* is a multi-dimensional space made up of a domain of values for each context attribute. The *situation spaces* are subspaces of the application space defined over regions of acceptable values of selected context attributes; situation spaces model real-life situations, e.g., "the whole family is in the kitchen" or "a person is ironing". A *context state* is the collection of current context attribute values at a given moment [7].

In the *situation and context identification* layer (see Fig. 1), the context state provided by the perception layer is analyzed to infer the ongoing situation spaces (representing real-life situations) and also produce a measure of confidence in their occurrence. As the same context state can correspond to several different situation spaces (and vice versa), reasoning techniques are needed to discern the actual ongoing real-life situations in spite of uncertainty [7].

Leveraging the situation-detection capability provided by the Context Spaces theory, an additional mechanism is used by the *situation and context identification* layer to extract information about ongoing complex inhabitant activities. This mechanism is realized adapting an existing plan recognition algorithm, called PHATT [8,9], in order to integrate with our architecture, as explained in [10]. This mechanism takes as input the context dynamics, concretely represented by the flow of observed Situation Spaces that occur over time, and extracts information about ongoing complex inhabitant activities [10].

Unfortunately, the computations required by the second and the third layers to obtain abstract data and to analyze context and situations are too heavy for

our nodes to be processed on. To remedy to this problem, more powerful nodes acting like sinks are used. These nodes are small "plug and play" computers called *plug computers* [11](ref. Fig. 2). Their role is to perform the algorithms of layers 2 and 3.

As explained in [4], the *exploitation* layer acts as an adapter, allowing applications to address to the infrastructure their requests for context services at a high level of abstraction. In our architecture, this layer will provide information about context to augmented appliances, which will adapt their behavior in a semi-automatic way.

In the presented architecture, many research problems appear at different levels. This paper focuses only on the low-level data fusion to provide context models with a first abstraction from raw data sent by sensor nodes.

## 3 Data Fusion and Belief Functions

In this section, we highlight the main problem encountered when fusing data due to data imperfection. In the existing data fusion methods [12], only the belief functions theory has been retained. Thus, the theory basics are presented as well as the main tools used in section 4.

### 3.1 Data Imperfection

In our problem, different kinds of sensors are required to sense the environment. Unfortunately, a lot of problems occur because of data heterogeneity and data loss in wireless communications. As presented in [13], data may also be imperfect for various reasons such as:

- Randomness, due to physical systems (in our case, sensors);
- Inconsistency, due to overload of data or conflicting sources;
- Incompleteness, due to loss of data which may easily happen with wireless sensors;
- Ambiguity (or fuzziness), due to the model or to the natural language imprecision;
- Uncertainty, due to not fully reliable sources;
- Bias, due to systematic errors;
- Redundancy, due to multiple sources giving the same information.

Actually, redundancy is not always an imperfection. It can help to build a better idea about what is really happening in the environment and may be also used to increase the precision on measures or to detect failures when at least half of the sources are reliable.

To process imperfect data, a lot of uncertainty theories exist [12] but we use, for now, only one of them, the belief functions theory (BFT) developed by G. Shafer in [14]. It furnishes tools to process incomplete or imprecise data and to manage conflict and uncertainty.

### 3.2 Belief Functions Theory

Here, we will introduce the basics required in our case of multi-sensor data fusion. All these tools will be used and illustrated with examples in section 4. In the BFT, the first thing that should be defined is a set of possible worlds $\Omega = \{\omega_1, \omega_2, ..., \omega_n\}$ called the *frame of discernment*. These worlds have to be exclusive and if possible exhaustive. To give an example, we can define a set of the possible postures of someone by $\Omega = \{Seated, Standing, LyingDown\}$. Once the frame of discernment is created, a *mass function* (also called *basic belief assignment* or *body of evidence*) representing the degree of belief associated to each subset of $\Omega$ is defined such that:

$$\begin{aligned} m : 2^{\Omega} &\longmapsto [0,1] \\ \sum_{A \subseteq \Omega} m(A) &= 1 \end{aligned} \tag{1}$$

Every subset A with $m(A) > 0$ is called a *focal set* and may be considered as a part of belief. As the mass functions are applied to the powerset of $\Omega$ (the set of all subsets of $\Omega$), contrary to the probability theory, the beliefs may be non-specific, i.e. accorded to a set of possible worlds. Thus, the belief functions theory offers a double way to express the uncertainty using degrees of belief but also non-specificity.

Other functions strictly equivalent to the mass function are defined from the latter. The first one, called *belief*, defined by

$$bel(A) = \sum_{\emptyset \neq B \subseteq A} m(B) \tag{2}$$

may be interpreted as the degree of certainty that something is occurring or the probability of provability associated to a subset of $\Omega$. The second one, called *plausibility*, defined by

$$pl(A) = \sum_{B \cap A \neq \emptyset} m(B) \tag{3}$$

can be interpreted as the support accorded to the fact that something is possible. The mass functions, or bodies of evidence, can be combined using the *unnormalized Dempster's rule of combination* introduced by Smets et al. in [15] and given by

$$m_1 \textcircled{\cap} m_2(A) = \sum_{B \cap C = A} m_1(B) m_2(C) \tag{4}$$

The mass on the empty set $m(\emptyset)$ under the *open-world assumption* is the degree of belief that the true state of the world is unidentified, i.e. not defined in the frame of discernment. It is also often interpreted as the conflict resulting from the combination of evidence. The Dempster's rule of combination is normalized by this conflict in this way:

$$m_1 \oplus m_2(A) = \frac{\sum_{B \cap C = A \neq \emptyset} m_1(B) m_2(C)}{1 - m_1 \textcircled{\cap} m_2(\emptyset)} \tag{5}$$

The *vacuous mass function* ($m(\Omega) = 1$) is the neutral element for these combination rules. A lot of different rules of combination have been proposed to redistribute the conflict on the subsets of $\Omega$ or to manage the combination of conflicting sources. Without being exhaustive, it is possible to find in the literature the rules of combination of R.R. Yager [16], D. Dubois and H. Prade [17], C.K. Murphy [18], L.-Z. Chen [19] and a generalization by E. Lefevre [20].

As seen in section 3.1, the data may be imperfect and the sources not reliable. Thus, a discounting operator, which transfers a part of the mass to the complete set, i.e. the total ignorance, has been defined such that

$$m_\alpha(A) = \begin{cases} \alpha.m(A) \text{ if } A \subseteq \Omega \text{ and } A \neq \Omega \\ \alpha.m(A) + (1 - \alpha) \text{ if } A = \Omega \end{cases} \text{ with } \alpha \in [0, 1] \tag{6}$$

All the presented tools enable the manipulation and the accumulation of evidence that something is occurring. They are applied to the problem of data fusion in order to compute pieces of context identified in our architecture in the section 2. However, no clue is given about how to use them in a real context and how to build the mass functions. The next section presents those aspects.

## 4 Applying BFT to Multi-sensor Data Fusion

In the previous sections, we presented the encountered problems when fusing heterogeneous data. Those problems are particularly true in our architecture presented in section 2 where data are gathered by GuruPlugs [11], acting as sinks, from sensor nodes via wireless communication. In this section, we illustrate how we can apply the belief functions theory to compute context attributes, e.g. a posture or a presence in a room, from collected raw data. Different methods are suggested to build mass functions from sensor measures. Then, we try to explain how we compute our context attributes. Finally, the results of a first experimentation of the belief functions on an embedded system are presented.

### 4.1 Building Belief Functions

There exist methods to build belief functions from statistics and probabilities [21,22], but they are way too much complex to be detailed here. Moreover, the precise probability of occurrence of different worlds given a sensor measure is nearly impossible to obtain even with experience. For simplicity's sake, we suggest the use of easier methods to build our belief functions.

**Discounting.** A first way to obtain mass functions using Boolean sensors (true - false) is to consider that the information about the state is reliable and then

to add ignorance due to the error rate $\alpha$ of the sensor using the discounting operation [23,24]. Thus, a *simple mass function* is obtained with the form

$$\begin{aligned} m(s) &= 1 - \alpha \\ m(\Omega) &= \alpha \end{aligned} \tag{7}$$

where $s$ is the actual state of the sensor. The error rate $\alpha$ can be obtained via the sensor data sheet, via statistics and may vary in time to take into account the sensor wear-out or the low-battery effects.

Another use of discounting is the elimination, or at least the impact minimization, of the minority opinion. In [25], a conflict between sources is computed two by two to analyze which one is in contradiction with others. Then, the discounting factor is computed given the global conflict of a source among others. The Chen's combination rule [19] applies a similar philosophy to eliminate the conflicting sources using a weighted average, more weight being accorded to the majority opinion.

**Sets of mass functions.** As in [25], it is possible to define for each sensor and for each context attribute we would like to compute, a set of mass functions on which a projection is done to obtain the body of evidence given a sensor measure.

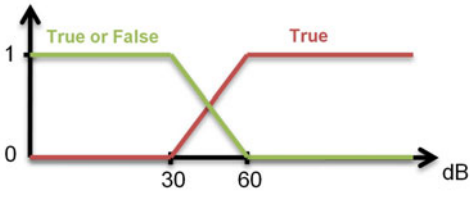



**Fig. 3.** Example of a set of mass functions associated to a sound sensor in the case of the context attribute "Presence in a room"

In the example on Fig.3, if a sound sensor returns a measure of 54dB, the mass function associated to this measure will be given by two focal sets with the values $m(True) = 0.8$ and $m(True\ or\ False = \Omega) = 0.2$. This mass function is then considered as a body of evidence that someone may be present in the room, i.e. the context attribute "Presence in a room". Of course, the discounting operation can be applied on the mass function obtained in this way to take into account the sensor error rates.

One thing to respect when building the set of mass functions is the *least commitment principle*: the specificity should not be too high when the sensor measure is clearly not good evidence. Always in the given example of the sound sensor, the fact that there is noise in a room may be a good indicator that someone is present. On the contrary, the fact that no noise is detected is not a proof that the room is empty. This is why the mass function associated to a

nearly no-noise-measure should be vacuous ($m(\Omega) = 1$) to stay neutral when combining with other bodies of evidence. Thus, the system, when evidence are not sufficient, is capable to stay in a neutral state of total ignorance.

### 4.2 Computation of Context Attributes

In our architecture defined in section 2, the belief functions theory is used to compute pieces of context we call context attributes such as a posture or a presence. To compute them, sensors capable of bringing some evidence are required. When sufficient evidence is gathered, the obtained mass functions are combined using, for now, the Dempster's rule of combination. Unfortunately, the mass functions are not sufficient to reason about which subset should be selected as containing the true state of the world. The belief (bel) and plausibility (pl) functions, both defined in the section 3, are too pessimistic and too optimistic to make a decision on which world to select. Thus, Ph. Smets [26] suggested to separate the *credal* part where evidence is accumulated and the *pignistic* part where the decisions are made. He introduced the *pignistic transformation* given by

$$BetP(A) = \sum_{B \subseteq \Omega} \frac{|A \cap B|}{|B|} \text{ with } |.| \text{ the cardinality of a set} \tag{8}$$

which is a neutral compromise between the belief and the plausibility ($bel(A) \leq BetP(A) \leq pl(A)\ \forall A \subseteq \Omega$). It can be considered as a bet on the probability that something is occurring and is often a better decision criterion than the belief or the plausibility.

One can be tempted to make a decision only on one possible world. Within the belief functions, the certainty or the result of the pignistic transformation can be drastically increased by reducing the precision of the results, i.e. increasing the cardinality of the selected subset of possible worlds. Hence, the final result may be non-specific but certain like "I'm sure that someone is either seated either standing" compared to a specific but uncertain result like "I'm half sure that someone is seated". In some cases, for example to know if someone is awake, knowing that the person is just not lying down may be sufficient.

This non-specificity can also be a good way to implement fault-tolerant mechanisms. If a sensor crashes, the fact that some evidence is missing could only reduce the specificity of the final result without impacting the certainty. As explained, sometimes the certainty is more important than the specificity.

### 4.3 First Experimentation

In our project, the system should be able to provide quick and adapted services to the occupants. The functional architecture presented in section 2 requires a lot of computation and unfortunately, the sensor nodes are computationally limited. Thus, tests must be run on embedded systems to demonstrate the feasibility of our approach. The basics presented in section 3 and some additional combination rules and functions such as the commonality [27] or the distance [19] have

been implemented using the C language in order to be tested on an embedded computer. Performance tests have been run on a GuruPlug [11] equipped with a 1.2GHz Sheeva CPU Core which is quite limited compared to a basic desktop computer. The tests consist in simulating a set of sensor measures to create the mass functions from the sets of mass functions presented in section 4.1. Then, the basic operations such as combination, distance, belief, plausibility and so on are executed for all these mass functions. The results are compiled in Tab. 1.

**Table 1.** Times of execution of a set of basic operations (combinations, distances, bel, pl, etc...) according to the number of simulated sensors and the size of the frame of discernment (possible worlds)

| Possible worlds | 5 sensors | 10 sensors | 15 sensors | 20 sensors |
|---|---|---|---|---|
| 3 | 12.2ms | 29.7ms | 51.3ms | 77.4ms |
| 5 | 14.1ms | 33.3ms | 58.5ms | 87.4ms |
| 7 | 16.2ms | 38.3ms | 66.3ms | 99.1ms |
| 9 | 20.8ms | 48.6ms | 82.4ms | 122.0ms |

The results are promising as the time of execution stay reasonable to provide quick services. However, it can be seen that adding several sensors to our system can have a dramatic effect on the execution time. The size of the frame of discernment can be also crucial. Hence, in our cases, we try to reduce as much as possible the size of the set of possible worlds, considering only few possible values for context attributes. The number of sensors can be limited considering only a sufficient number of evidences, by creating subgroups of sensors, required to obtain a satisfying certainty. Thus, a lot of sensors may be used only in case of emergency, for instance if some sensors crash, or if the certainty is not sufficiently high for the context model to reason on the context attributes.

## 5 Open Issues

Even if the belief functions framework offers powerful tools for data fusion and uncertainty management, some points are still to be cleared.

The first thing concerns the building of belief functions as presented in section 4 and in [21,22]. Unfortunately, probabilities or statistics concerning the error rates of sensors are quite hard or even impossible to obtain. The building of sets of mass functions is also still obscure and may stay completely empirical, only confirmed by experimentation results. Our system will then require a lot of experimentation and is partly based on expertise on the Smart Home domain.

The conflict is something difficult to deal with. The mass of the empty set, often assimilated to the conflict, can include also the open-world assumption mass [15] and a part of auto-conflict [28] consequence of the fact that the Dempster's rule of combination is not idempotent. Even if some combination rules or algorithms offer a way to decrease the impact of conflicting sources [18,19,25], the identification of conflicting sources is still problematic.

The communication between the different layers of our architecture will require the definition of abstractions, e.g. "Need for certainty" or "Need for precision". We are currently working on that in order to exploit at their full potential the tools we presented in sections 3 and 4, especially the conditioning defined in [27] which seems to be very promising.

## 6 Conclusions

In this paper, we showed how it is possible to use the belief functions theory and its powerful tools to aggregate raw data and create a first level of abstraction for context modeling. These pieces of context are more understandable and easier to work within the context models, reducing the gap between the human activity and the information it is possible to sense in a Smart Home environment. The presented method is general and can be adapted to all type of context attribute as long as sensors which are able to bring evidence on context attributes exist. We think that the belief functions framework is promising because not only it enables to deal with uncertainty and to manage conflict but it also offers flexibility. The groups of sensors used to gather evidence can be dynamic and are not predefined. The belief model can be updated dynamically, e.g. if some problems are detected or to add new context attributes to the system, without modifying anything to the hardware.

We have demonstrated that an implementation is possible on embedded computers. Now, much work is still required to integrate this layer into the global architecture and to experiment the presented methods on real-life scenarios.

## References

1. Aldrich, F.: Smart homes: Past, present and future. In: Harper, R. (ed.) Inside the Smart Home, pp. 17–39. Springer, London (2003)
2. Kim, E., Helal, S., Cook, D.: Human activity recognition and pattern discovery. IEEE Pervasive Computing 9, 48–53 (2010)
3. Weiser, M.: Some computer science issues in ubiquitous computing. Commun. ACM 36, 75–84 (1993)
4. Coutaz, J., Crowley, J.L., Dobson, S., Garlan, D.: Context is key. Commun. ACM 48, 49–53 (2005)
5. Phidgets, http://www.phidgets.com/
6. Shelby, Z., Bormann, C.: 6LoWPAN: The Wireless Embedded Internet. John Wiley & Sons, Ltd., Chichester (2009)
7. Padovitz, A.: Context Management and Reasoning about Situations in Pervasive Computing. PhD thesis, Monash University, Australia (2006)
8. Goldman, R.P., Geib, C.W., Miller, C.A.: A New Model of Plan Recognition. Artificial Intelligence 64, 53–79 (1999)
9. Geib, C.W., Goldman, R.P.: A probabilistic plan recognition algorithm based on plan tree grammars. Artificial Intelligence 173(11), 1101–1132 (2009)
10. Dominici, M., Fréjus, M., Guibourdenche, J., Pietropaoli, B., Weis, F.: Towards a system architecture for recognizing domestic activity by leveraging a naturalistic human activity model. In: IRISA Internal Report 1977 (2011)

11. GuruPlug, http://www.globalscaletechnologies.com/t-guruplugdetails.aspx
12. Smets, P.: Theories of uncertainty. In: Press, I. (ed.) Handbook of Fuzzy Computation. IOS Press, Amsterdam (1998)
13. Dubois, D., Prade, H.: La problmatique scientifique du traitement de l'information. Rapport de recherche 02-08R, IRIT, Université Paul Sabatier, Toulouse (March 2002)
14. Shafer, G.: A Mathematical Theory of Evidence. Princeton University Press, Princeton (1976)
15. Smets, P., Kennes, R.: The transferable belief model. Artificial Intelligence 66(2), 191–234 (1994)
16. Yager, R.R.: On the dempster-shafer framework and new combination rules. Inf. Sci. 41, 93–137 (1987)
17. Dubois, D., Prade, H.: Representation and combination of uncertainty with belief functions and possibility measures. Computational Intelligence 4, 244–264 (1988)
18. Murphy, C.K.: Combining belief functions when evidence conflicts. Decision Support Systems 29, 1–9 (2000)
19. Chen, L.-Z., Sh, W.-K.i., Deng, Y., Zhu, Z.-F.: A new fusion approach based on distance of evidences. Journal of Zhejiang University Science 6A(5), 476–482 (2005)
20. Lefevre, E., Colot, O., Vannoorenberghe, P.: Belief function combination and conflict management. Information Fusion 3(2), 149–162 (2002)
21. Denoeux, T.: Constructing belief functions from sample data using multinomial confidence regions. International Journal of Approximate Reasoning 42(3), 228–252 (2006)
22. Aregui, A., Denoeux, T.: Constructing consonant belief functions from sample data using confidence sets of pignistic probabilities. Int. J. Approx. Reasoning 49, 575–594 (2008)
23. McKeever, S., Ye, J., Coyle, L., Dobson, S.: Using dempster-shafer theory of evidence for situation inference. In: Barnaghi, P., Moessner, K., Presser, M., Meissner, S. (eds.) EuroSSC 2009. LNCS, vol. 5741, pp. 149–162. Springer, Heidelberg (2009), http://dx.doi.org/10.1007/978-3-642-04471-7_12
24. Liao, J., Bi, Y., Nugent, C.: Activity recognition for smart homes using dempster-shafer theory of evidence based on a revised lattice structure. In: Proceedings of the 2010 Sixth International Conference on Intelligent Environments, IE 2010, pp. 46–51. IEEE Computer Society, Washington, DC (2010)
25. Ricquebourg, V., Delafosse, M., Delahoche, L., Marhic, B., Jolly-Desodt, A.M., Menga, D.: Fault Detection by Combining Redundant Sensors: a Conflict Approach Within the TBM Framework. In: COGIS 2007, Cognitive systems with Interactive Sensors. Stanford University, Stanford (2007)
26. Smets, P.: Decision making in the tbm: the necessity of the pignistic transformation. Int. J. Approx. Reasoning 38(2), 133–147 (2005)
27. Smets, P.: The transferrable belief model for quantified belief representation. In: Smets, P. (ed.) Handbook of Defeasible Reasoning and Uncertainty Management Systems. Quantified Representation Uncertainty and Imprecision, vol. 1, pp. 267–301. Kluwer Academic Publishers, Dordrecht (1998)
28. Martin, A., Osswald, C.: Generalized proportional conflict redistribution rule applied to sonar imagery and radar targets classification. CoRR abs/0806.2008 (2008)

# Dynamic Bayesian Networks for Sequential Quality of Experience Modelling and Measurement

Karan Mitra[1,2], Arkady Zaslavsky[1,2], and Christer Åhlund[2]

[1] Caulfield School of Information Technology, Monash University
900 Dandenong Road, Caulfield East, Victoria, Australia, 3145
[2] Luleå University of Technology
SE-971 87 Luleå, Sweden
{karan.mitra,arkady.zaslavsky,christer.ahlund}@ltu.se
http://karanmitra.me/research

**Abstract.** This paper presents a novel context-aware methodology for modelling and measuring user-perceived quality of experience (QoE) over time. In particular, we create a context-aware model for QoE modelling and measurement using dynamic Bayesian networks (DBN) and a context-aware state-space approach. The proposed model is then used to infer and determine users' QoE in a sequential manner. We performed experimentation to validate the proposed model. The results prove that it can efficiently model, reason and measure QoE of the users'.

**Keywords:** Algorithm, Bayesian network, dynamic Bayesian network, context-awareness, quality of experience, quality of service.

## 1 Introduction

Measuring user perceived quality of experience is a challenging task. This can be attributed to the fact that QoE not only depends on network parameters, but also on user's, device and the environmental context parameters [24,13,3,15]. These parameters together, affect users' expectations and their cognitive, behavioral and psychological states [24,22]. These states then dictate how they perceive the overall QoE. We argue that QoE of users' evolves over time i.e., based on previous interactions with the system, application or the service. By repeatedly using a service/object/tool, users' may get more accustomed and comfortable and measuring QoE at a single point in time and space will not yield correct measurements. Rather, their QoE needs to be assessed over a period of time (e.g., several days or several experimental runs) so that correct conclusions can be drawn.

Karapanos *et al.* [10] conducted a study concerning mobile phone usage. Their results show that users' perception of innovativeness inscreased during the first month and then remained stable. Also, users learnability was low for the first week and then increased sharply when users' got accustomed to their mobile

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 135–146, 2011.

devices. Perkis, Munkeby and Hillestad [19] conducted a 4 week study regarding QoE in 3G networks. Their results show that user expectations decreased after two weeks. They also show that the mean opinion score (MOS) regarding video application also decreased in the last two weeks from 3.4 to 3.1 which was measured on the scale of 1 to 5. These results strongly suggests that users' QoE varies with time. In this paper, we focus on QoE related to multimedia applications such as voice over IP (VoIP). We note that the current state-of-the-art (see, [14,24,21,15,3,2,5,23,9,11,4,17,19,22,12,13]) does not address the problem of QoE measurement of multimedia applications over time. Mostly experiments are conducted in the lab settings or living lab settings [4] and then conclusions regarding QoE are drawn. This is particularly true for research done on voice and video quality assessment domain [16]. There is also no provision in the current ITU-T and ETSI recommendations and standards such as the ITU-T G.107 [5] and ETSI STF 354 [2] to measure QoE over time.

Thus, in this paper, we consider the problem of measuring QoE over time. We believe this will help stakeholders such as network and codec engineers and application designers [3] in: 1.) understanding customers behavior over time and hence, helping them in correctly measuring customers QoE; 2.) providing customized and personalized services to customers in the form of user-centric SLAs and value-added services and applications; and 3.) creating opportunities for network/services/codec optimizations based on the feedback provided by the users in realistic settings, and by correctly inferring the root causes associated with their feedback. Unlike the state-of-the-art, we formulate QoE assessment problem as a sequential reasoning and inference problem where after each test, users' assess the QoE offered by the network operators by consciously or subconsciously bearing in mind the previous interactions.

We apply context spaces theory [18] to model users' QoE and dynamic Bayesian networks (DBN) [20] to determine the overall QoE of users'. Recently, Mitra *et. al* [12,13] developed a decision-theoretic approach for QoE modelling and measurement in mobile and pervasive computing environments. In this paper, we extend their approach to incorporate sequential QoE measurement. In all, we present the following contributions: 1.) we present a novel approach for QoE estimation as a context-aware sequential decision making problem which is not considered by practitioner's till-date. In particular, we formulate this problem as an inference problem in the light of QoE estimation and 2.) we extend a general context model [18] to include sequential data fusion and inference.

In section 2, we present the related work. In section 3, we present our methodology to measure QoE. In section 4, we present the results and a discussion for the same. Finally, section 5 presents the conclusion and future work.

## 2 Related Work

Sung [22], developed algorithms and architectures for QoE reasoning and adaptation. However, their approach was limited to mobile video applications. Wu *et al.* [24] define and classify QoE related parameters and try to find the correlation

between the QoE and QoS related classes. Their approach can be impractical when there are several QoE and QoS parameters as finding correlations between each and every parameter is a complex task. Furthermore, they do not consider the use of user and device related context parameters. Moller *et al.* [15], provides a taxonomy based on which application designers can select relevant QoE and QoS parameters to build multimedia applications. However, they do not present methods for modelling QoS and QoE relationships. Brooks and Hestnes [3,8] stress the need to consider a combination of subjective and objective methods to determine QoE. Mitra *et al.* [12,13], developed a context-aware, decision-theoretic approach for QoE measurement using Bayesian networks and utility theory. Their results show that they can achieve 94% accuracy for QoE estimation for two QoE classes. Karapanos *et al.* [10] suggests the need for finding causal relations between QoE parameters over time.

We gather that none of the techniques mentioned in the state-of-the-art address the problem of sequential QoE assessment. We state that a unified framework for sequential QoE modelling and measurement is required. Thus, in this paper, we develop and present a novel context-aware methodology for QoE modelling, reasoning and inference over time.

## 3 Context-Aware Quality of Experience Modelling and Measurement Using State-Space Approach and Dynamic Bayesian Networks

In this section, we present a methodology for sequential modelling, reasoning and measurement of QoE experienced by users'. Our model uses the cues collected from context sources such as sensors and network probes as observations and measures the QoE over time. We consider context spaces theory (CST) [18] for modelling context information and then extend it to DBNs to measure QoE. The main premise behind CST is the mapping of context information/attributes onto context states. These context states are fused to determine the overall situation occurring at a particular point in time. Formally, there are three important concepts in CST: context attributes defined as $a_i^t \in A$; context states defined as $S_i^t \in S$; and situation spaces defined as $R_i^t \in R$.

A *context attribute*, $a_i^t$, is the data that is used to infer a situation. For example, GPS coordinates ($a_{GPS}^t$), delay ($a_{delay}^t$), jitter ($a_{jitter}^t$) and packet loss ($a_{PL}^t$). A *context state* is the current state of the user or application at time $t$. It contains a number of context attributes to represent the state. A particular context state comprising of several context attributes is represented as $S_i^t = (a_i^t, \ldots, a_n^t)$. For example, consider a VoIP application with $a_{delay}^t = 20$ ms and $a_{PL}^t = 0\%$ packet loss. The resultant MOS calculated using the ITU-T E-Model [5] would be around 4. Thus, the context state, user satisfaction ($S_{US}^t$) is "very good" by assuming $S_{US}^t$ is measured on a 1 to 5 Likert scale. Where 1 represents "poor", 5 represents "excellent" and 2 and 3 represents "fair" and "good", respectively. If the application is used by the user at his/her home, another context attribute, location ($a_{loc}^t$), is added to infer the same context state. Finally,

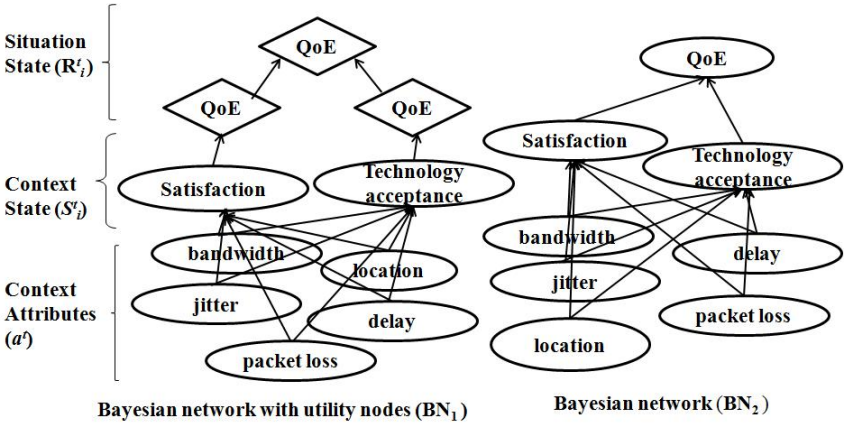



**Fig. 1.** BN with utility nodes ($BN_1$) converted to a simple BN ($BN_2$) to incorporate a new chance node, QoE. All the utility nodes are converted back to chance nodes by performing linear scale transformation and re-learning the model parameters.

a *situation space* represents a real-life situation. It is a collection of regions of attribute values corresponding to a predefined situation. It is denoted by a vector space $R_i = (a_1^R, \ldots, a_n^R)$ where an acceptable region, $a_n^R$, is defined by the set of variables $V$ which satisfies the predicate $P$ i.e., $a_n^R = \{V|P(V)\}$. Considering the aforementioned context states, the overall situation would be *"user is in his home and his QoE is very good"*.

Before defining a DBN for QoE measurement, we need to define a simple context-aware QoE model based on static BN and utility nodes. We refer to notations of [20] to define a BN as a directed acyclic graph (DAG) where random variables form the nodes of a network. The directed links between the nodes form the causal relationships. The direction of a link from X to Y means that X is the parent of Y. Any entry in the network can be calculated using the joint probability distribution (JPD) denoted as:

$$P(x_i, \ldots, x_n) \prod_{i=1}^{n} P(x_i|Parents(X_i)) \tag{1}$$

Where, Parents $(X_i)$ is the parent of $x_i$. Figure 1 shows our BN which is extended to include utility nodes to calculate the overall QoE. At the lowest level, context information such as bandwidth $(a^t_{bandwidth})$ and location $(a^t_{location})$ is collected from sensors and network probes. At the intermediate level, context states such as technology acceptance $(S^t_{TA})$ and user satisfaction $(S^t_{US})$ are inferred. These states are *hidden*. Finally, the top-most state is the situation state or the goal state which is inferred to determine the overall QoE situation of the user $(R^t_{QoE})$.

Based on [12], to infer $(R^t_{QoE})$, expected utility of each context state is calculated. Using multi-attribute utility theory, these utilities are added together as a weighted sum to determine a scalar value. This scalar values in then linearly transformed on the scale of 0 and 1 and is mapped on the bi-polar scale to

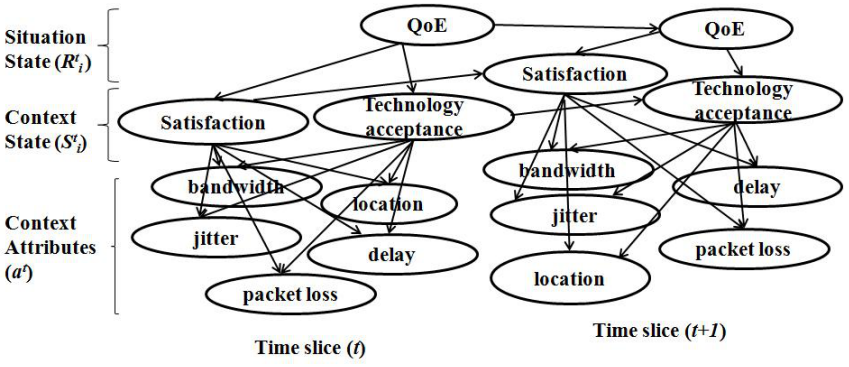



**Fig. 2.** DBN for QoE measurement and prediction. Two-time slices are assumed.

determined the QoE situation, $R^t_{QoE}$. Once it is determined, a new BN ($BN_2$) is created and it is added to the $BN_2$ as a chance node and the network is re-learnt using the previous context attributes, states and the new QoE values.

### 3.1 Sequential QoE Assessment and Measurement Using Probabilistic Inference

In the previous section, we use CST, BN and utility theory for QoE modelling, assessment and measurement. However, the proposed approach did not consider QoE evolving over time. Thus, in this section, we develop a new probabilistic model for assessing and measuring QoE over time. This results in capturing the behaviour and experiences of a user that also evolves over time and which cannot be measured at only one instance and within a limited scenarios. User interacts with their system on a daily basis and generates new ratings. The resulting data is used to improve the model with better modelling and prediction capabilities. Once a BN is created, as mentioned in the previous section, we can unroll it further to formulate a DBN which can incorporate temporal inter-dependencies [20].

In our case, the strategy is to find an approximate hypothesis ($h_n$) that best describes the QoE of users' at each experimental trail as we are dealing with partially observable environment and the user choices about QoE classes are not deterministic. To infer approximate QoE, we compute the marginals at each time step for the state sequences related to hidden variables such as $S^t_{TA}$, $S^t_{US}$ and $R^t_{QoE}$. In order to use the DBN, we unroll the $BN_2$ to several time slices as required by the stakeholders. This will include setting up of temporal dependencies (transition probabilities). In our case, we will use the data to learn the transition probabilities along with other parameters of the network. Figure 2 shows a simple DBN for QoE measurement over time (assuming two time slices). As can be seen in the figure, $BN_2$ is unrolled and is repeated along two time slices.

### 3.2 Model Description and a Context-Aware Algorithm for QoE Assessment Over time

In this paper, for our DBN, we assume a first order Markov process, i.e., current state depends only on the previous state. For example, $QoE_{t+1}$, $S_{US}^{t+1}$ and $S_{TA}^{t+1}$ are dependent only on previous states, i.e., $QoE_t$, $S_{US}^t$ and $S_{TA}^t$. Based on [20], We formally write it as a transition model defined as: $\mathrm{P}(S_t|S_{t-1})$ where $S_t$ is the current state of the system. In our case, we have three different hidden context state variables: $S_{TA}^t$, $S_{US}^t$ and $R_{QoE}^t$ which are inferred by the observations or the context attributes ($A_t$) such as bandwidth ($a_{bandwidth}^t$) and location ($a_{location}^t$). As shown in figure 2, the current observations or attributes are dependent on the current state variables and can be written as: $P(A_t|S_t)$. This is called as the sensor model. Additionally, prior or the initial probabilities of the states are defined. This is written as: $S_{US}^0, S_{US}^0$ and $QoE_{US}^0$. Using transition and sensor models and prior distribution, the complete joint distribution for the DBN can be written as:

$$P(S_0, ..., S_t, A_1, ..., A_t) = P(S_0) \prod_{i=1}^{t} P(S_i|S_{i-1}) P(A_i|S_i) \tag{2}$$

Using e.q.2, any query regarding QoE states and the context attributes can be answered. QoE measurement can be done in each state based on the inferring the probability of the hypothesis, $h_n \in S_t$. In particular, given the evidence or the context attributes ($A_t$), we calculate the marginals $P(S_t|A_t)$ for all the context and situation states. For inference in our DBN, exact and approximate algorithms can be used based on the space and time requirements [20]. In this paper we use an exact algorithm called Clustering [20] to infer all the hidden states.

In addition to inferencing, we must perform learning based on the data collected from the test subjects. In DBNs, the problem of learning can be divided into two parts, i.e., learning the sensor model parameters and learning transition probabilities (transition model) between the state variables. We consider the use of expectation maximization (EM) algorithm [20]. In order to learn the model, data can be collected from the users' then parameters can be learnt. In EM algorithm, there are two main steps: E-step which compute posteriors over the states and M-step which adjusts the model parameters maximize the likelihood of posteriors calculated in the E-step. The algorithm to calculate $R_{QoE}^t$ is given below. As we have the transformed $BN_2$ into a DBN, this algorithm calculates the QoE at each time step based on current and and previous states along with the context attributes. As mentioned previously, we assume that our model follows the $1^{st}$ order Markov process where only the current and only previous states are considered. It might happen that the state space is large or the $R_{QoE}^t$ needs to be measured in an online recursive manner. We use a sliding window (W) of a particular size to keep sufficient statistics for the model. For example, we can keep in history the states of past ten tests to infer the current QoE. After which, the latest observations and the states are added to the sliding

window and the model is re-adjusted. This algorithm provide enough power to accurately determine QoE at each time interval of infinite size while remaining within the Markov framework.

*A context-aware algorithm to learn and measure QoE sequentially.*

```
1. Initialize the model and the sliding window (W).
   (a) Using the collected data learn network parameters.
2. Project to the next time-step.
   (a) Generate new observations.
   (b) Based on the new observations and prior probabilities, infer
       hidden parameters (QoE states).
   (c) Record the observations and the states.
      i. Append it to latest observations in the sliding window.
3. Re-learn the network parameters from the new observations.
4. Go to Step 1 and repeat until final state.
5. Select the states with arg max (QoE) using the most likely
   state algorithm.
```

Once the new observations are generated at experimental trial, our model infers all QoE states for the current time-step by keeping in memory previous QoE states. The inferred values are then appended to the observation set and the W is re-adjusted. Then EM is run again to re-estimate the model parameters. This process repeats till the final state is reached. After parameter estimation, we have the smoothed estimates of the QoE for each time slice. To determine the most likely states, we use the most likely state algorithm [20]. This algorithm performs filtering on the all the states to find the most like state in linear-time.

## 4 Results Evaluation

### 4.1 Experimental Setup

For results evaluation, we considered QoE assessment related to VoIP application that can use both ITU-T G.729 narrow band (NB) and ITU-T G.722 wide band (WB) codecs. We considered two QoE metrics, user satisfaction (US) and technology acceptance (TA) represented as state variables, $S^t_{TA}$ and $S^t_{US}$ to measure the overall QoE ($R^t_{QoE}$). In this paper, US is represented as the mean opinion score (MOS) and is calculated on the scale of 1 to 5. Where, 1 means "poor" and 5 means "excellent". "fair", "good" and "very good" are represented as 2, 3 and 4, respectively. TA is classified as a behavioural factor [13] that determines whether the user will use the technology, application or the service again by consciously or sub-consciously observing the underlying QoS and environmental related context. TA is calculated as a boolean variable represented as either "yes" or "no" to represent "user will accept the technology" and "user will reject the technology", respectively.

**Table 1.** Selected context attributes and ranges

| context attributes ($a_n$) | values (ranges) |
|---|---|
| codec | ITU-T G.729, ITU-T G.722 |
| packet loss (%) | low (0% - 5%), medium (5% - 7%), high (7% - 10%) |
| delay (ms) | low (0 ms - 150 ms), medium (150 ms - 300 ms), high ( >300 ms) |
| location | home, tram stop, office |

**Experimental scenarios and the choice of data set:** We generated data simulating five users to be assessed over a period of six weeks. The choice of the number of users' is selected due to fact that in the living-labs settings such as [4], it is very difficult and expensive to collect user data over a period of several days and usually studies are conducted with a limited number of test subjects. We would also like to determine whether a small data set can be used to obtain satisfactory results. In our simulations, we considered a case where users' randomly give ratings based on the VoIP calls made at various times of the day. In all, we expect users' to make at least four calls daily (per user). Thus, every day, we expect 20 observations for all the test subjects. We used a non-overlapping time-window (W) of 1 week i.e., the overall QoE is predicted at the end of one week using 140 observations taken together. Indeed, stakeholders can adjust the size of W according to their needs. Based on W, we train the DBN for all the observations generated at every week for six weeks.

Based on these observations, we generated data for six weeks related to user satisfaction (US) and technology acceptance (TA) for both the codecs based on ITU-T recommendations [5,6]. Table 1 shows the values and ranges for several context parameters. We set TA according to US i.e., if US lies between 3-5, the user will most likely accept the technology, else he/she will reject it. It was shown in [19], that user tends to give a lower score at the end of four week period. Thus, we bias the calculated US using this intuition and reduce the US by 2 (from "excellent" to "good"), uniformly, over a period of four weeks. Based on this, TA is also reduced accordingly as shown in figure 3**.** We further manipulated the conditional probability tables (CPTs) to include the effects of location on the overall QoE based on the fact that users' social context changes at different locations [8]. For example, we expect a user to be more comfortable at home rather than his/her office due to stress or background noise which might affect his/her QoE. This is reflected by assigning higher probability to home followed by tram stop and the office to maximize $R^t_{QoE}$. Context states $S^t_{US}$ and $S^t_{TA}$ are inferred and then fused together to determine $R^t_{QoE}$.

### 4.2 Results Analysis

We developed a prototype using the GeNIe/SMILE package and APIs [1]. We used exact algorithm called Clustering [20] as the DBN inference algorithm. Figure 3 shows our DBN used for results evaluation. We consider finite horizon cases where we already know the finish time ($T_f$=6) of the assessment period

in advance (set by stakeholders). At each time-step ($t_s$), QoE is inferred by our algorithm. Using the simulated data, parameters of the DBN are learnt using the expectation maximization algorithm (EM) [7,20], as shown in our algorithm. Indeed, data generated at each experimental run can be based on any number of test cases. As can be seen in figure 3, QoE of users' decreases uniformly from $t_1$ to $t_3$ and remains same at $t_4$ while using ITU-T G.729 codec, and the model is able to correctly track the varying QoE. This validation was performed using the leave-one out cross-validation procedure to test whether the proposed model is able to correctly estimate the QoE values over time. We can notice that the proposed model can correctly determine the most likely state sequence for QoE from time $t_1$ to $t_4$.

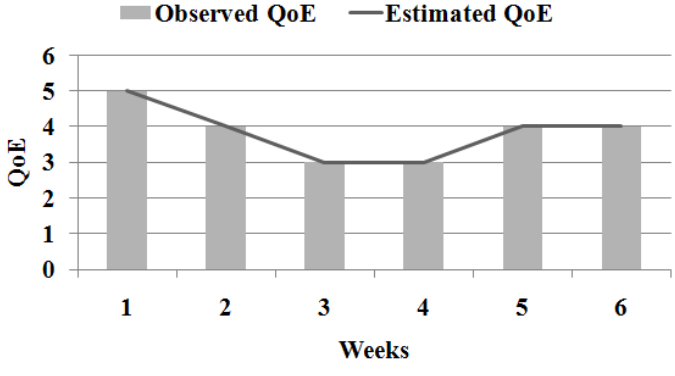



**Fig. 3.** Observed and estimated QoE values for six weeks. We can see that DBN is able to correctly match the observed QoE values.

To further validate the proposed model, we consider another case, where after assessing users' QoE over time, a smart VoIP application decides to change the codec proactively such that users' QoE is maximized. We simulate this case by changing the codec from NB to WB once the QoE of the user is predicted to fall below or remain constant for some time at a particular threshold (3 for MOS) at the end of four-week ($t_4$). It was shown in [16], that ITU-T G.722 codec gives higher QoE than NB codecs such as ITU-T G.711 and ITU-T G.729 codecs. Thus, we generated data based on this observation, for time $t_5$ to $t_6$ and use our algorithm to re-estimate by re-learning the model parameters. As can be seen in figure 3, QoE of users' increases from 3 to 4 after the codec change at time $t_5$ and then remains constant at $t_6$. Figure 3, shows the true and predicted estimates of the QoE values at each $t$. We can notice that the QoE decreases with time from week 1 to week 4 and it then starts to increase from week 5 due to the changes made by the smart application. We show that how our model is able to correctly match the true QoE estimates. We again performed leave-one-out cross-validation to test whether the proposed model is able to correctly estimate the $R^t_{QoE}$ values. From these results, we can gather that the proposed model is able to establish causal relationships between several QoE states and is able track the variations in QoE involving over time.

### 4.3 Discussion

Figure 3, shows the true and predicted estimates of the QoE values at each $t$ (for six weeks). It shows that the QoE decreases with time from $t_1$ to $t_4$ and it then starts to increase from $t_5$ due to the codec changes made by the smart application. We show how our model is able to correctly match the true QoE estimates using leave-one-out cross-validation based on three QoE states ($S^t_{TA}$, $S^t_{US}$ and $R^t_{QoE}$) and several context parameters $a_{PL}, a_{delay}, a_{location}$ and $a_{codec}$, over time. Our model is flexible as it allows the addition and deletion of QoE states and context attributes as and when required. It ensures that QoE states can be measured correctly under uncertainty using DBNs. This is achieved by fusing these states probabilistically based on current observations and prior estimates of the states. This helps to determine the overall QoE in a realistic settings based on the experiences of the users'. We are currently investigating the use of our QoE measurement model to incorporate several other QoE classes by involving more use case scenarios in our study to further validate the proposed model.

## 5 Conclusion and Future Work

In this paper, we present a novel context-aware methodology for quality of experience (QoE) modelling and measurement over time. The proposed methodology is based on a state space approach and dynamic Bayesian networks. It can incorporate several QoE classes and context parameters to accurately determine and predict QoE over time. We use the expectation maximization algorithm to deal with learning under uncertainty and missing user data. Our methodology is beneficial for network operators, codec engineers or application designers who are interested in measuring users' QoE in realistic settings. To the best of our knowledge, this is the first such attempt to measure QoE over time. In future, we will extend this methodology to incorporate sequential decision making in heterogeneous access networks.

## References

1. Genie software package, `http://genie.sis.pitt.edu/about.html` (access date: January 02, 2011)
2. ETSI STF 354. Guidelines and tutorials for improving the user experience of real-time communication services, `http://portal.etsi.org/stfs/STF_HomePages/STF354/Default.aspx?Selection=Home`
3. Brooks, P., Hestnes, B.: User measures of quality of experience: why being objective and quantitative is important. IEEE Network 24(2), 8–13 (2010)
4. De Moor, K., Ketyko, I., Joseph, W., Deryckere, T., De Marez, L., Martens, L., Verleye, G.: Proposed framework for evaluating quality of experience in a mobile, testbed-oriented living lab setting. Mob. Netw. Appl. 15, 378–391 (2010)
5. ITU-T Recommendation G.107. ITU-T Recommendation G.107, Methods for subjective determination of transmission quality (2008)

6. ITU-T Recommendation G.113. Appendix I: Provisional planning values for the equipment impairment factor Ie (1998)
7. Heckerman, D.: A tutorial on learning with bayesian networks. In: Proceedings of the NATO Advanced Study Institute on Learning in Graphical Models, pp. 301–354. Kluwer Academic Publishers, Norwell (1998)
8. Brooks, B., Hestnes, P., Heiestad: QoE (quality of experience) - measuring qoe for improving the usage of telecommunications services. Technical report (2006)
9. Jumisko-Pyykkö, S., Hannuksela, M.M.: Does context matter in quality evaluation of mobile television? In: Proceedings of the 10th International Conference on Human Computer Interaction with Mobile Devices and Services, MobileHCI 2008, pp. 63–72. ACM, New York (2008)
10. Karapanos, E., Zimmerman, J., Forlizzi, J., Martens, J.: Measuring the dynamics of remembered experience over time. Interacting with Computers 22(5), 328–335 (2010)
11. Liu, L., Zhou, W., Song, J.: The research of quality of experience evaluation method in pervasive computing environment. In: 1st International Symposium on Pervasive Computing and Applications, pp. 178–182 (2006)
12. Mitra, K., Åhlund, C., Zaslavsky, A.: A decision-theoretic approach for quality of experience measurement and prediction. In: Proceedings of the 2011 International Conference on Multimedia and Expo, ICME (2011)
13. Mitra, K., Zaslavsky, A., Åhlund, C.: A probabilistic context-aware approach for quality of experience measurement in pervasive systems. In: Proceedings of the 2011 ACM Symposium on Applied Computing, SAC (2011)
14. Moebs, S.A.: A learner, is a learner, is a user, is a customer: Qos-based experience-aware adaptation. In: MM 2008: Proceeding of the 16th ACM International Conference on Multimedia, pp. 1035–1038. ACM, New York (2008)
15. Moller, S., Engelbrecht, K.-P., Kuhnel, C., Wechsung, I., Weiss, B.: A taxonomy of quality of service and quality of experience of multimodal human-machine interaction. In: International Workshop on Quality of Multimedia Experience, QoMEx 2009, pp. 7–12 (29-31, 2009)
16. Moller, S., Waltermann, M., Lewcio, B., Kirschnick, N., Vidales, P.: Speech quality while roaming in next generation networks. In: IEEE International Conference on Communications, ICC 2009, pp. 1–5 (2009)
17. Nokia. Quality of Experience (QoE) of mobile services: can it be measured and improved? white paper (2006)
18. Padovitz, A., Loke, S.W., Zaslavsky, A., Burg, B., Bartolini, C.: An approach to data fusion for context awareness. Modeling and Using Context, 353–367
19. Perkis, A., Munkeby, S., Hillestad, O.I.: A model for measuring quality of experience. In: Proceedings of the 7th Nordic Signal Processing Symposium, NORSIG 2006, pp. 198–201 (2006)
20. Norvig, P., Russel, S.: Artificial Intelligence: A modern approach, 2nd edn. (2006)
21. Soldani, D.: Means and methods for collecting and analyzing qoe measurements in wireless networks. In: International Symposium on a World of Wireless, Mobile and Multimedia Networks, WoWMoM 2006, pp. 531–535 (2006)
22. Sung, J.: Football on Mobile Phones: Algorithms, Architectures and quality of experience in streaming video. PhD thesis, Umeå University (2006)

23. Key, W., Greger, W.: Determining utility functions for streaming low bitrate football video. In: Ninth International Conference on Internet and Multimedia Systems and Applications (2005)
24. Wu, W., Arefin, A., Rivas, R., Nahrstedt, K., Sheppard, R., Yang, Z.: Quality of experience in distributed interactive multimedia environments: toward a theoretical framework. In: MM 2009: Proceedings of the Seventeen ACM International Conference on Multimedia, pp. 481–490. ACM, New York (2009)

# Management of the Products on Display Shelves Using Advanced Electronic Tags Equipped with Ad Hoc Network Communication Capability

Kyohei Tanba, Daisuke Kasamatsu, and Kazumasa Takami

Graduate School of Engineering, Soka University,
1-236 Tangi-cho, Hachioji-shi, Tokyo, 192-8577, Japan
k_takami@t.soka.ac.jp

**Abstract.** Advanced IC tags equipped with ad hoc network communication capability are used for product location management to flexibly deal with frequent changes in product layout on display shelves as is often the case in rental shops. A tag is attached to each product. It manages product and location information. The paper proposes a method in which tags communicate with each other to updating product location information autonomously. It also proposes a method of managing and searching for product information (identification of the shelf on which a given product is located, and product attributes) held by each tag using a distributed hash table. The proposed methods have been implemented on a MANET emulator, and the performance of the methods, such as the probability of successful location information update, and search time, has been evaluated.

**Keywords:** IC tag, ad hoc network communication, distributed hash table, location management, MANET.

## 1 Introduction

As the ubiquitous society advances, there is a growing interest in a smart space that uses autonomous communication between mobile terminals attached to humans and objects embedded in the surroundings and communication between these objects in order to provide human-friendly services. Methods of managing data distributed in the ambient environment efficiently have been proposed [1][2]. IC tags, such as RFID (Radio Frequency IDentification) tags, are now commercially used for inventory management, inspection, and prevention of theft, of a variety of products and materials [3][4]. Furthermore, research and development of many service programs (hereafter called APLs) in Mobile Ad hoc NETwork (MANET) have been promoted [5]-[8].

There are already document management systems in operation in which a passive tag is attached in each document file, and multiple RFID tag readers are installed on each shelf in order to identify the shelf on which a given document file is located and to manage information about the file [9]. Such systems make it possible to trace who took out or returned what document file and when, but requires updating of location

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 147–159, 2011.

information each time the layout on shelves is changed. This can be extremely time-consuming in a place like a rental shop where the display layout of a large number of products is changed frequently.

In this paper, we use advanced IC tags equipped with an ad hoc network communication capability (hereafter referred to as "A_IC tags") to manage product location information in order to flexibly deal with frequent changes in product layout in a rental shop. A_IC tags are attached to products and shelves to manage product information. A method of updating product location information autonomously through coordination between A_IC tags is proposed. Also, a method in which a distributed hash table is used to manage and search for product information (identification of the shelf on which a given product is located, and product attributes) held by each tag. Section 2 describes the assumed usage, system configuration, and a proposed product management method using shelf tags. It also identifies issues for which solutions need to be found to implement the proposed methods. Section 3 proposes solutions to these issues. Section 4 describes the experimental system developed for the evaluation of the method. However, the debugging of APLs using a real MANET requires a considerable number of devices, manpower, and vast testing space. Furthermore, it is not possible for a simulator, such as ns-2, GloMoSim, or OPNET, to test a developed APL by executing it. For these reasons, an emulator is required as a test environment for APLs. We developed the proposed APLs on MANET emulator [10]. Section 5 presents the evaluation results, and Section 6 gives the conclusions and identifies an issue for future work.

## 2 Product Management at a Rental Shop Using Advanced IC Tags

### 2.1 Configuration of the Product Management System

Product management at a rental shop must be efficient and provide convenience for customers. It is convenient for customers if they can rapidly find the location of the desired product from among a large array of products. The shop needs to be able to identify the locations of its products even when the product display layout is changed frequently to suit the sales strategy of the shop. If the shop can manage and search for the locations of, products easily, it can operate efficiently and provide convenience for customers.

An example of the configuration of a management system that uses A_IC tags attached to products on shelves is shown in Fig. 1. An A_IC tag holds information about the location and attributes of the product to which it is attached. An ad hoc network is formed to interconnect A_IC tags. If this network is connected to a wide-area network, the return by a customer of a rented product at a return post can be detected by the A_IC tag installed to it, and a mail notifying the return can be sent to the customer and the shop. In addition, a service can be conceived in which A_IC tags are attached to mailboxes or dedicated return posts installed near train stations, and mailboxes of customers' residences, and return by a customer or shipping by the shop of a product can be notified to both the customer and the shop.

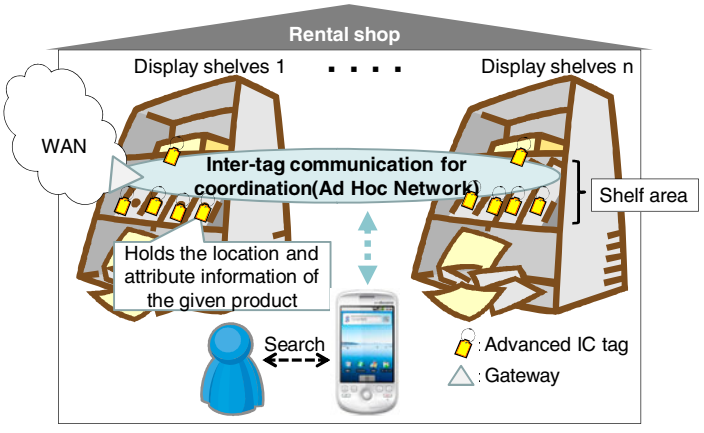



**Fig. 1.** Example of the configuration of a management system that uses A_IC tags attached to products on shelves

## 2.2 Issues with a Management System Using Shelf Tags

In a management system using shelf tags, a number of shelf tags are attached to each shelf, and each shelf tag broadcasts its area number to product tags. When a product tag detects more than a certain number of shelf tags having the same area number, it replaces the area number within its location information with the area number sent by theses shelf tags. An overview of this system is shown in Fig. 2. Each of Areas A and B has 4 shelf tags. The shelf tags of Area A transmit its area number ("Area A" in this case) to product tags. Whenever a product is moved to a new location, it is surrounded by a new set of shelf tags, and its product tag needs to identify the shelf tags of the area to which it has been moved.

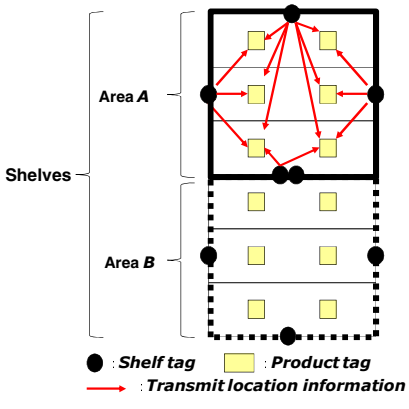



**Fig. 2.** Method of using shelf tags and transmission of location information

Issues that need to be addressed in implementing this system are as follows:

(1) Required functions of an A_IC tag, and information held by the tag

Currently, product management relies on bar codes, which hold information about only the product category and product name. This is not sufficient for A_IC tags that are to be used for product location management. It is necessary to identify the types of information the A_IC tag should hold. It is also necessary to define the required functions of the A_IC tag.

(2) Autonomous update of location information

To update product location information efficiently, it is necessary to develop an algorithm that enables the A_IC tag attached to each product obtain its location information autonomously.

(3) Method of searching for the location and attributes of a product

It is not efficient to register data in a centralized database each time product location information is updated as a result of a change in the display layout of a shop. It is necessary to develop a search method that is based on a distributed database for which no manual data registration is required. A mechanism is also need to autonomously notify the salesclerks of any stray product that happens to be placed on a wrong shelf or a wrong shelf area.

## 3 Solutions

### 3.1 Ad Hoc Communication and Tag Types

In a product management system using shelf tags, it is necessary that tags communicate with each other so that they can update their location information autonomously, and that the communication network is reconfigured autonomously when the product layout is changed. For this reason, MANET (Mobile ad hoc Network) is adopted for inter-tag communication. AODV (Ad hoc On-Demand Distance Vector) is adopted as the routing protocol [10]. An A_IC serves as a MANET node, and is assigned an IP address, which consists of the network part and the host part. The shelf tag is assumed to have sufficient memory capacity to hold the information of all product tags in one area.

Two types of A_IC tag are defined: product tags, which are attached to products, and shelf tags, which are attached to shelves. Tag type information is set to each tag. A product tag holds two types of information: location information (IP address, tag type, and area number) and product information (product category, product ID, product name, and numbered product name). *Product ID* is a unique identifier given to each product name. *Tag type* indicates whether the tag is a shelf tag or a product tag. *Area number* identifies the bay and the shelf. *Numbered product name* is used to distinguish between products of the same product name. The location of a product is identified by the area number of the area in which it is located. The information held by a shelf tag is IP address, tag type, area number, and the category of the product located in that area.

### 3.2 Algorithm of Updating Location Information

A product tag receives data not only from the shelf tags of the area it belongs to correct area but also from the shelf tags of other areas. It needs to detect the correct shelf tags. For example, if a product tag receives location information from the shelf tags of Area Number 1000 and also from the shelf tags of Area Number 1500, it cannot determine which of the two is correct.

To enable a product tag to detect the correct area, a group of shelf tags transmit the same area number. Shelf tags are classified into one parent tag, which represents the area, and child tags. Distinction between the parent and child tags is made in the code of a shelf tag. A child tag holds the IP address of the parent tag in addition to all information held by its parent tag.

Let $P$ be the number of shelf tags in one area, $A_h$ the host part of the IP address of the parent tag. Then, $A_h+1$, $A_h+2$, ・・・, $A_h+(P-1)$ are assigned to the host parts of the IP addresses of $P-1$ child tags. The association between the parent tag and child tags that share the same area number can be recognized by calculating the difference between the host parts of these tags. Suppose $P=4$, and $A_h=150$. Then, the group of shelf tags for an area consists of one parent tag and 3 child tags. The values assigned to the host parts of the addresses of the child tags are 151,152, and 153, respectively. Since the difference between the value of the parent tag and the values of the parent and child tags are 0, 1, 2, and 3, it can be determined that these shelf tags belong to the same group.

This algorithm can be described as follows:

Step 1: $A_h$, $A_h+1$, $A_h+2$, ・・・, $A_h+(P-1)$ are assigned to the host parts of the IP addresses of the parent tag, which has Area Number $E_N$, of $P-1$ child tags, and these assignments are broadcast.

Step 2: Determining the child tags that belong to the area number of the parent tag.

A product tag receives area numbers from $n$ parent tags and $m$ child tags. For each of the $n$ parent tags, it calculates the difference, $\Delta A_h(n, m)$, between the host part of the IP address of the parent tag and the host parts of the IP addresses of $m$ child tags, and determines the child tags that belong to the area number of the parent tag.

2-1: If $\Delta A_h(n, m)=|A_h(n) - A_h(m)| \leqq P-1$, then the product tag determine that these shelf tags are in the same area.

2-2: Otherwise, the product tag calculates differences for another parent tag.

Step 3: Determining the area number.

3-1: If there are $P_T$ or more shelf tags that belong to the same area number ($P_T \leqq P$), the product tag replaces its area number with Area Number $E_N$, which is the area number of these shelf tags.

3-2: If the number of shelf tags that belong to the same area number is less than $P_T$, then the product tag does not change its area number.

### 3.3 Database Using a Distributed Hash Table

We propose to configure the database using a distributed hash table (DHT) in order to manage distributed product data efficiently, ensure scalability, and disperse risk. Each

parent tag holds information about the product tags in its area. Products are managed through communication between parent tags. A parent tag has a hash value, H(k), with the product's category information, *k*, as the key. Search for a product is made by finding a match with respect to two attributes: the hash value and the product name. Figure 3 outlines relations between tags in the distributed database. The hash function is used to calculate a hash value from the character string of the product category name. It is expressed as:

$$H(k) = k \bmod c$$

(where *k* is the sum of the character codes of the product category name, and *c* is a constant)

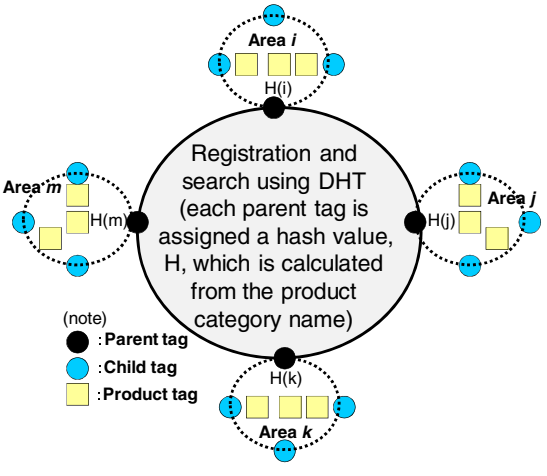



**Fig. 3.** Configuration of the database using the DHT of the product data

To enable DHT and the routing table of the MANET to be managed in an integrated manner, the hash value is set to the host part of the IP address of the parent tag. So, the host part of the IP address of the parent tag of Area *i* (this tag manages the product of category *i*) becomes $A_h$=H(i).

### 3.3.1 Algorithm for Registering Product Information in a Parent Tag

To enable prompt response to requests for search for product information, secondary memory added in a parent tag. It holds the product information held by all the product tags in the area. The algorithm used to register product information in the parent tag to aggregate product information is given below:

Step 1: When the location information updating algorithm described in Section 3.2 has completed the update of the area number, the IP address of the parent tag that has sent that area number is stored in the product tag concerned as supplementary information of the area number.

Step 2: The product tag sends the product information to the IP address of that parent tag.

Step 3: The parent tag holds the product information it has received.

### 3.3.2 Algorithm for Searching for a Product

The parent tag manages where a product is located by having the hash value derived from the category information of that product in the host part of its IP address. The distributed hash table held by the parent tag can be used as the routing table of the MANET. Search for a product is executed by having a parent tag send a search request to parent tags of other areas that are within the range that can be reached by its radio wave. The specific algorithm is as follows:

Step 1: The user enters the category information and product name of the desired product on the search terminal.

Step 2: Each parent tag calculates a hash value from the entered category information, and constructs an IP address with that hash value in its host part, adds the product name to it, and sends the IP address to adjacent parent tags.

Step 3: Each adjacent parent tag extracts the hash value from the IP address of the receiver, and checks it against its own hash value.

3-1: If they match, the parent tag searches the product tag information it holds for the product name, and returns the search result to the search terminal. If the parent tag finds several product names, it returns information about all of them.

3-2: If they do not match, the parent tag relays the received search request to its adjacent parent tag.

Step 4: The search terminal displays the product information it has received from the parent tag.

### 3.3.3 Updating of Parent Tag Information When the Display Layout is Changed

When the display layout is changed, many products may be moved to other shelves, and consequently, the category assigned to a shelf area may be changed to a different category. Therefore, the category information of the parent tag must also be changed. However, when a product of a wrong category is placed to a shelf area by mistake, the category information of the parent tag may be replaced by wrong category information. This can cause erroneous operations of the system. To prevent this from happening, we have introduced a mechanism that examines the category names of the product tags managed by the parent tag, selects one by majority, and replaces the category name of the parent tag with the selected one automatically. The specific algorithm is as follows:

Step 1: After the location information update algorithm described in Section 3.2 has been executed, each parent tag counts the number of product tags whose information it holds for each category.

Step 2: The parent tag replace its category name with the category name that has registered the highest count.

Step 3: The parent tag calculates a new hash value of the new category using the hash function.

Step 4: The parent tag constructs its IP address using the new hash value.

### 3.3.4 Determining Stray Products

To enhance the level of accuracy of product management, it is important that there is a mechanism that automatically detects any product that is located by mistake in an

area it does not belong to, and notifies the system administrator of it. If a parent tag detects a strange category name that numbers less than a certain value, $P_C$, it determines that there is a stray product from another area, and notifies the system administrator of it. If the number of the occurrences of the category name that seemingly does not belong to the area is a certain value, $P_C$, or greater, the parent tag concerned assumes that the display layout has been changed, and the processing goes back to the process described in Section 3.3.3. The algorithm for determining stray products is as follows:

Step 1: When a product tag receives category information from the parent tag during the location information update described in Section 3.2, it checks it against its own category name. If they are different, the product tag sends an "error" signal to the parent tag along with its own product information.

Step 2: The parent tag counts the number of times it has received an "error" signal.

2-1: If the error count is smaller than a certain value, $P_C$, the parent tag determines that a stray product(s) is in its area, and sends an alarm to the system administrator.

2-2: If the error count is $P_C$ or higher, it determines that the display layout has been changed, and does nothing.

# 4 Evaluation

## 4.1 Experimental System

An environment for executing ad hoc network emulator [11] and a product management system using shelf tags were implemented on a personal computer. Inter-tag communication was simulated by treating a virtual node as a virtual A_IC tag. A bay was modeled as shown in Fig. 4. Let *d* be the interval between adjacent products, *N* the number of products in one area, *S* the number of shelves, and *L* the length of a shelf. It was assumed that *L* is a fixed length of 120 cm, and that *N*=240 if *d*=1 cm, *N*=30 if *d*=10 cm, and *N*=15 if *d*=20 cm. It was further assumed that there were two areas on one bay and that the system can be scaled to 50 bays. The constant of the hash function was 255. For

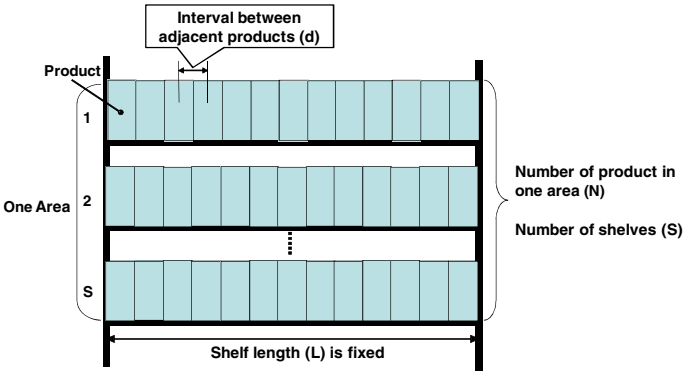



**Fig. 4.** Model of one area on a bay

example, if the category is k=ziburi, the hash function outputs a hash value of H(k)=151. The collision resolution strategy of the hash value was not implemented.

An overview of the processing of location information update in one area is described below using a window displayed by the emulator. The window is shown in Fig. 5. The update of the location information of product tags occurs when it receives location information from the four shelf tags in the same area ($P_T$=P). Shelf tags broadcast their area information at regular internals. Therefore, the update of the location information of product tags also occurs regularly.

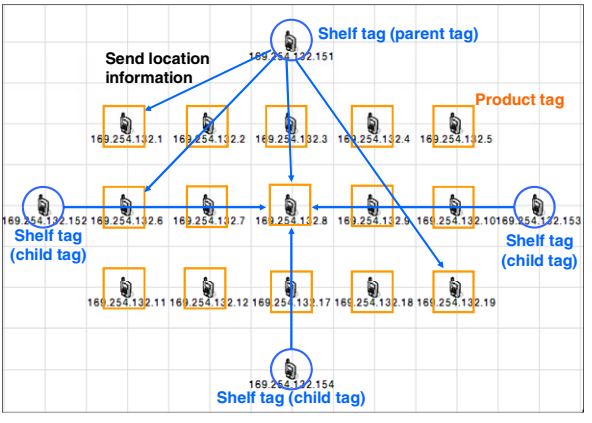
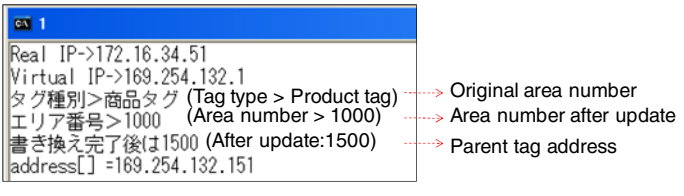
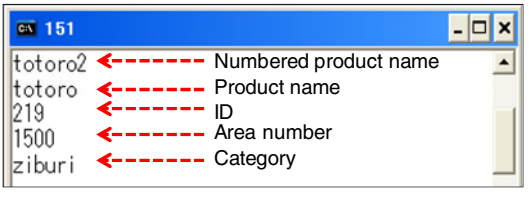



**Fig. 5.** Window displayed by the emulator and location information update (*S=3*, *d=20*, *N=15*, and *P=4*)




(a) Update of the area number

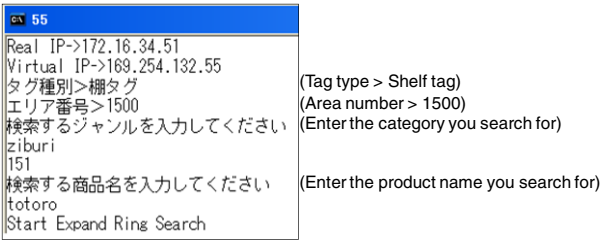



(b) Example of information registered in a parent tag

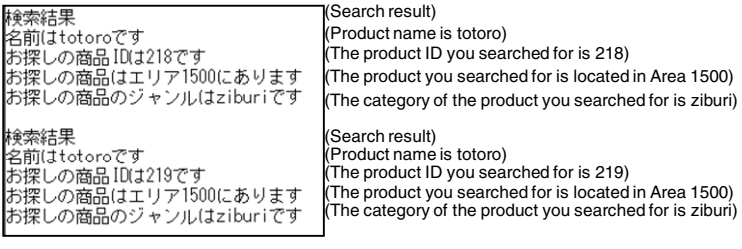
**Fig. 6.** Location information update and registered product information

Fig. 6(a) shows a case where the location information of one product tag (IP address =169.254.132.1) was updated. All the product tags whose location information has been updated send their product information to the parent tag. An example of product information sent to the parent tag is shown in Fig. 6(b).

Product search was made using a virtual node as the search terminal. The category name and product name of the desired product was entered in the search terminal. The entered information and a window that shows the search result are shown in Fig. 7.




(a) Input to the search terminal




(b) Search result output

**Fig. 7.** Example of search for product information

### 4.2 Response Time in Location Information Update and in Product Search

It is assumed that the number of products in one area, $N$, is 240 if the interval between the adjacent products, $d$, is 1 cm. Likewise, $N$=30 if $d$=10 cm, and $N$=15 if $d$=20 cm. The time it took from the transmission of information from shelf tags to the update of location information in product tags was measured. The result is shown in Fig. 8. The change in search time as the number of areas was increased up to 100 is shown in Fig. 9. When $d$=1 cm, and the number of products was increased up to 240, the response time stayed around 3 seconds. The search time includes the time it takes for parent tags to relay information to other parent tags and the time it takes for the target parent tag to return the search result. When the number of areas was increased, the search time increased proportionally to the number of areas. It was found that the search time in the case where the search involved relaying of information through all areas is 4 times as long as the search time in the case where information was relayed only once.

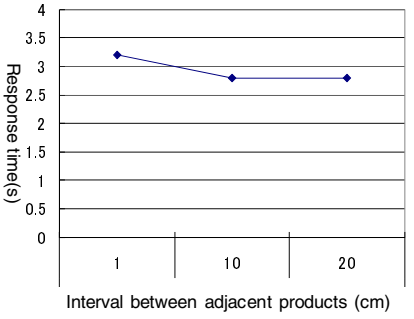



**Fig. 8.** Change in response time in location information update as the number of products increases

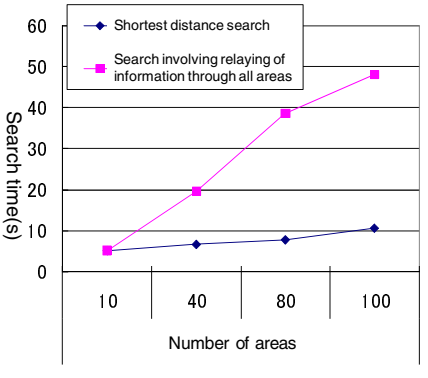



**Fig. 9.** Change in search time as the number of areas increases

### 4.3 Probability of Successful Location Information Update

We evaluated the probability of successful location information update using the shop model. It consisted of 3 bays with 6 areas, and $d$=20 cm, $N$=15, $L$=120 cm, and $S$=3. We measured the probability of successful update as we changed the range reached by the radio wave from a product tag. By success it is meant that the area number of a product tag is replaced by the correct area number, and the correct parent tag is recognized. The range reached by the radio wave from a shelf tag is set to 120 cm, which allows the tag's own area to be covered with as little interference on products in other areas as possible. The range reached by the radio wave from a parent tag is set to140 cm, which is the minimum distance at which a parent tag can communicate with the parent tags in the immediate right and left areas. The measurement result is shown in Fig. 10. The probability of successful update stays around 20% almost irrespective of the range the radio wave from a product tag can reach. A possible reason for the failure of 20% is that a product located at the edge of an area happened to be within the range reached by the radio wave of a shelf tag of a neighbor area and replaced its location information with that of the neighboring area.

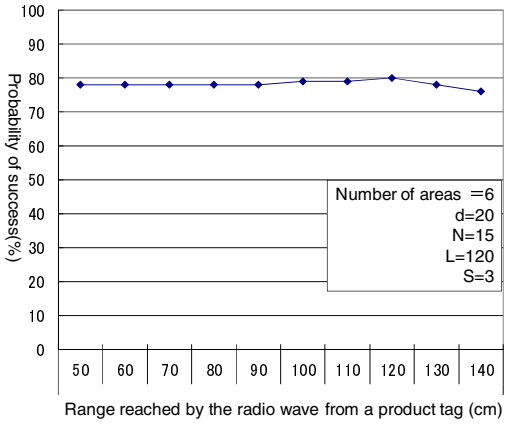



**Fig. 10.** Probability of Successful Location Information Update

## 5 Conclusions and Future Issues

This paper has proposed a product management system that can flexibly manage the location of products even when the product display layout is changed frequently. The system use advanced IC tags for shelf tags. The paper proposed specific algorithms: algorithm for updating location information though inter-tag communication, and algorithm for registering and searching for product information using a distributed hash table. An experimental system was developed using a MANET emulator and assuming virtual ad hoc nodes as advanced IC tags. The proposed algorithms were evaluated through operational experiments, such as update of location information, and search and registration of product information. The performance of the proposed system was checked by measuring the response time in location information update and the response time in search for product information, as the number of products and the number of areas were changed. The probability of successful location information update was measured, as the number of areas was changed. The probability was found to be around 80%.

One of the issues that need to be addressed in the future is to study ways to raise the probability of successful location information update.

## References

1. Murata, M.: Towards Establishing Ambient Network Environment. IEICE Trans. Commun. E92-B(4) (April 2009)
2. Stoica, L., Morris, R., Karger, D., Kaashoek, M.F., Balakrishnan, H.: Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications. In: Proc. of the ACM SIGCOMM 2001 (August 2001)
3. Hori, T., Wda, T., Ota, Y., Uchitomi, N., Mutsuura, K., Okada, H.: A Multi-Sensing-Range Method for Position Estimation of Passive RFID Tags. WiMob, 208–213 (October 12-14, 2008)

4. Shahbazi, S., Harwood, A., Karunasekera, S.: Achieving Ubiquitous Network Connectivity Using an RFID Tag-Based Routing Protocol. In: 14th IEEE International Conference on Parallel and Distributed Systems (December 2008)
5. Horng, G.-J., Horng, C.-F., Jong, G.-J.: Mobile RFID of Wireless Mesh Network for Intelligent Safety Care System. In: World Congress on Engineering and Computer Science 2007 (WCECS 2007), San Francisco, US (2007)
6. Sezaki, K., Konomi, S., Suzuki, R., Furusawa, T.: Demo Abstract: An RFID-Based Positioning System for Ad-Hoc Networks. In: MobiHoc 2008 Demo (2008)
7. Lee, E.-K., Yang, S., Oh, S.Y., Gerla, M.: RF-GPS: RFID assisted localization in VANETs. In: IEEE MASS Workshop on Intelligent Vehicular Networks, Macau, P.R.C. (October 2009)
8. Jo, M., Youn, H.Y., Cha, S.-H., Choo, H.: Mobile RFID Tag Detection Influence Factors and Prediction of Tag Detectability. JSEN 9(2), 112–119 (2009)
9. (March 2011), http://www.itoki.jp/security/re/pdf/ssb09.pdf
10. Kasamatsu, D., Kawada, K., Okada, D., Shinomiya, N., Ohta, T.: An Emulator for Testing Service Programs in Ad Hoc Networks. The IEICE Transactions on Communications J92-B(4) (2009)
11. Perkins, C., et al.: Ad hoc On-Demand Distance Vector (AODV) Routing: RFC 3561 (March 2011), http://www.ietf.org/rfc/rfc3561.txt

# Customized Check-in Procedures

Dmitry Namiot[1] and Manfred Sneps-Sneppe[2]

[1] Lomonosov Moscow State University,
Faculty of Computational Mathematics and Cybernetics,
Moscow, Russia
dnamiot@gmail.com
[2] Ventspils University College
Ventspils International Radioastronomy Centre,
Ventspils, Latvia
manfreds.sneps@gmail.com

**Abstract.** This paper describes a new mobile service: customized check-ins. This service lets any business provide a customized form for mobile users that allow them mark (describe) business-related events in the social networks. So instead of the traditional check-ins models, introduced by the communication services, this service introduces business-oriented check-ins. For the business this service introduces a new way for advertising in the social networks. And this advertising model does not require any intermediate service. For the consumers this service introduces a way for exchanging access to the own social graph for some benefits (e.g. gifts, discounts, coupons etc.).

**Keywords:** – checkin, mobile, HTML5, QR-code, coupon, Facebook.

## 1 Introduction

At this moment one of the popular models for the modern internet services (especially – for geo services) is “check-in”. Check-in is a presence status, associated with some particular location. For example, in Foursquare users "check-in" at venues using a mobile website, text messaging or a device-specific application by running the application and selecting from a list of venues that the application locates nearby. [1] Each check-in awards the user points and sometimes "badges". [2]

Points are awarded for "checking in" at venues. Users can choose to have their check-ins posted on their accounts on Twitter, Facebook or both. Users can also earn badges by checking in at locations with certain tags, for check-in frequency or for other patterns such as time of check-in. Foursquare was created with a core set of 16 badges, designed to reward and motivate all of users. The company has stated that users will be able to add their own custom badges to the site in the future.

So earned points (badges) are finally the things users can exchange with some benefits at venue (e.g. discount, free offering, gift etc.). Of course, it is true only if/when business that owns this venue participates in Foursquare’s programs. Let us just highlight an important note to our future discussion – badges are being developed by the communication service, not by the business. And any customization here is actually an agreement between the business and communication services. The

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 160–164, 2011.

business (the source for rewards) is actually not free in the forms these rewards will be presented for business users.

The similar model is actually reproduced by the various implementations of "Places" services: Twitter places, Facebook places etc. E.g. as per official page, Facebook places, for example, lets you easily share where you are, what you're doing and the friends you're with right from your mobile. You can check in and your updates will appear on the Place page, your friends' News Feed and your Wall. [3] The business model is similar. You can check in to get individual discounts, share savings with friends, earn rewards for repeat visits or secure donations for good causes.

But here is at least one serious remark that remains true for all these services. All they are communication services at the first hand. And any check-in at the first hand solves communication tasks: how to let my friends/followers know where I am. The biggest question that remains is very simple and natural. Why do we ask business to deliver benefits via advertising some 3-rd party service? It looks very natural to let business define the format that should be used (exchanged) for benefits. It is actually the main idea our QRpon [4] service was born from.

## 2 The Model

Let us start from the business side. Here QRpon offers a specialized CMS (content management system) that lets any business create a special mobile web site. This web site lets users automatically, just after confirming the identity, post business-defined information on the Facebook's wall. In the exchange for this posting (action) mobile web site will show a confirmation for the benefits. E.g. coupon, discount info etc. In other words – anything that could be presented to the staff on the business side for claiming the benefits. How to present this mobile site for the potential users? It is where QR-codes help us [5]. CMS lets businesses create mobile web site and an appropriate QR-code. Because it is mobile web (HTML5) application there is no need for downloading. Just scan QR-code and get URL opened.

Automatically, this approach obviously supports also physical check-ins. There is no way to mark you "at this location" being actually nearby (based on GPS location) or even far away (via API). QR-code should be scanned, and it is a physical action that could be performed on-site only.

So for the business this approach offered a mobile web site (sites – business can create more than one site, update them often etc.), presented on-site with QR-code sticker, that lets visitors exchange posting in the social network (e.g. Facebook's wall) for some benefits. And all site's aspects (what is presented on the site, what should be posted to the social network, what should be presented as a confirmation) are defined by the businesses themselves. Another possible explanation – try to think about the current check-in system (e.g. Foursquare) and just replace the standard posting (notice) from Foursquare with your own text. Obviously your potential users do not need to download (install) mobile application and do not need to register in some new service (beyond their Facebook accounts). And another important difference from Foursquare (Facebook, Twitter etc.) check-ins is the need for the physical presence.

The mobile CMS mentioned here is really simple. Practically, the business just needs to provide 3 pieces of text: the description (text for the first page of future mobile site), the text that should be posted to social network and the text that should be displayed on the mobile site after the posting. So the mobile site itself has got just two pages: the offer and the result (coupon, gift/discount info etc.). And the transition from the first page to the second pages posts data to social network.

CMS creates a mobile site as well as the QR-code for the link to that site. This QR-code could be placed anywhere on the business side. So for access to the benefits (coupons, discounts) potential users need scan it with own mobile phone. QR code usage is very natural here. As per Google, the mobile phone acts as a cursor to connect the digital and physical worlds [6].

**Fig. 1.** QR-code for mobile site

As soon as QR-code is scanned it is just one click deal to open mobile URL. The user will see the first page, created by QRpon CMS (offer). After that user can confirm (accept) this offer, using his/her Facebook ID. Mobile site uses Facebook Connect [7] here.

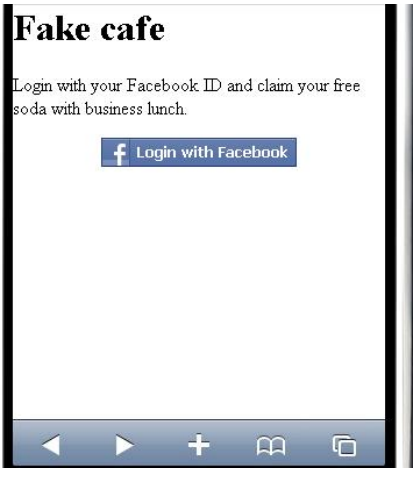



**Fig. 2.** Initial screen

As soon user's identity is confirmed, the status text (also defined previously in QRpon CMS) is posted to the user's wall. And as a confirmation for that, mobile web site shows its second (final) page. It is a confirmation for getting benefits (coupon, discount info etc). It could be presented to the staff at the business side etc. The following figure illustrates this (the same text is already posted to user's wall in Facebook):

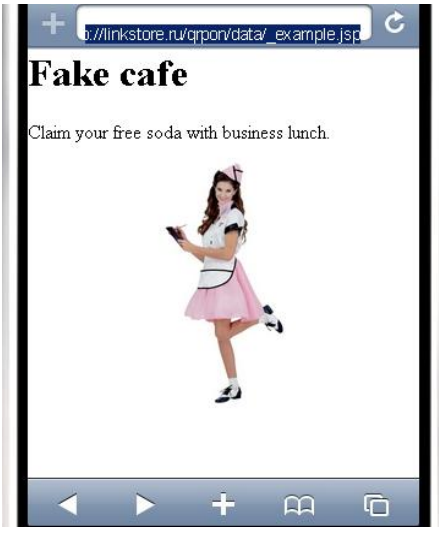



**Fig. 3.** Confirmation screen

So all the steps in this process are:

a) completely defined by the business
b) do not use any intermediate site / service

It is a fully customized check-in process (or completely customized, user defined badges in the terms of exiting communication services).

What could be added here in the future versions (or as some fermium service)? This service at the moment of posting data to the social network has got access to the user's social graph. It means, particularly, that we can program output (our confirmation page) depending on the social graph size for example. E.g. the more friends our customer has, the more potential readers we will see our posting. So the benefits could be increased for example etc.

In other words, the confirmation screen generation might be actually some production (rule based) system. It could be a set of rules like this:

*IF* (some condition) *THEN* (some conclusion)

Where conditional part includes a set of logical operations against user's social graph data and conclusion is our output (coupon, gift confirmation etc.). So we are

going to say here, that our system could be actually some sort of expert systems (production based) that generates conclusions (badges) by the social graph defined conditions. And our store for rules will present if the future versions some implementation of well known RETE algorithm [8].

Obviously, that the proposed approach lets us accumulate an interesting statistics for the business. Facebook API has got a TOS (terms of services), we can not simply log raw data, but even the accumulated info could be very interesting. Just because our application gets access to the basic social graph data at the moment our user accepts an offer we can accumulate for example sex-age histogram for our buyers etc.

In the same time, TOS for social networks API let us keep ID's for users. Just for keeping that info (without any data) we can easily discover new and returning user and easily implement such feature as "Majors" - users with the most visits, or users with the most visits within the given interval. In other words all the functionality (related to the business delivery) check-ins in the modern communication services could be provided directly. We simply do not need communication service (e.g. Foursquare) for that.

## 3 Conclusion

This paper describes a new model for customized check-ins. This model lets businesses provide own forms for badges exchangeable for some benefits (e.g. discounts, gifts, coupons) without the external communication services. It is actually a new approach for advertising in the social networks. QRpon model presents mobile service that lets any business exchange some benefits for the clients with posting (advertising) in the social networks.

This service does not require downloadable mobile applications and based entirely on the mobile web (HTML5). Via extensively used QR-codes this service builds a bridge between the virtual world of social networks and traditional bricks and mortar businesses.

## References

1. Washington Post: SXSW: Foursquare Scores Despite Its Flaws, http://www.washingtonpost.com/wp-dyn/content/article/2009/03/18/AR2009031802819.html
2. http://en.wikipedia.org/wiki/Foursquare_%28social_network%
3. Facebook Places, http://www.facebook.com/places/
4. QRpon, http://qrpon.linkstore.ru
5. QR-code, http://en.wikipedia.org/wiki/QRcode
6. http://mashable.com/2011/03/11/mayer-sxsw-talk/
7. Facebook Connect, http://developers.facebook.com/docs/guides/web/
8. Forgy, C.: On the efficient implementation of production systems. Ph.D. Thesis, Carnegie-Mellon University (1979)

# Architecture and Comparison of Two Different User-Centric NFC-Enabled Event Ticketing Approaches

Serge Chaumette[1], Damien Dubernet[1], Jonathan Ouoba[1],
Erkki Siira[2], and Tuomo Tuikka[2]

[1] LaBRI, University of Bordeaux, France
{serge.chaumette,damien.dubernet,jonathan.ouoba}@labri.fr
[2] VTT, Finland
{erkki.siira,tuomo.tuikka}@vtt.fi

**Abstract.** This paper describes two different approaches to use Near Field Communication (NFC) enabled mobile phones in a ticketing system dedicated to event management: *(i)* a offline version where some equipments are not connected to the Internet ; *(ii)* a online version where an Internet connection is available on all the equipments composing the infrastructure. These two propositions are compared so as to evaluate their pros and cons in terms of user experience, security, economical aspects, reliability and speed of use. We also identified a scenario with six use cases and decided to focus on ticket issuance and ticket presentation.

**Keywords:** Events, E-ticketing, NFC, Mobile Phone, Secure Element.

## 1 Introduction

### 1.1 Overview

The use of electronic tickets (e-tickets) has been significantly growing in the past few years. It still remains that a vast majority of the existing e-ticketing systems are proprietary solutions, primarily designed for the transportation industry and thus cannot be used in other domains.

For instance e-tickets can be helpful in the event ticketing industry which is a multi-billion dollar business. For example in North America gross concert revenue was 4.2 billion USD in 2008 and movie ticketing in 2009 was worth over 29.9 billion USD worldwide[1][2]. Ticketing has gone electronic in some stages of the chain, but the tickets themselves are still physical and these physical tickets have to get somehow to the buyer and also allow easy validation process. One solution is Near Field Communication (NFC)[3].

### 1.2 NFC Technology

NFC is an emerging technology that takes its roots in Radio Frequency Identification (RFID). It is a wireless communication technology that has a range of about 10 centimetres. One of the most important drive for the NFC has been the mobile phone industry

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 165–177, 2011.

where many notable manufacturers are integrating it within their devices (or they at least claimed that the NFC will be part of their future phones[4]).

NFC offers three modes of operation: reader/writer, peer-to-peer and card emulation. The reader/writer mode makes it possible for NFC devices to interact with passive NFC tags. The peer-to-peer mode supports direct communication between NFC devices, and the card emulation mode allows a NFC device to act as if it were a smart card. NFC devices offer support for an embedded smart card chip that is called a secure element. This secure element is connected to the NFC chip by the so called Single Wire Protocol (SWP)[5]. This secure element can be a (U)SIM card[6] or an integrated chip[7]. In card emulation mode, NFC devices do not create their own RF field but are powered by the electromagnetic field of an external device. The supported smart card types are MIFARE ISO/IEC 14443 Type A and Type B, FeliCa and ISO 15693.

The first implementations of NFC based ticketing appeared in public transportation systems as there is an existing infrastructure for smart card based ticketing that NFC-based ticketing can integrate. For example RMV, the local Frankfurt (Germany) public transportation company, has implemented a NFC transport ticketing pilot[8]. Doubtless, the NFC technology can also bring many advantages for e-ticketing in the domain of event management. Nevertheless, in the current mobile phone environment, there is a lack of standard propositions in this domain of event e-ticketing that would rely on the use of NFC-enabled mobile phones. As a starting point, we leaned on Suikkanen and Reddmann work[9] in which they have identified two basic approaches for NFC-ticket validation: offline and online ticketing. In the proposed paradigms, the validation is done either locally (offline) or through and Internet connection (online).

### 1.3 Contributions

The NFC Forum, as a consortium of different stakeholders in the field of NFC technology, believes that the cost of providing event ticketing, in terms of card issuance and management, can be driven down by using NFC-based systems[10]. Consequently, there is a strong need of moving the e-ticketing system for events to the NFC-enabled mobile devices field[11]. There is also a need to compare between the solutions and the context in which they can/cannot be used. Therefore, we propose a description of two solutions derived from the offline and online paradigms, and evaluate them in terms of security, reliability, speed of use, user experience and economical aspect. We present a six-phases event ticketing scenario and focus on the tickets issuance and the tickets validation processes which are key points in most e-ticketing architectures. For simplicity concerns, the terms 'ticket' and 'e-ticket' are used interchangeably in the rest of this paper.

## 2 Existing E-Ticketing Solutions

### 2.1 Overview

There are numerous initiatives and many companies that work in the field of e-ticketing. Most of the proposed solutions target transportation systems and are not necessarily

adapted to events. Moreover, some of the solutions adapted to mobile phones use the 2-D bar code system (and not NFC), which has disadvantages in some situations[9]. One interesting example in e-ticketing is the Cityzi[12] initiative in Nice (France) that uses NFC-enabled mobile phones but that provides no solution for the management of event tickets. Nevertheless, there were some trials to combine e-tickets and NFC but no standard solution for mobile devices appeared. We present here a small sample of some projects that we believe are representative of what is deployed today.

### 2.2 The Domain of Transportation

In the airline industry, electronic tickets have long replaced paper tickets. Some airline companies also issue electronic boarding passes which are sent to the mobile phone of the passenger or printed out by the passenger himself. The Air France online check-in option[13] is an example of this system where the user receives his identifier via SMS, MMS or email as a 2-D bar code.

Public transport operators within cities are also very keen on deploying e-ticketing systems. The Oyster card in London[14] and the Yikatong in Beijing[15] are transportation cards based on the *MIFARE*[16] technology. In Asia, the Octopus Card[17] in Hong Kong is based on the *FeliCa* standard[18] which is a contactless card technology.

Using NFC technology solutions, Ghiron, Sposato, Medaglia and Moroni developed and tested a *Virtual Ticketing application* prototype for transport in Rome where the virtual tickets are stored in a secure element embedded within a mobile equipment[19]. This user-oriented offline implementation showed that NFC could improve usability of e-ticketing systems.

The French transportation company *Ligne d´Azur* in collaboration with Cityzi in Nice also provides a mobile phone based application to buy tickets and validate them using NFC[20].

### 2.3 The Domain of Events Management

Digitick, which is an online event ticketing company, offers as Air France does, a 2-D bar code system. The mobile phone users can buy their tickets through website and then download them as images representing the corresponding 2-D codes[21]. At the event entrance they present the code which must be displayed on the phone screen for validation purpose. In this online oriented solution, no NFC is involved.

Another existing solution is the Tapango system[22] which is an electronic voucher system based on NFC cards as e-Wallets. The system reduces the use of paper tickets and was implemented by the Artesis´ research lab. With Tapango, the users first buy tickets via a webinterface, then at the event location they need to synchronize their e-Wallet (by means of a machine connected to the Internet) to 'physically' acquire the tickets and finally they present the NFC card at the entrance to get access. The use of NFC-enabled mobile devices is presented as a step to come in the evolution of the system.

In the SmartTouch project[23], a pilot related to event ticketing in the theatre of the city of Oulu (Finland) was deployed[24]. The users were able to receive tickets on their

NFC-enabled phones and the control of the tickets was achieved with another NFC-enabled mobile phone. Despite the fact that the ticket validation was relatively slow (using the peer-to-peer NFC mode), the users showed a real interest.

## 3 Event Ticketing Concepts

### 3.1 Scenario and Use Cases

Thereafter is reminded that for simplicity concerns, the terms 'ticket' and 'e-ticket' are used interchangeably throughout this paper. Event ticketing follows a well-known scenario. If we want to attend an event, we first need to choose a type of event and a specific venue. After the choice is done, we gather all the available information on the selected show before deciding rather to buy tickets or not. In some special cases we can directly receive advertisements about events related to our hobbies. Once we have made a decision, we find a date and a time that suit us the best and we then go to the shop (which can either be virtual or physical) where we can buy the approriate number of tickets (most of the time the possibility is given to buy tickets for friends). The day of the show, we go to the venue and at the entrance we present the ticket which corresponds to the event. Six use cases can be identified from the scenario: selection of event, event description visualization, reception of alerts, event tickets issuance, event tickets exchange and ticket presentation at the event site.

In the 'selection of events' use case, the user browses through different categories (for example theater, cinema, concert, etc.) to find the shows he could be interested in. He queries to look for available tickets in the selected categories. Finally, he receives the propositions which correspond best to his choices. For the 'visualization of event' description use case, the user simply visualizes a multimedia presentation of a specific event on his mobile phone before deciding rather to buy tickets or not. Concerning the 'reception of alerts', the user registers to receive alerts as soon as information is available for the kind of events he is interested in. In the 'tickets issuance' use case, the user selects one or several tickets for an event, chooses a payment option, enters

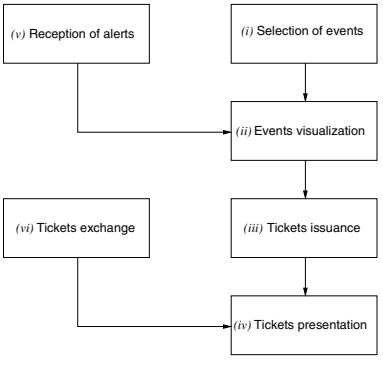



**Fig. 1.** Event ticketing scenario with use cases

the necessary information and validates the transaction. The tickets are then issued and pushed to his mobile phone. The 'exchange of tickets' gives the user the possibility to transfer tickets for instance to a friend by sending them to his mobile phone. Finally, for the 'ticket presentation' at the event site, the user shows his mobile phone to be granted access. These six cases (figure 1) represent the scenario or the steps to follow to attend an event using a mobile phone. This paper focuses on the description of *event tickets issuance* and *ticket presentation at event entrance* steps.

### 3.2 E-Tickets for Events

To properly define the e-tickets in the event ticketing context, we must consider the architecture which is commonly used in e-ticketing systems. To be precise, a e-ticketing system can be seen as a token-based authentication platform that involves 3 main entities : an Issuer, a User and a Verifier[25]. The e-ticket represents the token which circulates between the different entities. Figure 2 briefly explains the role of each entity of the system. An event e-ticket gathers various pieces of information (ticket ID, event ID, price, seat number, etc.) for a particular event[26]. It contains at least the information that can be found on regular paper tickets. Event e-tickets can also contain cryptographic data such as checksums or digital signatures from ticket issuers so that the integrity and authenticity of the tickets can be verified/guaranteed (figure 3).

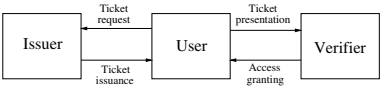



**Fig. 2.** Common e-ticketing architecture

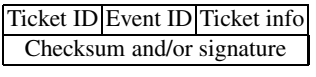



**Fig. 3.** E-ticket data model

## 4 Two Different Approaches: Offline vs. Online

### 4.1 Offline System

**Use Case.** Adam wants to go to see a concert downtown. He connects to the concert website using a computer or his mobile handset and finds a link to the ticket issuer's website. He enters all the necessary details such as his mobile phone number or his payment option and finalizes the purchase. He receives the ticket in a digital format which is sent over the mobile operator network to his mobile phone. When Adam arrives at the concert location, he taps his phone on the reader at the entrance. The ticket is transferred to the ticket verifier which authorizes (or not) the access. The light turns green and Adam can enter to find his seat.

**Architecture and Interactions.** Four main entities (figure 4) are involved in the offline ticketing approach: the ticket issuer, the ticket verifier, the user with its NFC-enabled mobile phone and the Trusted Service Manager (TSM). In our context the TSM is the entity which manages the loading, the deletion and the personalization of data on the secure element of a mobile phone through a mobile operator network[27].

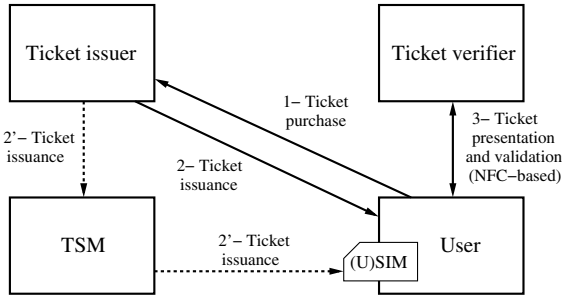



**Fig. 4.** Offline System Architecture

The entities interact as follows. The user takes the decision to attend an event, selects the event and sends the payment information to the ticket issuer. Then, the ticketing system issues the ticket and sends it to the secure element of the mobile phone of the user using the proper mean of communication (via a TSM or a secure channel). At the event gate, the user presents his NFC phone to provide the NFC-enabled ticket verifier with his e-ticket in order to be granted access. In this offline approach, the ticket verifier has the ability to control the tickets without the use of any external infrastructure, i.e. without any network connection.

**Requirements.** The user must own a NFC-enabled mobile phone with the ability to receive a e-ticket, to store it (in a secure element) and to transfer it to a NFC reader. All these actions can be managed by a mobile application deployed on the phone and its secure element. The Trusted Service Manager has its usual role since its main functionality is to load, when necessary, the digital ticket to the secure element embedded in the mobile phone by using the mobile operator network. The ticket issuer offers a web server and is responsible for the generation and issuance of the tickets. The web server shows the information concerning the available tickets to the user and receives the payment details. The digital tickets are then built by gathering different pieces of information and formatting them properly. Additionally, a digital signature can be applied if required for the verification. In this case, the signature is achieved by the ticket issuer with a private key, the corresponding public key[28] being provided to the ticket verifier. The ticket issuer transfers the ticket either over the mobile network or through any other kind of secure connection to the mobile phone secure element. Before the control, the ticket verifier must be provided with the relevant information regarding the corresponding event. In most cases, the event identifier and the ticket's period of validity are necessary elements for the verification. During the validation phase, the ticket verifier makes sure of the authenticity and integrity of the received ticket. In order to avoid a re-use, the verifier must either keep track of the tickets that are presented or modify their validity (date, status, etc.). A ticket verifier can be composed of a NFC reader embedded in a mobile phone or connected to a computer and linked to an application for the cryptographic and ticket management operations.

### 4.2 Online System

**Use Case.** Adam wants to go and see a university theatre show. He goes to the event organiser's website, where there is a link to the ticket issuer's website. He already has an account at this website so, he logs in, adds the proper ticket to his shopping cart and pays for it. When Adam goes to the show, he taps his phone on the reader and once the permission to enter has been checked by connecting to the ticket issuer, the light turns green and Adam can enter.

**Architecture and Interactions.** The online event ticketing paradigm is based on the premise that no dynamic information (here a ticket) is installed on the user's device. The assumption is that there is just a static identifier stored in the secure element of the user's mobile device, that same static identifier being stored in the ticket issuer's backend system where all dynamic information are processed. This means that a user does not have any ticket with him (in his mobile phone) when he goes to an event; he only has an identifier that will be used by the verifier to check by the ticket issuer that he is authorized to attend the event.

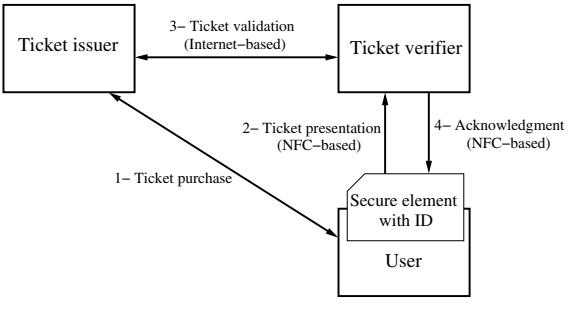



**Fig. 5.** Online System Architecture

**Requirements.** Online event ticketing requires a secure element to be available in the user's mobile device, where to store static identifiers. This secure element may be a (U)SIM or another secure element that is embedded in the device. Obviously for convenience reasons, an application loaded on the secure element ensures a proper setting and provides the static identifier when required.

The online event ticketing system is described in figure 5. The relationship between the event organiser and the ticket issuer is similar to what they are today: the ticket issuer sells the tickets and the event organiser validates them. In the online events ticketing scheme, event organisers need a working Internet connection at the event gate. At the gate the right to enter is verified by reading the identifier from the secure element of the user's mobile device and then sending it to the ticket issuer's system. It returns the authorization (or not) to enter.

When the user buys a ticket for a specific event from a ticket issuer, the ticket is stored in the ticket issuer's back end system and it is connected to the static identifier stored in the user's mobile device secure element. Because the user does not carry the

ticket information with him, the ticket issuer's back end system needs to be able to provide this ticket information when requested. This may be done by using the ticket issuer's Internet site.

## 5 Paradigms Comparison

### 5.1 Overview

To compare the two paradigms that we have described, we will focus on five criteria: user experience, security, economical aspects, reliability and speed of use. Some prototypes illustrating the online and offline systems were developed to run reliability and validation speed tests. The Nokia 6212 classic[29] has been used for this purpose. This NFC-enabled mobile phone can run J2ME[30] midlets and embeds an internal secure element which can run Java Card[31][32][33] applets. With two of these phones we have implemented a basic prototype for each paradigm we consider, online and offline. The first handset acts as the client device and has a ticket or a static identifier stored in its internal secure element. The second handset is the validator device and reads the ticket from the client device in order to check its validity (either offline or online). The secure element of the client phone is loaded with a Java Card applet containing the ticket information (either the whole digital ticket or the static identifier). The validator phone runs a J2ME midlet in charge of retrieving the ticket information and checking the validity of the ticket (either locally or by accessing a remote database via a HTTP request).

### 5.2 Security

In our study, the security analysis targets the validation phase. In this phase, online and offline approaches both make use of a secure element and rely on the NFC card emulation mode. Consequently, they achieve the same level of security. To proceed to the ticket validation, the user taps his phone on the reader what leads to a direct communication between the secure element and the reader. The use of a secure element prevents the static identifier or the whole ticket (depending on the approach) from being forged or spoofed as a secure element is assumed to be a tamper resistant device[34].

In both cases, to improve communication security, a mutual authentication is performed. This authentication, which uses the GlobalPlatform standard[35], ensures that the information exchanged (between the phone's secure element and the validator) only involves authorized entities (a real user and a real validator).

### 5.3 Reliability

Regarding the ticket issuance phase, both models have a strong need for a steady Internet connection. However, for the validation phase, only the online approach requires a connection in order to query the ticket issuer's database. Any disconnection at that time would prevent tickets from being validated. Another issue arises, for both paradigms, if the mobile handset runs out of battery during the validation process. In this situation, there can be no communication between the reader and the mobile device, thus

preventing the validation process from taking place. However, some NFC phones have a 'battery off' feature which allows the secure element to interact with the reader regardless of the battery state[36]. We believe that this feature will be common in future NFC-enabled phones and will thus improve the reliability of both approaches.

### 5.4 Speed of Use

Fast ticket validation is an important requirement. In the mass transit sector the transaction time should not exceed a few hundreds milliseconds[37]. In our tests validation is achieved in three steps: mutual authentication, data retrieval and network use (for the online approach). For the offline approach, the data retrieval targets the transfer of a 1 kilobyte ticket from the secure element to the validator and there is no network needed (as the ticket validity check is performed locally). For the online approach, the data retrieved from the secure element is the static identifier (a 10 bytes String) and the network is used to access the ticket issuer database (3G connection). The results presented in figure 6 show that the offline solution is much faster. It is nevertheless necessary to observe that the Nokia 6212, which is used as a validator, has no broadband Internet connection capability (no WIFI for instance) and that the validation time of the offline option increases with the size of the ticket.

### 5.5 User Experience

Our e-ticketing applications offer three main features: ticket purchase on a website; ticket presentation for validation purpose at the event entrance; listing of tickets that have been bought. As the online and the offine versions allow to perform the same actions and since the underlying processes (for the website and the tickets listing) are transparent to the user, we can assume that there is no major difference in the user experience. Nevertheless, for the ticket validation phase, we can find a small difference. In the online approach, there is no need to select the ticket to validate (no ticket is stored in the mobile phone and unique identifier is used), but in the offline approach the user must select the ticket he wishes to present what can slightly reduces the user-friendliness aspect.

### 5.6 Economical Aspect

NFC-based event ticketing brings new players to the industry as Mobile Network Operators (MNO) and TSMs providers can offer new channels for events ticketing; but it remains to be seen if they can offer these channels cheap enough or offer cost savings. Anyway, as customers learn to use ticketing with NFC in public transportation, the event ticketing domain becomes also under pressure to go compatible with NFC.

Generally, the economical aspect always depends on the business model which is used. For now, online and offline event ticketing are open fields where a lean business model might be created and a market niche be found. With the opportunities offered by NFC, the experience gained in transportation system and the release of new smartphones endowed with NFC, we believe that the big players can afford investments in

this business sector. For these companies, there is no significant difference in the investments that have to be made in terms of infrastructure (TSM, adapted NFC readers, backend systems, website) between the two approaches. Both systems could thus coexist.

### 5.7 Summary

The two paradigms, offline and online ticketing that we have presented, mainly differ in the need or not of online connection. Speed of validation and user experience were identified as parameters that can help decide which paradigm should be used in which context. However, the online and offline approaches are more or less equivalent in terms of performance. The comparison of the two approaches is summarized in figure 7. For each category, the symbol + represents an advantage and the - is a disadvantage; the = symbolizes a similar level of performance.

| | Offline | Online |
|---|---|---|
| Authentication | 161 | 161 |
| Data retrieval | 605 | 52 |
| Network Use | 0 | 4091 |
| Total (in ms) | 766 | 4304 |

**Fig. 6.** Validation speed tests results

| | Offline | Online |
|---|---|---|
| Security | = | = |
| Reliability | = | = |
| Speed of use | + | - |
| User experience | - | + |
| Economical aspects | = | = |

**Fig. 7.** Comparison of both approaches

## 6 Future Work

### 6.1 Pilots

Within the framework of the Smart Urban Spaces[38] (SUS) European project in which this research around NFC-based event ticketing is conducted, the deployment of pilots using some aspects of the proposed models description are planned. These pilots target small events such as private concerts or theater plays. For this kind of events, there is a strong need for a flexible and cheap ticketing system as the organisers cannot afford big investments. Some French and Finnish cities (working with LaBRI and VTT) like Helsinki or Caen are potential candidate to deploy these pilots.

### 6.2 Perspectives

The next step in our research is to focus on e-ticketing system based on NFC-enabled devices dedicated to small events. Small events, which are events with a limited number of attendees, represent an uncovered niche. This research will take into account the ease of deployment, a lightweight architecture (with no need of big external infrastructure such as a TSM for instance) and the reduction of the costs still targeting offline and online options.

Another point, regarding event ticketing management, is to work on a e-ticket standard description and the associated storage procedures inside mobile phones. As far as we know, there is no real event e-ticket standard and it would be thus relevant to make contributions in this area.

Finally, by leaning on the previous points, we will work on the concept of interoperability from the perspective of users running mobile NFC-based applications (such as the event e-ticketing system) in different European cities in the framework of the SUS project.

## 7 Conclusion

Two different approaches to deal with ticketing issues and their respective architecture, described as *offline* and *online* solutions, were presented in this paper. In the former solution, the platform responsible for validating the tickets at the entrance has no need to communicate with the ticket issuer, and thus does not use an Internet connection. Conversely, in the latter solution, a direct link exists between the ticket issuer and the ticket verifier platform. The prototypes that we have developped and that make use of NFC-enabled phones, demonstrate the feasibility of the proposed solutions. Although both of these approaches have advantages and drawbacks, the comparison shows that globally they can achieve the same level of performance. As big companies will most likely deploy online and offline applications for big events, it is certainly relevant to target small events in the next research topics in the context of a NFC-enabled mobile phone ticketing system.

**Acknowledgements.** The presentation of this paper is part of the work on the Event ticketing with NFC-enabled mobile phones service in the framework of Smart Urban Spaces[38] project. We would like to thank all the partners of the Smart Urban Spaces project with whom we have been working.

## References

1. Livenation financial reports: 2008 annual report (September 2010), `http://phx.corporate-ir.net/phoenix.zhtml?c=194146\&p=irol-reports`
2. Mpaa 2009 theatrical market statistics (September 2010), `http://www.mpaa.org/Resources/091af5d6-faf7-4f58-9a8e-405466c1c5e5.pdf`
3. Zmijewska, A.: Evaluating wireless technologies in mobile payments - a customer centric approach. In: Proceedings of the International Conference on Mobile Business, Los Alamitos, USA, pp. 354–362 (2005)
4. Samsung galaxy s ii (March 2011), `http://www.samsung.com/global/microsite/galaxys2/html/feature.html`
5. Madlmayr, G., Langer, J., Scharinger, J.: Managing an nfc ecosystem. In: Mobile Business 2008, Barcelona, Spain (2008)
6. Characteristics of the universal subscriber identity module (usim) application. 3GPP, Tech. Rep. 3GPP TS 31.102 version 9.3.0 (June 2010)
7. Reveilhac, M., Pasquet, M.: Promising secure element alternatives for nfc technology. In: 1st International Workshop on NFC, Hagenberg, Austria (2009)

8. Preuss, P., Reddmann, D., Weigt, F.: RMV-HandyTicket fr NFC-Handys. In: Tuikka, T., Isomursu, M. (eds.) Touch the Future with a Smart Touch, Espoo, Finland, pp. 89–90 (2009); no. Research notes 2492
9. Suikkanen, J., Reddmann, D.: Vision: Touching the Future, Ticketing. In: Tuikka, T., Isomursu, M. (eds.) Touch the Future with a Smart Touch, Espoo, Finland, pp. 233–236 (2009); no. Research notes 2492
10. The keys to truly interoperable communications. NFC Forum, Tech. Rep. (2007)
11. Near field communication in the real world - turning the nfc promise into profitable, everyday applications. Innovision Research and Technology, Tech. Rep. (2007)
12. Cityzi (February 2011), http://www.cityzi.fr/
13. Air france online check-in (February 2011), http://www.airfrance.com/HR/en/common/guidevoyageur/e_services/mobile_cab_airfrance.htm
14. Transport for london - oyster card website (September 2010), https://oyster.tfl.gov.uk
15. Beijing municipal administration and communications card co - website (September 2010), http://www.bmac.com.cn
16. Mf1s5009. NXP, Eindhoven, The Netherlands, Tech. Rep. rev. 3-189131 (July 2010)
17. Octopus card website (September 2010), http://www.octopus.com.hk
18. Felica page (October 2010), http://www.sony.net/Products/felica/
19. Ghiron, S.L., Sposato, S., Medaglia, C.M., Moroni, A.: Nfc ticketing: A prototype and usability test of an nfc-based virtual ticketing application. In: International Workshop on Near Field Communication, pp. 45–50 (2009)
20. Bpass (February 2011), http://www.veolia-transport.com/fr/medias/zoom/bpass-nice.htm
21. Digitick (February 2011), http://www.digitick.com/pocket-css4-digitick-pg3021.html
22. Neefs, J., Schrooyen, F., Doggen, J., Renckens, K.: Paper ticketing vs. electronic ticketing based on off-line system 'tapango'. In: International Workshop on Near Field Communication, pp. 3–8 (2010)
23. Smarttouch (February 2011), http://ttuki.vtt.fi/smarttouch/www/?info=intro
24. Rouru-Kuivala, O.: Vision: Touching the Future, Ticketing. In: Tuikka, T., Isomursu, M. (eds.) Touch the Future with a Smart Touch, Espoo, Finland, pp. 171–173 (2009); no. Research notes 2492
25. Sadeghi, A.-R., Visconti, I., Wachsman, C.: User privacy in transport systems based on rfid e-tickets
26. Siu, S., Guo, Z.S., Fong, S., Zhuang, S.: Extending e-ticketing service with mobile transactions
27. Trusted service manager service management requirements and specifications. EPC-GSMA, Tech. Rep. EPC 220-08 version 1.0 (January 2010)
28. Bellare, M., Rogawayy, P.: Optimal asymmetric encryption - how to encrypt with rsa. In: De Santis, A. (ed.) EUROCRYPT 1994. LNCS, vol. 950, pp. 92–111. Springer, Heidelberg (1995)
29. Nokia 6212 specifications (March 2011), http://europe.nokia.com/support/product-support/nokia-6212-classic/specifications
30. Li, S., Knudsen, J.: Beginning J2ME Platform - From Novice to Professional. Apress, New York (2005)
31. Virtual machine specification - java card platform. Sun Microsystems, Tech. Rep. version 2.2.2 (March 2006)
32. Runtime environment specification - java card platform. Sun Microsystems, Tech. Rep. version 2.2.2 (March 2006)

33. Application programming interface - java card platform. Sun Microsystems, Tech. Rep. version 2.2.2 (March 2006)
34. Requirements for nfc mobile: Management of multiple secure elements. Global Platform, Tech. Rep. version 1.0 (February 2010)
35. Card specification. Global Platform, Tech. Rep. version 2.2.1 (January 2011)
36. Nfc in public transport. NFC Forum, Tech. Rep. (January 2011)
37. Transport for london (December 2010), http://www.nfctimes.com/news/transport-london-calls-faster-nfc-sims
38. Smart urban spaces website (September 2010), http://www.smarturbanspaces.org

# Mobile Electronic Memos

Giovanni Bartolomeo[1], Stefano Salsano[1], and Antonella Frisiello[2]

[1] University of Rome Tor Vergata, via del Politecnico, 1
00133 Rome, Italy
{Giovanni.Bartolomeo,Stefano.Salsano}@Uniroma2.it
[2] Istituto Superiore Mario Boella, via Boggio, 61
10138 Turin, Italy
Frisiello@ISMB.it

**Abstract.** This paper explores a novel concept called Mobile Electronic Memo whose purpose is to overcome the limitations in terms of interoperability and accessibility imposed by the current technology. To achieve this goal, Mobile Electronic Memos decouple the semantics of the information from the physical and pragmatic constraints of the medium that transports it. We examine related works and standards that could be good candidates for implementing MEMs and describe a prototype implementation that has been realized as a proof of concept.

**Keywords:** Electronic information, interoperability, accessibility.

## 1 Introduction

Often the information one needs is already available, somewhere, in an electronic format. In most cases, the information is downloadable from the Internet; or it could be gathered from nearby devices, from the environment (e.g., through sensors) or even from real world objects (e.g., using RF-IDs and QR codes). Once obtained, this information could be stored on consumer electronics (computing devices, mass storages, memory cards), or be shared on the Web. It could be also reused and transmitted asynchronously over a long distance through emails or multimedia messages; or it might be sent as an input to other devices or applications (navigators, ATMs and vending machines, home or car appliances). Zillions of information are daily exchanged this way; but, in many cases, interoperability between the device providing an information and the one consuming it is not automatic, but it is mediated by human intervention, often involving manual inputs.

Unfortunately, this well known limitation might cut off several categories of people from the benefits of the ICT, resulting in a possible reduction of the inclusion opportunities (one aspect of the so called “digital divide”) not because of the lack of technology but due to its excessive complexity [1].

For example, often elderly people have difficulties in interacting with their own home appliances (including mobile and e-health devices [2][3]) and they prefer to ask for help to more skilled relatives (or not to use these devices at all). While this

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 178–187, 2011.

situation is annoying for most users, it could even represent a barrier for people with impairments; in fact, often it is the surrounding ICT environment - rather than the nature of her own impairments – that tends to disable a person, reducing her possibilities to access information and services.

In this paper we illustrate a novel technology called Mobile Electronic Memo (MEM) whose purpose is to overcome the limitations imposed by the current handling of electronic information. A MEM attempts to achieve this goal by decoupling the information it carries from its physical and practical limitations. We explain this underlying principle and the relevant benefits it might bring in the next section. In section 3 we take a closer look at the design of MEMs, providing a description of a prototype implementation that has been realized as a proof of concept. Moreover, we hint at security and trust aspects, illustrating one key feature that has been implemented and experimented in our prototype: the signature of a MEM using an Universal Integrated Circuit Card (UICC). We conclude this paper with a discussion of related works and technologies, explaining also why, in our opinion, some of these technologies could be good candidates to implement MEMs as an industry-wide standard.

## 2 Benefits of a Pure Information Centric Model

Web pages, business cards, short messages and GPS Point of Interests: all of them are examples of electronic information. But why should they be primarily thought in terms of readable text instead of Braille dots or spoken words?

In his essay [4], Riva describes media under three main aspects: the physical aspect, the pragmatic aspect and the symbolic aspect. The physical aspect refers to the natural properties of the media (hardware, interfaces) and it is strongly related to the pragmatic aspect, which denotes people's interaction modalities with the media. The symbolic aspect indicates the meaning received by the people, i.e. the semantics. In traditional media there are tight constraints between and among these three aspects. This is one reason that explains why we are used to thinking at a web page or a business card as something readable, rather than e.g. tactile.

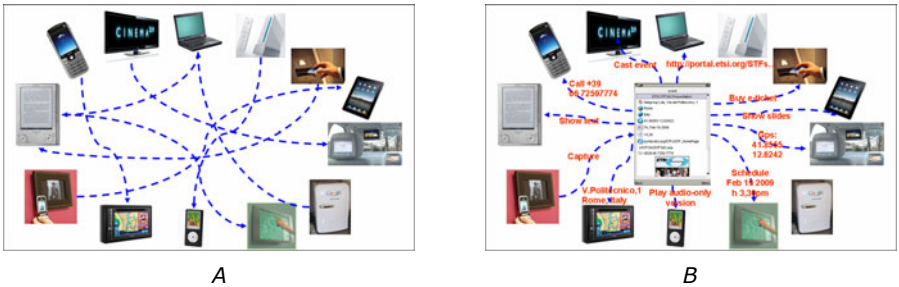



**Fig. 1.** From "spaghetti" information exchanges (a) to an information centric model (b)

On the contrary, making looser these constraints would bring two classes of benefits: firstly, development of multimodal and "assistive" technologies would be greatly simplified, being the same electronic information decoupled from specific devices and technologies; in the spirit of the "design for all" principles [5], even the word "assistive" could be no longer needed, because the information would be solely thought in terms of its semantics rather than of its rendering techniques. Secondly the "spaghetti" information exchanges depicted in (Fig. 1a) would disappear in favor of an electronic information centric model (Fig. 1b), where standardized information, expressed as MEMs, could be automatically exchangeable and interoperable across many different devices and appliances, through a number of different communication technologies.

## 3 Mobile Electronic Memos

A Mobile Electronic Memo (MEM) is a communication technology that does not assume a specific physical support or interaction mechanism, but represents the semantics of the information it carries in a neutral way, using a machine understandable representation (i.e. an ontology). In particular, it supports the following features:

1. it is a versatile container of information;
2. it is machine understandable – therefore independent from specific devices, user interfaces and human languages;
3. it can be transported across different kinds of bearers and networks;
4. it supports different user interaction paradigms, depending on the device or appliance it is executed on.

From an end user's point of view, MEMs may be created or captured from several sources; examples include, but not limit to: capture from the environment [6] via Near Field Communication (NFC), RFID, Bluetooth; optical capture through a QR code; location-based capture with GPS coordinates used to "lookup nearby MEMs", retrieval from the Web or from a mass storage memory, etc.

Once created or captured MEMs may be annotated, personalized, shared with other users or transferred to nearby devices. Users can browse created, received or captured MEMs using a variety of modalities.

MEM-aware devices and applications may provide several useful services based on the information MEMs contain: for example, a MEM aware navigator might drive the user to a location encoded in a captured MEM; a MEM aware smartphone might start a phone call or schedule an event in the user's agenda, using the information (telephone numbers, meeting details) contained in a received MEM; a MEM aware TV handset might auto-configure its settings taking their values from the information contained in a MEM, etc.

### 3.1 MOVE

Despite the current lack of standard technologies able to fully support the concept of MEM, many of the aforementioned features have been experimented in a prototype

system including MEM-aware software and devices. The prototype system, named Mobile Open and Very Easy (MOVE), has been developed in the context of the EU co-funded research project Simple Mobile Services (IST 2006 034620).

MOVE is based on a modular architecture. The base modules, implemented in Java Microedition (Java ME), are the Core, the Middleware, the GUI and the Outdoor & Indoor Navigator modules. In particular, the Middleware module consists of an abstract API that hides the underlying transport mechanisms, enabling devices running MOVE to pass messages using either the HTTP protocol or the SIP protocol.

The GUI module is based on a porting of Thinlet, a very efficient toolkit which uses a XML-based model for the graphical widgets. MOVE extends the original porting in order to provide an easier object oriented toolkit for GUI programming, fitting the different features and interaction modalities of a variety of devices. The Outdoor & Indoor Navigator module provides maps and navigation, and a collection of functionalities to search for a place, to search for a route and to get GPS (or approximate) positions. Many of its functionalities are provided by the open source Open Street Map platform – which however offers a not geographically uniform quality of the service: coverage is better in large cities, and variable, depending upon the specific country, in small cities, villages and rural areas.

In order to port MOVE over different mobile platforms, including Symbian, Android, BlackBerry, and Windows Mobile, we used an our own building tool that integrates the open source J2ME Polish Janus toolset. Third party programmers can develop additional modules for MOVE either using Java ME or platform specific languages. Modules developed using platform specific languages are wrapped into Java ME classes in order to provide unified interfaces to the rest of the framework. The communication with a platform specific API is usually realized using local IP sockets.

The full list of features offered by MOVE together with the list of supported platforms is provided in [7].

### 3.2 A Proof of Concept

In the following we illustrate part of the Simple Mobile Services project trial which consisted in using MEMs and the MOVE framework to announce a real event to different groups of people in the campus of the University of Rome Tor Vergata. This trial was intended to demonstrate how interoperability among several different technologies could be easily achieved by using MEMs; accessibility and multimodality were not particularly emphasized, but, as it has been observed, the highly degree of interoperability shown during this trial could be a key feature to foster the take-up of accessible and multimodal products and services based on MEMs.

In February 2009, two ETSI experts presented their standardization activity on "Personalization and User Profile Management" to researchers, post-docs and PhD-students at the University of Rome Tor Vergata. This presentation was also advertised by distributing MEMs to invited participants. Some participants received the invitation in an email message containing an hyperlink to the MEM. Others received the MEM in their smartphones (which were running MOVE) through an instant

messaging application. The organizers also exposed a physical poster which contained details about the presentation and a QR code. Capturing the QR code through the phone camera triggered the advertising MEM to be downloaded directly into the phone's memory. Some students were invited to diffuse the invitation to friends by transferring the MEM between their phones using Bluetooth, NFC or memory cards.

Smartphones and laptops stored the received MEM as a file in their local filesystems. The file was displayed on their screens as a traditional hypertext containing information about the presentation. However other kinds of media enrichments (e.g. audio clips) could have been included in the MEM, making it suitable for running on other devices (e.g. portable audio/video players, gaming devices, etc.)

The advertising MEM included a phone number to dial in order to obtain further information, the date, time and duration of the presentation and the GPS coordinates of the location where the presentation was going to take place. By clicking on the displayed phone number, smartphones were able to initiate a call toward that number. Opening the MEM in a (MEM-aware) calendar application triggered the information contained in the MEM to be converted into an iCal event suitable to be automatically scheduled into the user's agenda. Finally, the MEM could be transferred to a navigator appliance[1] (via Bluetooth). The navigator responded to this action setting the GPS coordinates contained in the received MEM as the next planned destination.

### 3.3 Trust in Mobile Electronic Memos

Given the very wide scope they aim to cover and, consequently, the possible presence of sensitive information, security and trust are one of the primary concerns that have been considered in designing MEMs. As an example, in order to prevent users from spam and phishing, MEMs or part thereof may be encrypted and digitally signed by their authors, allowing the recipient to verify their trustworthiness. In addition, one could think at digital signature as a way to implement non-repudiation capability, thus making MEMs suitable for proof of purchases and subscriptions.

In order to fit these goals, not only service providers but also end users should be able to sign and encrypt their own MEMs; this requirement could implemented using UICC functionalities [8]. Real UICCs featured with Javacard applets were used in the IST-SMS trial to realize the scenario depicted in Fig. 2, where two users exchange a trusted MEM. User A is asked to sign the MEM she is sending because a policy defined by user B does not allow the acceptance of unsigned MEMs. User A enters her PIN code in order to access the key-ring contained in her phone's UICC. Then, her device signs the MEM. Finally, the MEM is sent to user B which can browse its trusted content.

Obviously some transactions could employ lighter weight security mechanisms, in order to avoid the overhead due to the management of certificates and keys. For example, a digest might be sufficient to prove the integrity of the information when the MEM is sent via a slow, but trustworthy channel. Since the present paper is mainly intended to provide an overview of the concept of MEM, we omit to describe

[1] The navigator was emulated using a smart phone equipped with a navigation software.

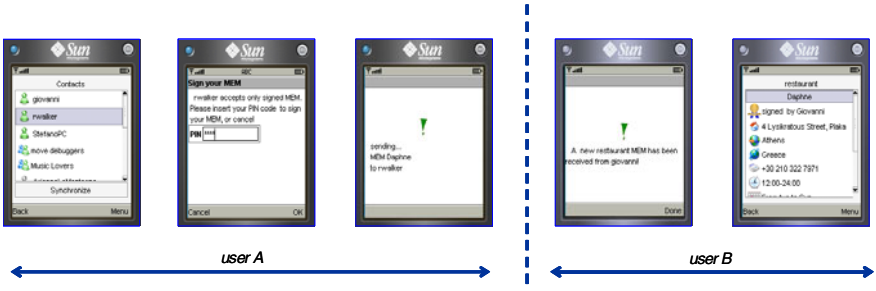



**Fig. 2.** Securing a MEM using UICC signature

other technical options; however, the reader could find lots of interesting details in a publicly available technical report about MEMs and their prototype implementation [9].

## 4 Related Works

There exist several standard formats designed for information exchanges and already supported by different kinds of devices (mainly computers and mobile phones); Few examples include:

1. vCard [10], the consolidated electronic format for business card;
2. iCal [11], a widely supported file format allowing users to send meeting requests and tasks to other users;
3. xAL [12] and PIDF [13], respectively OASIS and IETF standards for codifying addresses and geographic locations…

However, these formats are typically limited to bits of information intended for specific purposes, e.g., exchanging information about a specific place or an event.

Single versatile containers for heterogeneous types of information are under investigation in many standardization fora. For example, both the Microformats Forum [14] and the MPEG-21 Group, pursuing different goals, are standardizing flexible formats to support the structured aggregation of different kinds of information and resources. MPEG-21 explicitly provides support for alternative modes of presentation of digital resources (images, text, audio, etc.). It also specifies tools to support accessibility of digital resources to various users, including users with audio-visual impairments [15].

### 4.1 Semantic Technologies

As the main finding behind the idea of MEM is expressing information in a machine readable format, we now provide a very brief review of related works based on semantic technologies.

Ontologies are not new in computer science, however they have been traditionally limited to specific environments (reasoning and logics) or applications (the Semantic Web and Linked Data).

Their usage as a "neutral" medium to convey electronic information among heterogeneous devices is a relatively new topic of research which has been recently investigated in the context of the ARTEMIS-JU SOFIA (Smart Objects For Intelligent Applications) research project and in the Finnish national research projects DIEM (Device Interoperability EcosysteM) [16][17]. The two projects have produced a platform called SMART-M3 which combines distributed networked systems with the Semantic Web, enabling devices and software entities to exchange information conforming to a common ontology.

A second promising direction of research is semantic annotation of already existing formats. RDF-a [18] is a recent proposal to augment Web pages with RDF statements, turning them into richer container of semantic information that can be understood and automatically processed by software. RDF-a is currently used with Web pages and within Web browsers. Its combination with other file formats spanning across different devices and appliances (e.g. MPEG-21) is unexplored and, we suggest, could represent a good candidate for the implementation of the concept of MEM.

### 4.2 Machine-to-Machine Interactions

Fitting the goal of a common file format and of consolidated semantics might be not enough, as the interoperability outside the boundary of the Web could be harder to achieve; this is mainly because of missing common information presentation facilities and the lack of standards for the transport of the information. Then, even if one could be tempted to rely always on Web-based solutions, there are many practical cases in which these solutions are not viable. This occurs, for example, when:

1. The information is better transmitted -- over a relatively long range -- in a peer--to--peer fashion, using e.g. a short message or a MMS;
2. Many bearers are available and it is preferable to use a minimum range connection technique (e.g. a PAN, or even transfer by USB sticks and memory cards), as this might have a major impact on performances and energy efficiency;
3. Short range connections are the only transmission mechanism (e.g., when communicating with devices without Internet connection, such as navigators, home or car appliances, portable media players, etc.);
4. Transfer over memory cards is the only transmission mechanism (e.g., when in "flight mode")
5. Users might be reluctant to post privacy sensitive information on the Web, as at today there is no effective solution to the "digital forgetting" problem (sensitive or inappropriate information posted on the Internet could hunt one forever).

The ongoing standardization work at ETSI Machine-to-Machine Technical Committee (M2M TC) is trying to address these situations by specifying a framework supporting transmission of information between devices on a variety of protocols. In Table 1 we list several requirements collected for a MEM aware system, together with corresponding capabilities of a M2M system [19] matching these requirements.

**Table 1.** MEM requirements and M2M capabilities

| Requirement | M2M corresponding capability |
|---|---|
| Several different devices (smartphones, navigators, media players, personal organizers, home/office appliances, incar devices), should be supported. | A M2M system natively supports a wide range of different devices. |
| An uniform identification mechanism for end users, devices and groups is needed. | The M2M system allows flexible addressing schemas. |
| Various forms of asynchronous and unsolicited communications should be supported; "Autoplay" should be allowed. Various forms of delivery (anycast, unicast, multicast, broadcast) should be supported. | The M2M system provides a wide range of message delivery modes. It even provides delivery towards sleeping devices. |
| It should be possible to transmit MEMs over a large set of bearers (including short range connectivity and "delay tolerant links", e.g. memory cards). | The M2M system provides message exchanges across multiple communication paths, based on policy rules and taking into account charging, privacy, energy saving, performances. |
| Multimedia should be supported.<br><br>Outdoor localization should be supported. | The M2M system provides telecom capabilities exposure (localization, subscription configuration, authentication, reachability information, messaging and IMS access). |
| Indoor localization should be supported.<br><br>It should be possible to "capture" a MEM from the surrounding environment, for example using NFC, RF-ID, QR codes or Bluetooth. | Sensors connected to the M2M system could provide information to be included into a MEM (position, temperature, speed, light intensity, pressure,…) or they could provide a way to retrieve a whole MEM (e.g. reading a visual code, detecting an RF-ID,…). |
| MEMs may implement spam/phishing prevention, non-repudiation capability, confidentiality, proof of purchases and subscriptions. | The M2M system shall provide a secure and trusted environment (based on, e.g., the UICC) for security critical operations. |
| Some use cases for MEMs may involve charging (e-Tickets, pass tokens, etc.) | The M2M system shall support micropayments. |

## 5 Conclusions

In this paper we presented the concept of Mobile Electronic Memo (MEM), a versatile, machine understandable "aggregator" of information. Unlike other media, a MEM does not impose any particular physical or pragmatic constraint and focus only on the symbolic aspect of the communication. Technically, this is made possible by replacing human readable contents with a semantic description of the information. Major benefits brought by this approach include interoperability and accessibility.

Despite a prototype system (including some basic security features) has been implemented, at the time of writing there is no standard framework that could fully support MEMs; nevertheless we believe that a combination of existing XML/RDF based languages and technologies under development for machine-to-machine systems could provide a valuable opportunity for MEMs to evolve into an industry-wide standard.

**Acknowledgments.** We would like to thank all people that contributed to the development of MEMs in the context of the EU co-funded research project Simple Mobile Services. In particular we thank N. Blefari Melazzi, project coordinator, R. Glaschick (Siemens C-Lab) for his key contributions on security, privacy and trust aspects and C. Rust (Sagem Orga) and A. Rabbini (Telecom Italia) for having designed and provided special UICCs supporting MEM encryption and signature. Last but not least we thank ETSI members for the valuable feedback received during the presentation of our work to the ETSI Human Factors Technical Committee. Current activities on MEMs at University of Rome Tor Vergata are supported by CREALAB and ErgonixART in the context of the POR FESR project "wetourist" funded by FILAS/Regione Lazio.

## References

1. Klironomos, I., Antona, M., Basdekis, I., Stephanidis, I.: Promoting Design for All and e-Accessibility in Europe. Universal Access in the Information Society 5(1), 105–119, doi:10.1007/s10209-006-0021-4
2. Istepanian, R.S.H., Lacal, J.C.: Emerging mobile communication technologies for health: some imperative notes on m-health. Engineering in Medicine and Biology Society. In: Proceedings of the 25th Annual International Conference of the IEEE, September 17-21 (2003), doi:10.1109/IEMBS.2003.1279581
3. Hubert, R.: Accessibility and usability guidelines for mobile devices in home health monitoring. ACM SIGACCESS Accessibility and Computing (2006)
4. Riva, G.: Psicologia dei nuovi Media. Il Mulino, Bologna (2004)
5. The Center for Universal Design: The Principles of Universal Design, Version 2.0. North Carolina State University, Raleigh, NC (1997)
6. Herting, T., Broll, G.: Acceptance and Usability of Physical Mobile Applications. In: Mobile Interaction with the Real World (MIRW 2008), Workshop at the 10th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI 2008), Amsterdam, the Netherlands, September 2 (2008) ISBN 978-3-8142-2134-2

7. Salsano, S., Bartolomeo, G.: MOVE, Thinlet and HECL, Technical Report, http://netgroup.uniroma2.it/SMS/TechnicalReports
8. Rust, C., Salsano, S., Schnake, L.: The SIM card as an Enabler for Security, Privacy, and Trust in Mobile Services. In: ICT-MobileSummit 2008, Stockholm, Sweden, June 10-12 (2008) ISBN: 978-1-905824-08-3
9. Salsano, S., Bartolomeo, G., Glaschick, R., Walker, R., Blefari-Melazzi, N.: Mobile Electronic Memos for Simple Mobile Services. Technical report, IST-SMS project (2008)
10. Dawson, F., Howes, T.: RFC 2426 - vCard MIME Directory Profile. IETF (1998)
11. Dawson, F., Stenerson, D.: RFC 2445 - Internet Calendaring and Scheduling Core Object Specification (iCalendar). IETF (1998)
12. Customer Information Quality Technical Committee: extensible Address Language (xAL) – v2.0. Technical specifications. OASIS (2004)
13. Sugano, S., Fujimoto, G., Klyne, G., Bateman, A., Carr, W., Peterson, J.: RFC 3863 - Presence Information Data Format (PIDF). IETF (2004)
14. The Microformats community, http://microformats.org/
15. Vetro, A.: MPEG-21 Digital Item Adaptation: Enabling Universal Multimedia Access. IEEE Multimedia 11(1), 84–87 (2004), doi:10.1109/MMUL.2004.1261111
16. Liuha, P., Lappeteläinen, A., Soininen, J.-P.: Smart Objects for Intelligent Applications - first results made open. ARTEMIS Magazine (5), 27–29 (2009)
17. Koljonen, T.: ARTEMIS and the rest of the world. ARTEMIS Magazine (5), 30–31 (2009)
18. Birbeck, M., Pemberton, S., Adida, B.: RDFa Syntax: A collection of attributes for layering RDF on XML languages. W3C (2008)
19. Machine to Machine Technical Committee: TS 102 689 v1.1.1 Machine to Machine communications (M2M), M2M service requirements. Technical specifications. ETSI (2010)

# Better Performance of Mobile Devices in Time-Frequency Dispersive Channels Using Well-Localized Bases

Dmitry Petrov and Timo Hämäläinen

Department of Mathematical Information Technology, Faculty of Information Technology, University of Jyväskylä, PO. Box 35 (Agora), 40014 Jyväskylä, Finland
{Dmitry.A.Petrov,Timo.T.Hamalainen}@jyu.fi

**Abstract.** The main aim of this paper is to show on the conceptual level the practical and commercial benefits of signal construction and processing technology based on orthogonal frequency and time domains multiplexing (OFTDM). This technology utilizes mathematical framework of special orthogonal bases with the best time-frequency localization. Higher spectral and energy efficiency of telecommunication systems together with robustness against complex realistic channel conditions are achieved. In particular, the level of interference between subcarrier channels in time and in frequency domain is minimized. This approach can have wide implementation in wideband mobile networks (WiMAX, LTE), digital television (DVB-T/H) and other telecommunication systems.

**Keywords:** WiMAX, LTE, OFDM, OFTDM, Well-localized bases, time-frequency dispersion.

## 1 Introduction

Constriction of high-speed wireless digital telecommunication systems often faces the problem that real radio channel (propagation media) is time-frequency dispersive [1]. In particular this follows from the fact that radio signal comes to the receiver through multiple paths after many reflections from the nonstationary media inhomogeneities like city buildings, moving objects, hydrometeors, ionospheric layers, etc. Among the examples of such dispersive channels are wideband multiple access radio lines (mobile WiMAX, LTE), digital television (DVB-T / H), short-wave and ultra-short-wave radio lines.

Currently one of the most popular physical layer technologies for data transmission is multiplexing with orthogonal frequency division (OFDM). There is no doubt that this technology will be also used in future telecommunication standards.

As a result of time-frequency dispersion of the OFDM signal such effects as multipath propagation, amplitude-phase fading, Doppler shift and spreading are observed at the receiving side. Those effects are even stronger if receiver is moving in a car, situated in the building or near the strong source of electromagnetic emission

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 188–196, 2011.

like airport. As a consequence intercarrier (ICI) and intersimbol interferences (ISI) considerably worsen the quality of the received signal. Moreover ICI cannot be compensated or filtrated by regular digital processing methods.

It is necessary to mention that the structure and properties of OFDM signals are determined by the basis, which was used for its construction (signal basis). In classical OFDM systems the role of such basis plays the family of rectangular initializing functions shifted in time and frequency. Thus OFDM signals are constructed as the linear combination of such basis functions with real or complex data symbols (determined by the signal constellation: QAM, PSK, ets.) as the coefficients. Channel equalization is simplified because OFDM may be viewed as many slowly modulated narrowband signals rather than one rapidly modulated wideband signal.

In channels with time-frequency dispersion complex multiplicative interference affects the signal in addition to additive noise. Such interference has the factor of time dispersion characterized by multiplicative action in frequency domain, which is equivalent to the convolution with the signal in time domain. For the factor of frequency dispersion it is vice versa: multiplicative action in time domain and convolution in frequency domain. As a result signal basis is distorted, it becomes nonorthogonal and Nyquist theorem [2] is not fulfilled any more.

In other words the appearance of ICI and ISI in time-frequency dispersive channels is caused by the loss of orthogonality between disturbed basic functions. The demodulation procedure in the receiver becomes nonoptimal. The leakage of information from every channel subcarrier to the neighboring channels takes place. Notably the value of this mutual interference depends on the time-frequency localization of signal basis functions and is determined by the support of their uncertainty functions. The faster decay the tails of the uncertainty function the better is time-frequency localization of the signal basis and thus the less is the level of ISI and ICI.

The low symbol rate of OFDM systems allows to use a guard interval or cyclic prefix (CP) between consecutive OFDM symbols. The length of the CP - $T_s$ is longer than the time dispersion of the channel. Because of that it is possible to handle time spreading and eliminate ISI. Thus the effect of time dispersion can be effectively compensated but with the loss of spectral and energy efficiency. In particular, the cyclic prefix coasts a loss of spectral efficiency by $T_s/(T_s+T_0)$, where $T_0$ is the initial OFDM symbol duration [3]. It also implies the same order of power loss. In spite of several efforts [4], [5] towards the reduction of these overheads CP rests very simple and effective approach.

The rectangular form of forming functions used in classical OFDM systems in time domain correspond to sinc(x) or sin(x)/x functions in frequency domain. It is not optimal from the point of ICI. The level of out-of-band emission is overrated.

This is one of the reasons of connection breaks when a subscriber enlarges its velocity or when the signal/noise limit is exceeded. It causes synchronization upsets or inaccurate assessments of channel parameters when frequency dispersion and ICI are strong.

Now we will briefly consider several known methods of ICI reduction:

- In telecommunication systems based on OFDM principals signal basis functions represent the segments of harmonics. In frequency domain they have slowly decaying tails. It is possible to improve the robustness of such signals against Doppler effects by spreading the spectrum of basic functions, e.g. by shortening the duration of harmonics. This results in spreading of the signal's spectrum and in extension of distance between subcarriers. However such changes are not always allowable in terms of existing standard. In addition they do not rescue against the part of intercarrier interference caused by overlapping of side lobes of sinc functions.
- Sometimes ICI could be additionally mitigated by adding guard intervals in the form of dummy subcarriers at the borders of frequency range and between informational subcarriers.
- Other approach to the decision of ICI problem is based on the usage of window Fourier transform. In this case nothing is changed at the transmitting side but at the receiver the initial orthogonal FFT basis is replaced with nonorthogonal basis of weighted FFT with better localization in frequency domain. This method allows to reduce the component of ICI caused by overlapped side lobes of basis' functions. However window function spreads the spectrum of each subcarrier at the receiver. This results in overlapping of main peaks of basic functions and again in the leak of information from one subcarrier to another but in different from. In addition refusal from orthogonality increases the noise level.
- Next approach is based on the generalization of Nyquist-Kotel'nikov-Shannon sampling theorem. The main idea of this approach is based on oversampling and usage of series with well-localized core functions for signal interpolation. In other words received signal is discretized with frequency much higher than critical Nyquist frequency. After that for signal reconstruction so called atomic functions [5] are used. Family of such functions shifted in frequency domain can also be considered as a signal basis [6]. Because of oversampling the main lobe of atomic function can be selected in correspondence with frequency range of subcarrier channel of OFDM signal. Moreover such functions have fast decaying side lobes. This approach seems to be one of the most promising. Nevertheless basis constricted from atomic functions cannot be always made orthogonal. This complicates signal processing and reduces robustness against noise interference.

From the foresighted analysis it follows that the problem of ICI reduction in mobile OFDM systems is still actual and does not always have satisfactory decision.

## 2 Orthogonal Frequency and Time Domains Multiplexing

The main idea our research is to use well-localized basis function instead of rectangular ones used in classical OFDM. The optimal localization and "tuning" of basis parameters reduce the out-of-band emission and mutual interference of subcarriers in frequency domain.

Utilization of well-localized bases requires more complex synthesis and processing procedures. That is why the important part of the research is devoted to the development of computationally efficient methods which are comparable to discrete

Fourier transform (DFT) used in classical OFDM. Thus the whole scope of works includes the following stages:

1. Determine the type and structure of signal basis that fulfills several important requirements:
   - From the digital nature of signal processing it follows that the basis should be discrete and defined in finite number of points (finite support of basic functions). This makes clearer it's practical utilization and future technical realization.
   - Of cause, basis should be orthogonal.
   - Basis should have good time-frequency localization under some criteria.
   - The symmetry of basis' functions is not an imperative requirement but there is no reason to make overlapping of functions stronger from one of the sides in the signal.
2. After the structure of the basis was determined it was necessary to develop efficient methods of its synthesis.
3. In addition to that it was necessary to explain theoretically the choice of basis parameters which allow to adjust localization characteristics.
4. The matrix form of the basis gives rather straightforward approach to signal modulation and demodulation: these operations can be performed by vector matrix multiplication. However it is possible to achieve much better computational efficiency taking into account the structural particularities of basis' functions.

Steps from 1 to 4 are described in more details in works [7], [8], [9] and will be briefly considered in section 3. They form the mathematical framework of the OFTDM technology.

The next part of the research which is on the go now is more practical and includes following steps:

5. Analyze on the link level how channel conditions influence the characteristics of OFTDM system. In particular, it is necessary to receive bloc error rates (BLER) for different values of interference and block size. It will be the input for the next step.
6. Analyze the performance of the OFDTM based system on the system level and compare it to classical OFDM systems.

Future benefits of proposed approach are given in section 4.

## 3 Well-Localized Weyl-Heisenberg Bases and Signal Structure

Fourier transform is a powerful instrument of signal analysis in linear time invariant systems. Nevertheless it is complicated to use it for short-term or transitional processes when we need information about spectrum localized in time. Development of some universal basis (analogical to Fourier basis) which could simplify the processing of most types of signals is a very difficult problem [10]. Several known examples of such bases exit including wavelet bases, bases constructed from splines and atomic functions, etc. Weyl-Heisenberg bases were initially derived from Gabor bases and can be constructed by discrete shifts in time and frequency of initializing function (or family of initializing functions in more general case).

The quality of time-frequency localization of such bases is limited by two constrains:

- Fundamental Heisenberg uncertainty principle which states that with improvement of localization of function in time domain we lose in frequency localization and vice versa. Mathematically it is described by the following relation: $\sigma_t^2\sigma_\omega^2 \geq 1/4$, where $\sigma_t^2$ and $\sigma_\omega^2$ is variance in time and frequency domain correspondingly. The equality is attained only for Gaussian function.
- From the Balian-Low theorem, it follows that the use of only one initializing function leads to the fact that the obtained Wey-Heisenberg bases are poorly localized in the case of maximum density of discrete time-frequency lattice.

However the last constraint can be overcame by the use of two types of initializing functions in the basis and special orthogonality condition. Thus the number of basis' functions doubles in compare to classical Weyl-Heizenberg basis used in OFDM. At the same time instead of complex QAM modulation coefficients their real and imaginary parts are used so that changes should be made manly in the physical layer. This idea was firstly proposed in the paper [11]. The main difference and advantage at the same time of our approach is that the basis is initially considered in the finite-dimensional space of $N$-periodical functions. $N = L \cdot M$, where $M \geq 2$ is the number of subcarriers, $L$ - any natural number unequal zero which corresponds to the number of shifts in time domain.

Generalized Weyl-Heisenberg basis $\mathcal{B}[J_N]$ and transmitted OFTDM signal $s(t)$ in discrete time can be presented in the following form

$$s[n] = \sum_{k=0}^{M-1}\left(\sum_{l=0}^{L-1} c_{k,l}^R\, \psi_{k,l}^R[n] - \sum_{l=0}^{L-1} c_{k,l}^I\, \psi_{k,l}^I[n]\right),\ n \in J_N\,. \tag{1}$$

$$\psi_{k,l}^R[n] = g\left[\left(n - lM\right)_{\mathrm{mod}\,N}\right]\exp\left(j\frac{2\pi}{M}k\left(n-\alpha/2\right)\right), \tag{2}$$

$$\psi_{k,l}^I[n] = -jg\left[\left(n + M/2 - lM\right)_{\mathrm{mod}\,N}\right]\exp\left(j\frac{2\pi}{M}k\left(n-\alpha/2\right)\right), \tag{3}$$

$$\mathcal{B}[J_N] \overset{def}{=} \left\{\psi_{k,l}^R[n], \psi_{k,l}^I[n]\right\}, \tag{4}$$

where $c_{k,l}^R = \mathrm{Re}(a_{k,l})$ and $c_{k,l}^I = \mathrm{Im}(a_{k,l})$ are real and imaginary parts of complex information QAM symbols $a_{k,l}$ used in OFDM; $s[n] = s(nT/M)$; $g[n] = g[nT/M]$ and $g[n + M/2]$ - initializing functions; $J_N = \{0,1,...,N-1\}$; $\alpha$ - phase parameter.

The system of basic functions $\mathcal{B}[J_N]$ is orthogonal in terms of real scalar product defined on the Hilbert space of discrete functions on $J_N$

$$\left\langle x[n], y[n] \right\rangle_R = \mathrm{Re} \sum_{n=0}^{N-1} x[n] \cdot y^*[n], \tag{5}$$

where $*$ is the sign of complex conjugation.

Matrix representation of basis (4) $\mathbf{U} = (\mathbf{U}_R, \mathbf{U}_I)$ is a $N \times 2N$ block matrix with blocks $\mathbf{U}_R$, $\mathbf{U}_I$ - square $N \times N$ matrixes constructed from columns of basis' functions $\vec{\psi}^R_{k,l} = \left(\psi^R_{k,l}[0], ..., \psi^R_{k,l}[N-1]\right)^T$ and $\vec{\psi}^I_{k,l} = \left(\psi^I_{k,l}[0], ..., \psi^I_{k,l}[N-1]\right)^T$ for all indexes $k = 0, ..., M-1$, $l = 0, ..., L-1$. This presentation makes easier theoretical investigation of the basis. In particular, orthogonality condition in matrix form is

$$\mathrm{Re}\left(\mathbf{U}^* \mathbf{U}\right) = \mathbf{I}_{2N}, \tag{6}$$

where $\mathbf{I}_{2N}$ is $2N \times 2N$ identity matrix; signal modulation and demodulation will look like

$$\vec{S}^T = \mathbf{U}\vec{C}^T; \quad \vec{C}^T = \mathrm{Re}\left\{\mathbf{U}^* \vec{S}^T\right\}. \tag{7}$$

It is necessary to mention that Weyl-Heisenberg basis constructed from rectangular functions is orthogonal in time domain only because these functions do not overlap. Thus this orthogonality is artificial in some sense. In real dispersive channel consecutive OFDM symbols will overlap and that is why it is impossible to refuse from cyclic prefix. In OFTDM this problem does not exits ever more. Basis functions can be overlapped not only in frequency but also in time domain if orthogonality conditions (5) or (7) are fulfilled. Time-frequency structure of OFTDM signal is described on Fig-1. Thus CP can be used less often mainly to divide OFTDM symbols and for synchronization purpose.

In addition to that generalized Weyl-Heisenberg bases can have much better localization in frequency domain without the loss in spectral efficiency. Initializing functions $g[n]$ in (2), (3) can be selected in such a way that the matrix $\mathbf{U}_{opt}$ of the basis will minimize the following functional on the space of orthogonal matrixes: A

$$\mathbf{U}_{opt} : \min_{\mathbf{U} \in \mathfrak{A}} \left\| \mathbf{G} - \mathbf{U} \right\|_E^2, \tag{8}$$

where $\mathbf{G}$ is the matrix of some nonorthogonal basis with desired localization characteristics; $\|\mathbf{A}\|_E^2 = \mathrm{tr}(\mathbf{A}\mathbf{A}^*)$ is a Frobenius norm. The quality of localization can be estimated, for example, with the help of ambiguity function:

$$A(\tau, \upsilon) \triangleq \sum_{n=0}^{N-1} g[n]\, g^*\left[\left(n+\tau\right)_N\right] \exp\left(-j\frac{2\pi}{N}\upsilon n\right). \tag{9}$$

In general case any function with necessary localization properties can be used as an impute to problem (8). In particular, the ambiguity function of orthogonal generalized Weil-Heisenberg basis constructed from Gaussian function is presented on Fig. 2. The size of side lobes is very small in compare to the main lobe.

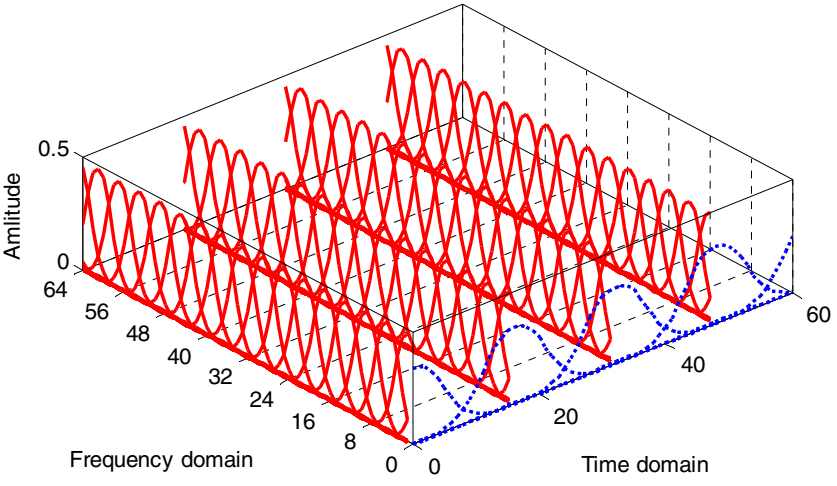



**Fig. 1.** Time-frequency structure of generalized Weyl-Heisenberg basis. Basis functions in time domain (dotted lines) and in frequency domain (solid lines) are shown.

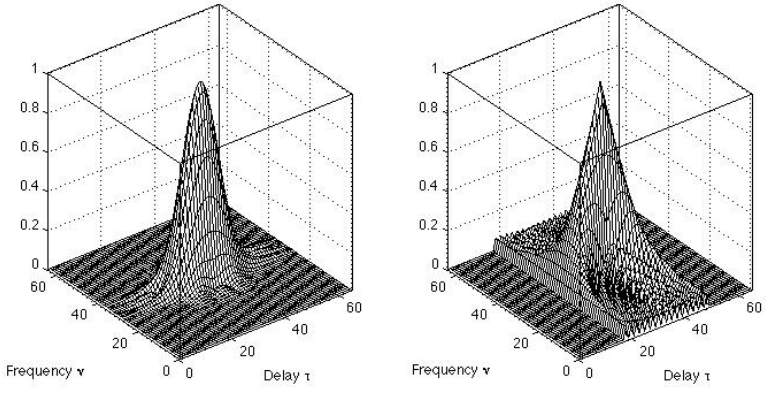



**Fig 2.** Module of ambiguity function of initializing function of generalized Weyl-Heisenberg function constructed from Gaussian function (left) and rectangular impulse (write)

Presented figures show two main advantages of proposed technique: firstly it is denser packing of the signal not only in frequency but also in time domain (Fig. 1) and, secondly, better localization in frequency domain (Fig. 2).

## 4 Conclusions

Of cause more simulation results are still required to justify the advantage of OFTDM scheme over classical OFDM. In particular in work [9] it has been already shown that

OFTDM signals have higher robustness against Doppler shift. Nevertheless the model used in that paper was rather simple and didn't include such important blocks as coding, interleaving, etc. Thus our future work includes such necessary and logical steps as:

- Extension of WiMAX link level model with OFTDM modulator.
- Utilization of link level model output in system level simulator [12].

In the conclusion it is important to formulate expected advantages of OFTDM technology:

- Application of OFTDM instead of OFDM on physical layer in channels with time-frequency dispersion improves spectral and energy efficiency. This effect is achieved because of additional intersymbol multiplexing used in OFTDM.
- In OFTDM signals the level of out of band emission is lower. Thus the requirements on the quality of transmitter's filter and on the guard intervals on the edges of the frequency range can be weaken.
- It is possible to improve the robustness of the system against ICI an ISI and to adopt better to the parameters of time-frequency dispersion.

In addition to factors mentioned above several economic benefits can be mentioned:

- One of the main effects of proposed approach is better interference resistance of the system and thus better reception quality of mobile users. It means that in equal conditions guaranteed quality will be achieved for the lower value of signal to interference and noise ratio (SINR). As a result less number of base stations is required in a given service area.
- Bad reception quality results in the necessity to use lower modulation indexes. Throughput and the number serviced subscribers go down. Thus in the same conditions OFTDM technology makes it possible to transmit data to larger number of users in compare to OFDM realisation. Increase in the maximum number of users with the constant number of base stations gives direct increase of profits.
- Improvement of the quality of services is the important factor in competitive struggle. It stimulates the demand for the new devices with better performance based in particular on proposed OFDTM technology.

## References

1. Proakis, J.G.: Digital Communications, 4th edn. McGraw Hill, New York (2000)
2. Oppenheim, A.V., Shafer, R.W.: Discrete-Time Signal Processing, PHI, 2/E (2000)
3. Dahlman, E., Parkvall, S., Skold, J., Berming, P.: 3G evolution: HSPA and LTE for mobile broadband. Academic Press, London (2008)
4. Keller, T., Hanzo, L.: Adaptive modulation techniques for duplex OFDM transmission. IEEE Transactions on Vehicular Technology 49(5), 1893–1906 (2000)
5. Zang, Z., Lai, L.: A novel OFDM transmission scheme with length-adaptive Cyclic Prefix. Journal of Zhejang University – Science A 5(11), 1336–1342 (2004)
6. Kravchenko, V.F., Rvachev, V.L.: Boolean algebra, atomic functions and wavelets in

7. Kravchenko, V.F., Churikov, D.V.: Kravchenko–Kotelnikov–Levitan–Wigner distributions in radio physical applications, in Days on diffraction 2008, pp. 79–84. St. Petersburg, Russia (2008)
8. Volchkov, V.P., Petrov, D.A.: Orthogonal, well-localized Weyl-Heisenberg basis construction and optimization for multicarrier digital communication systems. In: Proceedings of International Conference on Ultra Modern Telecommunications (ICUMT 2009). St. Petersburgh, Russia (2009)
9. Petrov, D.A.: Well-localized bases construction algorithms. Mathematical Models and Computer Simulations 2(5), 574–581 (2010)
10. Petrov, D.A., Hamalainen, T.: Efficient WH-OFTDM Signals Processing. In: Proceedings of International Conference on Ultra Modern Telecommunications (ICUMT 2010), Moscow, Russia (2010)
11. Mallat, S.: A wavelet tour of signal processing, 2nd edn. Academic Press, London (1999)
12. Le Floch, B., Alard, M., Berrou, C.: Coded Orthogonal Frequency Division Multiplex. Proc. of the IEEE 83(6), 982–986 (1995)
13. Sayenko, A., et al.: WINSE: WiMAX NS-2 extension (2010), http://sim.sagepub.com/content/early/2010/07/27/0037549710373334.full.pdf

# Analysing and Improving Energy Efficiency of Distributed Slotted Aloha

Haidi Yue, Henrik Bohnenkamp, Malte Kampschulte,
and Joost-Pieter Katoen

Software Modeling & Verification
Dept. of Computer Science 2, RWTH Aachen University, Aachen, Germany
{haidi.yue,henrik,Malte.Kampschulte,katoen}@cs.rwth-aachen.de

**Abstract.** This paper is concerned with the formal modelling and simulative analysis of an energy-efficient MAC protocol for gossip-based wireless sensor networks. This protocol is a variant of classical slotted Aloha in which the number of active TDMA slots is dynamically changed depending on the number of neighbours of a node. We provide a formal model of this protocol, and analyse energy consumption under the signal-to-interference plus noise ratio (SINR) radio model. We propose an amendment of the distributed slotted Aloha protocol by a simple dynamic power assignment scheme, and show that this significantly reduces the energy consumption (30%) and speeds up the message transmission.

**Keywords:** WSNs, Aloha, formal modelling, SINR, simulation, energy.

## 1 Introduction

We consider the setting of gossip-based wireless sensor networks (WSNs), in which battery-powered mobile sensors interact via a wireless communication network. The sensors are extremely simple, cheap to produce, and have limited data and processing capacities. The system lifetime is mainly determined by the battery lifetime, as recharging is typically not possible. To support the reliable communication between mobile sensors a medium access protocol (MAC) is needed. In order to extend the lifetime of a sensor, the protocol must be energy-efficient – it should attempt to idle once in a while, not only to save power, but also in order to profit from the so-called recovery effect [10]. To meet all these requirements, the Dutch company CHESS, experts on developing gossiping WSNs for various applications, has developed a variant of slotted Aloha, called distributed slotted Aloha. This protocol aims to significantly simplify previously proposed energy efficient MAC proposals such as gMAC [15], A-Mac [12] and the TDMA-W [6] protocols.

In contrast to the well-known classical slotted Aloha protocol [9], a node in distributed slotted Aloha has a *dynamic* number of active slots. Each node keeps track of a list of neighbour nodes, and adapts the number of its active slots when its neighbourhood increases or decreases. On the one hand, this is aimed to maximise throughput; on the other hand it aims to adapt the idle period of a

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 197–208, 2011.

frame in order to reduce energy consumption. The protocol is relatively simple to realise, and has moderate memory usage. The parameters of the protocol are tuned to extensive experiments in real networks. The aim of our study is to provide an abstract and easy-to-grasp formal model of the protocol, and analyse its energy-preservation capabilities.

In distributed slotted Aloha, each node has a constant signal power during its lifetime, and this is equal for all nodes. As signal power is a critical parameter in our studies, we use the signal-to-interference plus noise ratio (SINR) radio model as radio propagation model. The SINR model [11] is an intuitive model in which there is a relation between signal strength and distance, and the receipt of a message depends on the strength of interfering transmissions. In the SINR model, a node can decode a message only if the received signal strength divided by the strength of concurrent interfering senders (plus the noise) exceeds a threshold. This is more realistic compared to the unit disk model [15].

We model the distributed slotted Aloha protocol in the MoDeST language [3], a formalism that supports the modular specification of distributed systems in a mathematically rigorous, though user-friendly, manner. The simulation is performed using the Möbius [7] tool-set. The main advantage over the usage of standard simulation packages such as NS2, Opnet or OMNET, is that we obtain semantically sound simulation runs. Together with the fact that we do not model entire protocol stacks but rather abstract from lower layer effects, this avoids many of the credibility problems of standard simulations [5,1].

An important outcome of our analysis studies is that the constant and identical signal strength may result in a huge interference in densely populated areas, whereas nodes become disconnected in sparse areas. To overcome these unfavourable situations, we propose an amendment of the distributed slotted Aloha protocol in which nodes can adapt their signal power strength dynamically based on the number of neighbours of a node. In sparse areas, the strength is increased, whereas in dense areas, it is reduced. We show that this scheme substantially reduces energy consumption and improves message propagation.

Organisation of the paper. Section 2 introduces the distributed slotted Aloha protocol investigated in this paper. Section 3 explains the SINR radio model. Section 4 describes our modelling idea and the simulation setup. Section 5 contains results and Section 6 concludes.

## 2 Distributed Slotted Aloha

The well-known slotted Aloha protocol [9,13] has a fixed number of slots, and has the property that its throughput reaches the maximum when a node has the same number of neighbours as the number of the TDMA slots. The distributed slotted Aloha (dsA for short) strategy is an extension of classical slotted Aloha, uses the same principle but has a dynamic behaviour depending on the number of direct neighbours. When the number of neighbours increases (or decreases), a node will increase (or decrease) its number of active slots, to make the number of neighbours and the number of the current TDMA slots more or less equal, and hence attempts to achieves the maximum throughput as close as possible.

In the dsA protocol, a *frame* is the basic unit of time (Figure 1). A frame is subdivided into an active and idle period. Radio communication occurs in the active period and the idle period is considered for energy conservation. The active period is divided in slots of equal length, we call these slots *virtual slots*. The virtual slots are merged into several blocks of an equal number of slots, each block is called a *schedule*. A schedule can be either active or passive. Slots in a passive schedule are considered as idle slots. The active schedules are at the begining of a frame, virtual slots in active schedules are called *active slots*. At the begin of each frame, every node randomly chooses a sending slot (the so called TX-slot) from all its active slots. The receive action occurs also in active slots, however, not among all the active slots, but in a certain schedule. We will explain later how it works. In the following, we denote by $S$ the number of active schedules.

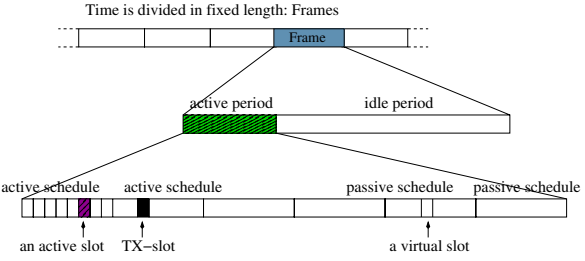



**Fig. 1.** Basic structure of distributed slotted Aloha

The distributed slotted Aloha algorithm has a fixed number of virtual slots per frame. In the current implementation at CHESS, this parameter is set to 80, and each schedule contains 8 virtual slots, i.e. there are 10 schedules in total. A node with $S$ ($1 \leq S \leq 10$) active schedules has hence $8 \cdot S$ active slots, which are settled at the beginning of a frame. Per frame, a node will send in one of the active slot, and receive in one of the $S$ active schedules.

Assume a node has 3 active schedules (Figure 2), i.e., it has 24 active slots. Since a node can receive only in one active schedule per frame, if it received in the first active schedule at the $n$th frame, then in the $(n+1)$th frame, it receives in the second active schedule, in the $(n+2)$th frame, in the third active schedule, and in the $(n+3)$th frame, it receives again in the first active schedule, and so on and so forth. But no matter in which active schedule it receives, it chooses randomly a send slot from all the active slots. In general, a node with $S$ active schedules needs $S$ frames to complete a *receiving cycle* over all $8 * S$ active slots. Active slots that are neither used to receive nor to send are considered idle.

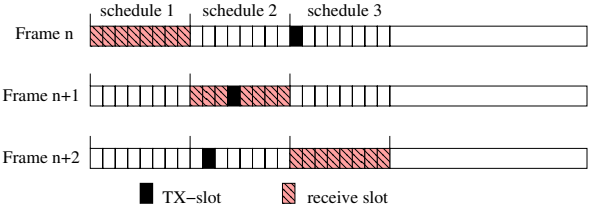



**Fig. 2.** Distributed slotted Aloha example of a node with 3 active schedules

The number of active schedules of a node is dynamic and depending on the number of neighbours. To achieve the dynamic management of its

neighbourhood, each node has an unique Id, and maintains a list containing the Ids of its neighbours. Each message contains the sender's Id. On receiving a message, if the sender is not yet in the neighbourhood list, it will be added to the list and the number of neighbours $b$ will be increased by one. Furthermore:

1. At the end of each frame, every node checks its neighbourhood list. If there are some nodes that have not seen for 49 frames, remove those nodes from the list. For each removed node, $b$ is reduced by one.
2. At the end of a receiving cycle, each node eventually updates $S$ by the following rules:

   – if $(8 \cdot S < 2 \cdot b)$ and $S < 10 \rightarrow S+=1$
   – if $(8 \cdot S > 2 \cdot b)$ and $S > 1 \rightarrow S-=1$.

   Intuitively speaking, if the number of active slots $(8 \cdot S)$ of a node is not sufficient to receive all its neighbours $(2 \cdot b)$ and the maximal number $S$ is not yet reached, increases $S$ by one, so that eventually more neighbours can be found. Otherwise, reduce $S$ by one. If $8 \cdot S$ is equal to $2 \cdot b$, let $S$ be unchanged. The factor 2 for $b$ comes from the consideration that not all messages from the neighbourhood can be received, and nodes assume that the actual number of neighbours is approximately two times the number of nodes in their neighbourhood list.

The dsA is basically a random access protocol with dynamic active slots. The idea behind this dynamic scheduling is to keep the real active phase of a node as short as possible, but obtain the same effect as if there would be more active slots. Assume a node has 24 one-hop neighbours, in a random medium access protocol with a fixed number of active slots, at least 24 receive slots per frame are necessary so that the node can get messages from every neighbour. This costs per frame three times more energy, than if there would be 8 receive slots per frame, and the messages are received over 3 frames. In other words, the dsA strategy delays the message transmission with the gain of longer node energy preservation and hence extends the network life. Later we compare the dsA strategy against the random access protocol with a fixed number of active slots (simple slotted Aloha, ssA for short), and show that it is indeed the case that dsA consumes much less energy (58.75%)to finish the all-to-all message communication, at the cost of slower propagation speed (2.5 times longer).

The dsA protocol uses constant and equal signal power for all nodes. The transmission power of nodes should be high enough to reach the intended receiver while causing minimal interference at other nodes. Hence the most critical parameter of this protocol is the value of the power. We choose SINR [11] as our radio propagation model which is more realistic than the unit disk model [15]. With this physical model, we first estimate the effectiveness of dsA. Later, we modify the protocol by letting nodes be able to regulate their signal power dynamically. The way how a node dynamically manages its signal power, and a comparison between constant power vs. dynamic power are presented in the section on results.

## 3 SINR

Receiving a message in a wireless context mainly depends on two things: the distance from which signals come, and the power with which they are transmitted. Those two variables form the basis of the SINR radio model [11].

Let $p_i$ be the sending power of node $i$ and $x_i$ its position. Then the relative signal strength of a message from node $i$ at node $j$ ($j \neq i$) is determined by

$$r_i(x_j) := \frac{p_i}{d(x_i, x_j)^\alpha} \tag{1}$$

where $d(x_i, x_j)$ is the distance between $x_i$ and $x_j$ and $\alpha$ the path loss exponent, which determines the power loss over distance. $r_i(x_j) = 0$ if $i$ is not sending. Depending on the environment, it is usually assumed that $\alpha$ has a value between 2 and 5 [8]. In an ideal vacuum, we have $\alpha = 2$ and we use this value for our experiments because it provides the upper bound of $r_i(x_j)$ with fixed $p$.

A signal can only be received if its relative signal strength is significantly higher than the strength of all other signals (for example, signals from other sending nodes) combined. This allows us to formulate a noninterference condition. Node $j$ will receive a signal from node $k$ if

$$r_k(x_j) > \beta(\sum_i r_i(x_j) + \nu(x_j)) \tag{2}$$

where $\beta$ determines the minimal share of the whole signal which is needed for a successful transmission and $\nu(x_j)$ is the background noise at node $j$. The value of $\beta$ must be between 0.5 and 1.

## 4 Modelling and Simulation Details

The modelling of the protocol described above is done in MoDeST, the "MOdeling and DEscription language for Stochastic and Timed systems" [3]. It allows us to describe the behaviour of discrete event systems in a compositional manner. As the possible slot assignments at each frame are of order $|slots|^{|nodes|}$, the situation we want to analyse are too complex to be solved in a formal way. Therefore, we simulate them and average the results. For this we will use Möbius [7], an integrated tool environment for the analysis of performability and dependability models. The conjunction between MoDeST and Möbius is realised by MOTOR [2,4], a tool that is integrated into the Möbius framework and aims to facilitate the transformation and analysis of MoDeST models.

**Modelling assumptions.** Per receive slot, a node can receive at most one message. Collisions occur when more than one node is sending to the same node at the same time. We assume there are no message loss during propagation, hence the only reason that a node received nothing is due to a collision.

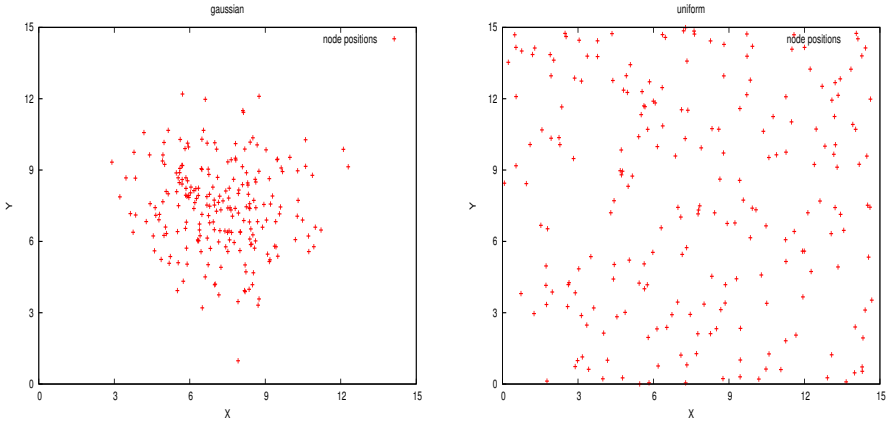
Distributed slotted Aloha incorporates a mechanism to synchronise clocks of neighbouring nodes. And with MoDeST we can treat every node as a single process equipped with a different clock. However, since clock synchronisation is not the focus of this paper, we assume in our model that all nodes are perfectly synchronised to each other.

Important simulation parameters for the experiments are explained below.

### 4.1 Node Arrangement

In order to investigate the protocol behaviours under distinct network connectivity conditions, we consider the following topologies.

1. Grid network of size $15 \times 15$ with 225 nodes, each node is placed at the vertex of the grid. It is large enough to create most of the interesting situations and at the same time small enough to become no burden on our computation time.
2. We generate 225 points inside a $15 \times 15$ area around the point $(7.5, 7.5)$, so that the choice of the coordinates of those points is governed by a Gaussian distribution (see Figure 3(a)). This kind of node arrangement builds a much more clustered structure comparing to the centre-less grid network.




(a) Gaussian (b) uniform

**Fig. 3.** Two random networks

3. The last graph represents a uniform distribution, i.e. both $x$ and $y$ coordinates of the 225 nodes are generated by a uniform distribution from 0 to 15. A typical graph generated by this setting is shown in Figure 3(b).

### 4.2 Simulation Parameters and Return Data

As the basic purpose of any network is to transfer information, we focus our analysis on measure its ability and particularly the cost for this transfer. The cost parameters of interest in our study are time and energy consumption. In the experiments, we consider the all-to-all communication, and measure the energy consumed by the whole network. We equip each node with a distinct flag. Initially, every node only has its own flag set. Every time when node $j$ receives a message from node $k$, all flags of node $k$ are set for node $j$. All networks considered in this paper consist of 225 nodes, which means there are $225 \times 225 - 225 = 50400$ new flags to set. So the return data of the experiment

are the energy consumption $E$ by the whole network when $m$ ($m \leq 50400$) flags are set. We consider the propagation task to be successfully finished if 99.9% of the 50400 new flags are set. Since the number of maximal possible active slots in the dsA protocol is 80, we vary the number of active slots in the ssA from 16 to 80.

Two important parameters in the SINR model are the signal power $p$ and the background noise $\nu$. We first observe from the in-equation (2) for the radio model that it suffices to vary one of these parameters only. Increasing the background noise by a factor $\eta$ has the same effect as decreasing the power of all senders by a factor $\frac{1}{\eta}$, as has the scaling of all distances by $\eta^2$. So to keep things simple, we fix the background noise $\nu$ and vary the signal power $p$.

To determine the range of values of $\nu$ that are of interest, we consider of which potentially effective transmission distances are reasonable, or in other words, how many neighbours on average a node should have, so that we can examine both sparse and dense networks. Considering the grid network, if we choose the maximum possible range for a message to be transmitted be 2.1, 4.1 and 6.1, each node has 12, 48 and 112 direct neighbours, respectively. Now we can deal with the choice of background noise. We determin it by consider the case for a single sender in the noninterference condition, as explained in the following.

Recall that the physical model declares a message from node $k$ to node $j$ as received if (2) holds. Under the ideal case that the only node sending was $k$, this inequality simplifies to $r_k(x_j) > \beta(r_k(x_j)+\nu(x_j))$. We know that the strength of the received signal from node $k$ at node $j$ is given by (1). Using this and solving for the distance $d(x_k, x_j)$ we get

$$d(x_k, x_j) = \sqrt[\alpha]{\frac{p_k(1-\beta)}{\beta\nu}}$$

as the maximum distance. Since there are always some other nodes interfering with $k$, we can only view this as an upper bound.

Using this equation, if we choose $p = 10$, $\alpha = 2$ and $\beta = 0.7$, we get noise levels 0.971, 0.255 and 0.115, respectively, which loosely approximate the range 2.1, 4.1 and 6.1 mentioned before. Accurate values for $\alpha$ and $\beta$ are not relevant, since from the above equation, we can always manipulate the signal power $p$ and obtain an equivalent result of distance.

**Table 1.** Energy demands of the nRF24L01 radio

| mode | current |
|---|---|
| transmit | 11.3 mA |
| receive | 12.3 mA |
| idle | 0.9 $\mu$A |

The sensor nodes developed by CHESS are equipped with an ATMega64 micro-controller and a Nordic nRF24L01 [14] packet radio. The energy demands of the Nordic nRF24L01 radio are summarised in Table 1. In the MoDeST model, we use the multiplication of those real values and the signal power $p$ as the energy consumption in each send, receive, and idle slot. All simulations ran at least 500 times. The confidence level of all simulations is set to 0.95 and the relative confidence interval is 0.1.

## 5 Results

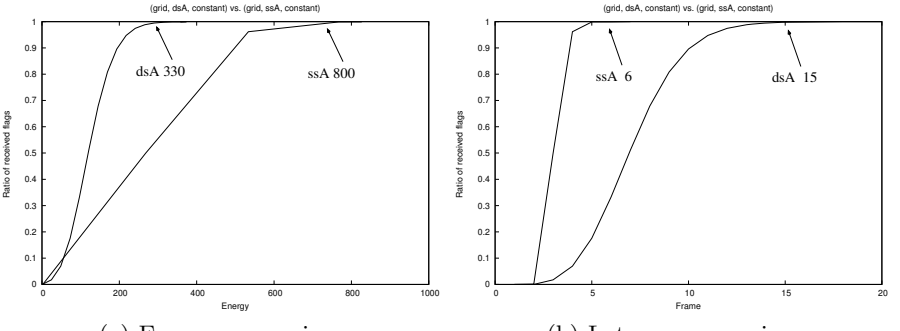
In this section we present the results of our analysis of the two protocols ssA and dsA, in the different topologies grid, Gaussian and uniform. Furthermore we investigate the influence of dynamic send-power management in dsA. We denote these different network configurations as a triple (*network*, *protocol*, *power*), where *network* can either be grid, Gaussian or uniform; *protocol* can be dsA or ssA; and *power* is either constant or dynamic.

### 5.1 DsA vs. ssA

To compare dsA and ssA, we choose $p = 15$, $\nu = 0.255$ and vary the number of active slots in configuration (grid, ssA, constant) from 16 to 80 in steps of 16. As we can see in Figure 4(a), the energy consumption is much lower in (grid, dsA, constant) than (grid, ssA, constant). When the propagation finishes, (grid, dsA, constant) consumes on average 330 energy units, whereas the best result in (grid, ssA, constant) consumes 800 energy units.

This means a 58.75% of energy saving with dsA. Thus, dsA saves energy, but as a penalty, throughput is slower. This effect is illustrated in Figure 4(b). There we see the fraction of flags distributed through the network vs. the time it took: with dsA, 15 frames, with ssA, just 6.

The same experiments, repeated with various other power parameters, show similar results: dsA propagates messages slower than ssA, but with less energy.




(a) Energy comparison (b) Latency comparison

**Fig. 4.** dsA vs. ssA

We repeat the experiment for Gaussian and uniform networks and the results show again that dsA propagates messages slower than ssA, but cost less energy.

### 5.2 Optimal Signal Power

In this section we investigate whether there is an optimal power level in terms of message propagation speed (thus, also energy).

To that end, we investigate the all-to-all message communication in (grid, dsA, constant), (Gaussian, dsA, constant) and (uniform, dsA, constant). We set again the background noise $\nu = 0.255$ for all the experiments and vary the signal power $p$. The results of the simulations for the Gaussian network are shown in Figure 5 (Note that the plot is focused to the $95\% - 100\%$ interval. Otherwise the curves would be too hard to distinguish). The curve for $p = 6$ represents the most efficient result, because it reaches the 99.9% mark first. For lack of space, the graphs for the grid and uniform network are not shown. All networks have in common that there exists an optimal signal power for each network. Any other values of $p$ that are larger or smaller than this value yield a greater energy consumption when the all-to-all communication is done. Where they differ is the optimal value of signal power, and the total energy consumed by the network for this power.

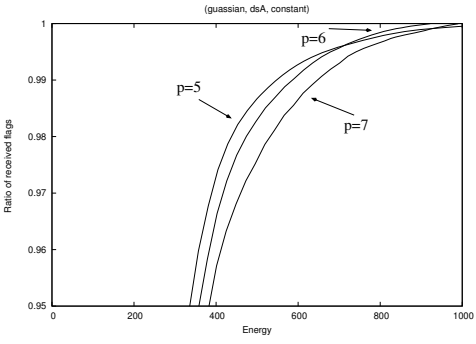



**Fig. 5.** Optimal transmission power for (Gaussian, dsA, constant)

We can summarise that for the (Gaussian, dsA, constant) scenario, the optimal power is 6, whereas in (grid, dsA, constant) and (uniform, dsA, constant), the value is 15 and 16, respectively. Our explanation why the power in (Gaussian, dsA, constant) is smaller is that nodes are more densely distributed. The average distances between nodes in grid and uniform are larger, requiring higher send power. This also explains why the optimal values of signal power for these two node arrangements are similar.

### 5.3 Dynamic Power Management

**Introduction.** Given a certain transmission power $p$ and background noise $\nu$, the number of nodes inside the maximal reachable transmission range of each node in the network is usually individual (except in the grid network). This means the dsA strategy with the identical transmission power for all nodes may cause a lot of interference in dense clusters of the network, which in sum might even interfere with nodes far away outside the cluster. In sparse areas in the network, on the other hand, nodes may disconnect from the network due to the interference, and need to send with higher power to make themselves heard.

One idea to deal with these two unfavourable situations is to vary the transmission power of nodes dynamically, so that nodes in dense areas are able to lower send power to reduce interference, and nodes in sparse areas can increase send power to maintain network connectivity.

This forms the basic idea of dynamic power management in dsA. There are two questions to answer: first, how to determine whether a node is in a dense or a sparse area of the network; second, how to adjust the power levels. The

first question we answer as pragmatically: we use the neighbourhood list that is already present in the dsA protocol and define a low- and high-water-mark $L$ and $H$ which, when tripped, cause an increase or decrease in power. The answer to the second question is inspired by the Nordic nRF24L01 radio. This radio has for different power levels of 0dBm, -6dBm, -12dBm, and -18dBm, *i.e.*,full power, and 1/4, 1/16 and 1/64 power. The radio has thus an exponential decrease of send power with each level.

The mechanism we use to adjust the power is then the following: for node $i$ with size of neighbourhood list $b_i$ and current power level $p_i$, at the beginning of each frame: if $b_i > H$, then $p_i' = p_i/n$; if $b_i < L$, then $p_i' = np_i$, where $n \in \{2, 3, 4, 5, \ldots\}$ the factor to increase or decrease the power and $p_i'$ the new power level.

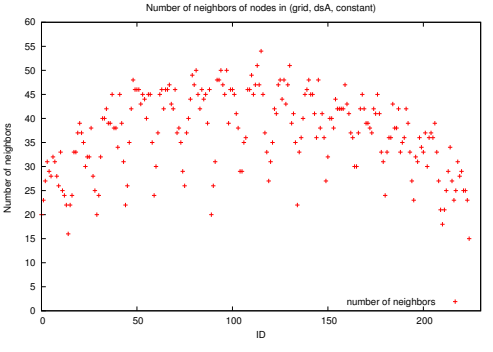
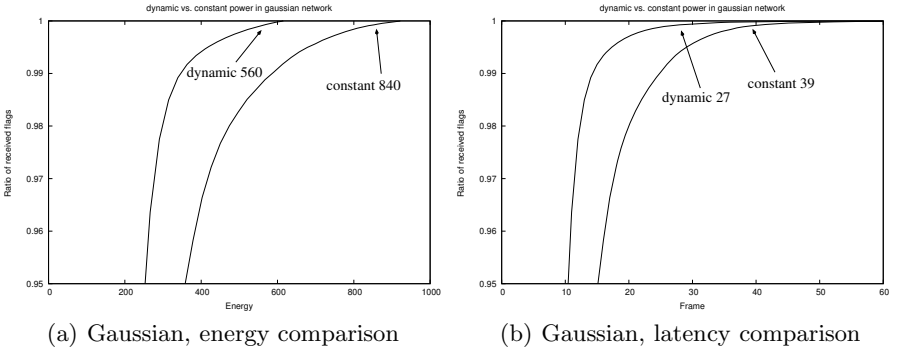



**Fig. 6.** Number of neighbours of all 225 nodes in (grid, dsA, constant) with $p = 15$ and $\nu = 0.255$

The question is how to determine $H$ and $L$. For this, consider Figure 6, which shows the number of neighbours of all the 225 nodes in (grid, dsA, constant) with background noise $\nu = 0.255$ and signal power $p = 15$, at the end of the 500th frame. Note that this $p$ is the optimal value of signal power for (grid, dsA, constant), as determined in Section 5.2. As we can see, most of the nodes in Figure 6 have between 20 and 50 neighbours, so we use these two values as the lower and upper bounds in dynamic power management.




(a) Gaussian, energy comparison (b) Gaussian, latency comparison

**Fig. 7.** Comparison of constant and dynamic power assignment in Gaussian network

**Experiments and Results.** We are again interested in the speed of flag propagation in the whole network. The results for the uniform and grid network topology show no influence of dynamic power management on the results compared to the ones in Section 5.2. We explain this with the relatively uniform structure of the topologies, where no area is much denser than the other.

**Table 2.** Output power setting for the nRF24L01

| output power | current |
|---|---|
| 0 dBm | 11.3 mA |
| −6 dBm | 9.0 mA |
| −12 dBm | 7.5 mA |
| −18 dBm | 7.0 mA |

We thus concentrate on the Gaussian case, *i.e.,*(Gaussian, dsA, dynamic). We set $\nu = 0.255$ and $p = 6$ as the maximum power level (which is the optimal value determined in Section 5.2). As the energy consumption of the radio we use Table 2 of the Nordic radio.

We conducted two different experiments for two different values of $n$, the factor used in the power adaption: $n = 2$ and $n = 4$. The case $n = 4$ thus mimics very closely the power levels of the Nordic radio. Interestingly, the result shows no improvement over dsA without power-management. However, with $n = 2$, it does, as is shown in Figure 7. In Figure 7(a), we compare the energy consumption between (Gaussian, dsA, constant) and (Gaussian, dsA, dynamic). Obviously, with the same initial power value, (Gaussian, dsA, dynamic) consumes 30% less energy than (Gaussian, dsA, constant). The results with other $p$ are similar and not shown here. In Figure 7(b), we see that, the dynamic power management not only reduces energy consumption but also accelerates propagation speed.

Apparently the choice of the factor $n$ is important to affect an improvement in the energy consumption and latency. The conditions under which such improvement can be achieved requires more investigation and is subject of our future work.

## 6 Conclusions and Future work

In this paper, we reported on the simulative analysis of the distributed slotted Aloha protocol, aimed for gossiping-based application in sensor networks. Our analysis shows that, comparing to a simple slotted Aloha strategy with a fixed number of active slots, the distributed slotted Aloha with dynamic number of active slots significantly reduces the energy consumption (almost 60%) for all-to-all communication, although throughput decreases. We showed that the optimal transmission power is different from network to network, which indicates a necessity of dynamic power management. We proposed a modification of the distributed slotted Aloha protocol by a simple dynamic power assignment scheme, and show that this not only reduces the energy consumption (30%) but also speeds up the message propagation.

Our future investigation will focus on different schemes for power-management and on determining the circumstances which make dynamic power management effective.

**Acknowledgement.** We thanks Bert Bos, Frits van der Wateren, Marcel Verhoef (all of CHESS) for discussions on the protocol and suggestions on the modeling. This research has been founded by the EU under grant number FP7-ICT-2007-1 (Guasimodo) and the DFG Excellence Cluster UMIC.

## References

1. Andel, T.R., Yasinac, A.: On the credibility of MANET simulations. IEEE Computer 39(7), 48–54 (2006)
2. Bohnenkamp, H., Courtney, T., Daly, D., Derisavi, S., Hermanns, H., Katoen, J.-P., Vi Lam, V., Sanders, B.: On integrating the Möbius and MoDeST modeling tools. In: Proceedings of the 2003 International Conference on Dependable Systems and Networks, vol. 4424. IEEE Computer Society Press, Los Alamitos (2003)
3. Bohnenkamp, H., D'Argenio, P.R., Hermanns, H., Katoen, J.-P.: MoDeST: A compositional modeling formalism for real-time and stochastic systems. IEEE Trans. on Software Engineering 32(10), 812–830 (2006)
4. Bohnenkamp, H., Hermanns, H., Katoen, J.-P.: Motor: The MoDeST tool environment. In: Grumberg, O., Huth, M. (eds.) TACAS 2007. LNCS, vol. 4424, pp. 500–504. Springer, Heidelberg (2007)
5. Cavin, D., Sasson, Y., Schiper, A.: On the accuracy of MANET simulators. In: ACM Workshop on Principles of Mobile Computing, pp. 38–43 (2002)
6. Chen, Z., Khokhar, A.: Self organization and energy efficient TDMA MAC protocol by wake up for wireless sensor networks. In: Sensor and Ad Hoc Communications and Networks, pp. 335–341 (2004)
7. Deavours, D.D., Clark, G., Courtney, T., Daly, D., Derisavi, S., Doyle, J.M., Sanders, W.H., Webster, P.G.: The Möbius framework and its implementation. IEEE Trans. on Software Engineering 28(10), 956–969 (2002)
8. Fanghänel, A., Kesselheim, T., Räcke, H., Vöcking, B.: Oblivious interference scheduling. In: ACM Symposium on Principles of Distributed Computing (PODC), pp. 220–229 (2009)
9. Hammond, J.L., O'Reilly, P.J.P.: Performance Analysis of Local Computer Networks. Addison-Wesley, Reading (1986)
10. Jongerden, M.R., Mereacre, A., Bohnenkamp, H., Haverkort, B.R., Katoen, J.-P.: Computing optimal schedules for battery usage in embedded systems. IEEE Trans. Industrial Informatics 6, 276–286 (2010)
11. Gupta, P., Kumar, P.R.: The capacity of wireless networks. IEEE Transactions on Information Theory 46, 338–404 (2000)
12. Rashid, R.A., Mohd, W., Ehsan, A., Embong, W., Zaharim, A., Fisal, N.: Development of energy aware TDMA-Based MAC protocol for wireless sensor network system. European J. of Scientific Research, 571–578 (2009)
13. Roberts, L.G.: Aloha packet system with and without slots and capture. ACM SIGCOMM Computer Communication Review 5, 28–42 (1975)
14. Nordic Semiconductors. nRF2401 Single-chip 2.4GHz Transceiver Data Sheet (2002)
15. Yue, H., Bohnenkamp, H., Katoen, J.-P.: Analyzing energy consumption in a gossiping MAC protocol. In: Müller-Clostermann, B., Echtle, K., Rathgeb, E.P. (eds.) MMB&DFT 2010. LNCS, vol. 5987, pp. 107–119. Springer, Heidelberg (2010)

# A Joint Power and Rate Adaptation Scheme in Multicarrier DS/CDMA Communications over Rayleigh Fading Channels

Ye Hoon Lee[1,⋆] and Nam-Soo Kim[2]

[1] Department of Electronic and Information Engineering,
Seoul National University of Science and Technology, Seoul 139-743, South Korea
y.lee@snut.ac.kr
[2] Department of Computer and Communication Engineering, Chongju University,
Chongju, 360-764, South Korea
nskim@chongju.ac.kr

**Abstract.** We present a joint power and rate adaptation scheme in multicarrier (MC) direct-sequence code-division multiple-access (DS/CDMA) communications under the assumption that channel state information is provided at both the transmitter and the receiver. We consider, as a power allocation strategy in the frequency domain, transmitting each user's DS waveforms over the user's sub-band with the largest channel gain, rather than transmitting identical DS waveforms over all sub-bands. We then consider rate adaptation in the time domain, where the data rate is adapted such that a desired transmission quality is maintained. We analyze the BER performance of the proposed joint power and rate adaptation scheme with fixed average transmission power, and compare the performance to that of power and rate adapted single carrier DS/CDMA systems with RAKE receiver.

**Keywords:** multicarrier, code division multiaccess, adaptive systems, Rayleigh channels.

## 1 Introduction

Various multicarrier (MC) transmission schemes have been introduced into code-division multiple-access (CDMA) systems to obtain such advantages as higher rate data transmission, bandwidth efficiency, frequency diversity, lower speed parallel type of signal processing, and interference rejection capability [1]-[5]. These proposed techniques can be categorized into two types, the combination of frequency domain spreading and MC modulation [2,3] and the combination of time domain spreading and MC modulation [4,5]. A MC based direct-sequence (DS) CDMA scheme, belonging to the second type, was proposed in [4] as an alternative to the conventional single carrier (SC) system to yield a frequency diversity improvement instead of a path diversity gain. It was shown in [4] that,

⋆ Corresponding author.

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 209–218, 2011.

MC and SC DS/CDMA systems exhibit the same bit error rate (BER) performance in frequency-selective fading channels, but, in the presence of narrowband interference, the former provides a much better performance than the latter.

When the transmitter and the receiver are provided with the channel characteristics, the transmission schemes can be adapted to it, enabling more efficient use of the channel. In CDMA cellular systems, power adaptation is employed to maintain the received power of each mobile at a desired level [6]. The rate adaptation and the combined rate and power adaptation for DS/CDMA systems were considered in [7] and [8], respectively, and extended to a generalized joint adaptation scheme in [9]. In [10] an MC DS/CDMA system utilizing hopping over favorite sub-bands was considered and a frequency-hopping pattern generation method based on the water-filling algorithm was proposed. In [11] MC DS/CDMA with an adaptive subchannel allocation method was considered for forward links. However, dynamic power or rate allocation in the time domain was not addressed in [10] and [11].

In this paper, we investigate a joint power and rate adaptation scheme in the frequency-time domain for MC DS/CDMA communications. We assume that perfect channel state information (CSI) is provided at both the transmitter and the receiver. We consider, as a frequency domain power allocation, transmitting each user's DS waveforms over the user's sub-band with the largest channel gain, rather than transmitting identical DS waveforms in parallel over all sub-bands. We then consider rate adaptation in the time domain, where the data rate is adapted such that a desired transmission quality is maintained. We analyze the average BER performance of the proposed joint power and rate adaptation scheme in MC-DS/CDMA systems, and compare the performance to that of power and rate adapted SC-RAKE systems. Our results show that combining the frequency domain power adaptation and the time domain rate adaptation in MC DS/CDMA systems has significant performance gain when the total system bandwidth is fixed.

The remainder of this paper is organized as follows. In Section 2 we describe the system model. In Section 3 we analyze the average BER performance of the proposed adaptation scheme in MC DS/CDMA systems. Numerical results and discussions are presented in Section 4. In Section 5 conclusions are made.

## 2 System Model

In MC-DS/CDMA communications, the entire bandwidth is divided into $M$ equi-width disjoint frequency bands, where $M$ is the number of sub-bands. A data sequence multiplied by a spreading sequence modulates $M$ sub-carriers, and is sent over $M$ sub-bands. The transmitter for user $k$ is shown in Fig. 1(a), where $d_k(\cdot)$ and $p_k(\cdot)$ are the binary data sequence and the random binary spreading sequence for user $k$, respectively. The amount of power $\alpha_{k,m} S_T$ of the user $k$ is transmitted over the $m$th sub-band, where $S_T$ is the average transmission power

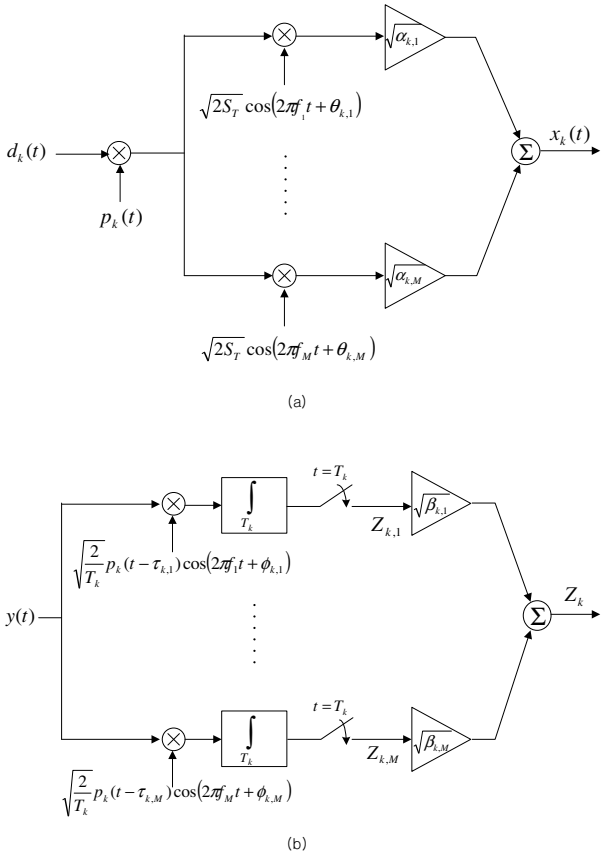



**Fig. 1.** Transmitter and receiver block diagrams for user $k$. (a) transmitter and (b) receiver.

and $\alpha_{k,m}$ is the transmitter power gain of the user $k$ on the $m$th sub-band, that is $E\left[\sum_{m=1}^{M}\alpha_{k,m}\right]=1$. The transmitted signal for user $k$ is, therefore, given by

$$x_k(t)=\sum_{m=1}^{M}\sqrt{2\alpha_{k,m}S_T}\,d_k(t)p_k(t)\cos(2\pi f_m t+\theta_{k,m}) \tag{1}$$

where $f_m$ is the $m$th carrier frequency.

We assume that the channel is frequency-selective Rayleigh fading, but the sub-bands are frequency-nonselective with respect to the spreading bandwidth and independent of each other. This can be achieved by selecting $M$ properly as in [4]. We also assume the channel variation due to the multipath fading is slow relative to the bit duration. We look only at a single cell system. The implications of a multiple cell system can be accounted for by the out-of-cell interference coefficient [6].

The receiver of user $k$ is shown in Fig. 1(b). The received signal $y(t)$ at the base station (BS) can be represented by

$$y(t) = \sum_{k=1}^{K} \sum_{m=1}^{M} \sqrt{2\alpha_{k,m} G_{k,m} S_T}\, d_k(t-\tau_{k,m}) p_k(t-\tau_{k,m}) \cos(2\pi f_m t + \phi_{k,m}) + n(t) \quad (2)$$

where $K$ and $\tau_{k,m}$ are the number of users and the delay on the $m$th sub-band for user $k$, respectively, and $\phi_{k,m} = \theta_{k,m} - 2\pi f_m \tau_{k,m}$. We assume that $\tau_{k,m}$'s and $\phi_{k,m}$'s are independent and uniformly distributed, the former over a bit interval and the latter over $[0, 2\pi]$. $G_{k,m}$ is the exponentially distributed random variable representing the channel power gain for user $k$ on the $m$th sub-band, and its probability density function (pdf) is given by

$$P_{G_{k,m}}(g) = \frac{1}{\Omega_0} e^{-g/\Omega_0} \quad (3)$$

where

$$\Omega_0 = E[G_{k,m}]. \quad (4)$$

In what follows, we will assume that $\Omega_0 = 1$. $n(t)$ represents the zero-mean white Gaussian noise with two-sided power spectral density $N_0/2$.

A coherent correlation receiver recovering the signal of user $i$ on the $m$th sub-band forms a decision statistic $Z_{i,m}$, given by

$$\begin{aligned} Z_{i,m} &= \sqrt{\frac{2}{T_i}} \int_{\tau_{i,m}}^{T_i+\tau_{i,m}} y(t) p_i(t-\tau_{i,m}) \cos(2\pi f_m t + \phi_{i,m})\, dt \\ &= \sqrt{\alpha_{i,m} G_{i,m} S_T T_i}\, d_i + I_{MAI} + \eta \end{aligned} \quad (5)$$

where $T_i$ is the bit duration, and $d_i$ is the data bit taking values $+1$ and $-1$ with equal probability, all for user $i$. The first term in (5) is the desired signal term. The second term $I_{MAI}$ is the multiple-access interference term induced by the other $K-1$ users, with zero mean and variance

$$E[{I_{MAI}}^2] = \sum_{\substack{k=1 \\ k \neq i}}^{K} \alpha_{k,m} G_{k,m} S_T T_c / 3 \quad (6)$$

where $T_c$ is the chip time. $\eta$ is the white Gaussian noise of mean zero and variance $N_0/2$.

At the combining stage, the correlator output of user $i$ on the $m$th sub-band is weighted with $\sqrt{\beta_{i,m}}$ where $\beta_{i,m}$ is the receiver power gain of the user $i$ on the $m$th sub-band, and then all the signals out of the correlators are combined coherently. In this work, we set $\beta_{i,m} = \alpha_{i,m} G_{i,m}$, since it is the optimal form of the diversity combining at the receiver [12]. Then, the combiner for user $i$ forms a decision statistic $Z_i = \sum_{m=1}^{M} \sqrt{\alpha_{i,m} G_{i,m}}\, Z_{i,m}$, and the bit energy-to-equivalent noise spectral density $E_b/N_e$ at the combiner output is given by

$$E_b/N_e = \frac{S_T T_i \left( \sum_{m=1}^{M} \alpha_{i,m} G_{i,m} \right)^2}{\sum_{m=1}^{M} \alpha_{i,m} G_{i,m} \left[ \sum_{\substack{k=1 \\ k \neq i}}^{K} 2\alpha_{k,m} G_{k,m} S_T T_c / 3 + N_0 \right]}. \quad (7)$$

## 3 Performance Analysis

First, we consider adaptively transmitting each user's DS waveforms over the user's sub-band with the largest channel gain, rather than transmitting identical DS waveforms in parallel over all sub-bands. The transmitter gain $\{\alpha_{i,m}\}$ in this case is given by

$$\alpha_{i,m} = \begin{cases} 1, \text{ if } \ G_{i,m} = G_i^{(1)} \\ 0, \text{ elsewhere} \end{cases} \tag{8}$$

where

$$G_i^{(1)} \triangleq \max(G_{i,1}, G_{i,2}, \cdots, G_{i,M}). \tag{9}$$

The pdf of $G_k^{(1)}$ is given by [13]

$$P_{G_k^{(1)}}(g) = Me^{-g}(1 - e^{-g})^{M-1}. \tag{10}$$

We now consider rate adaptation, where the data rate is varied so that the received $E_b/N_e$ is kept at a desired value while the transmission power is allocated as in (8). Thus it is a combined power adaptation in the frequency domain and rate adaptation in the time domain. It follows from (7) and (8) that, in order to maintain $E_b/N_e$ at a desired value $(E_b/N_e)_o$, the data rate is given by

$$\begin{aligned} R_i &\triangleq 1/T_i \\ &= \frac{1}{(E_b/N_e)_o} \cdot \frac{G_i^{(1)}}{\sum\limits_{k=1}^{\mathcal{K}_I} 2G_k^{(1)} T_c/3 + N_0/S_T} \end{aligned} \tag{11}$$

where $\mathcal{K}_I$ is the number of interfering users that transmit their DS waveforms over the same sub-band as user $i$. The probability of $\mathcal{K}_I$ being $n$ is given by

$$\Pr(\mathcal{K}_I = n) = \binom{K-1}{n} p^n (1-p)^{K-1-n}, \quad n = 0, 1, \cdots, K-1 \tag{12}$$

where $p$ is the probability that each user transmits one's DS waveform over the same sub-band as user $i$. In asynchronous systems, the probability $p$ is given by

$$\begin{aligned} p &= \Pr(\text{partially interfering}) + \Pr(\text{fully interfering}) \\ &= \frac{1}{M}(1 - \Lambda(M)) \cdot 2 + \frac{1}{M}\Lambda(M) \\ &= \frac{1}{M}(2 - \Lambda(M)) \end{aligned} \tag{13}$$

where $\Lambda(M)$ is the probability of two consecutive data bits being transmitted over the same sub-band, derived as $\Lambda(M) = 1/M$ for the random memoryless case.

The average data rate $\bar{R}_i$ with the rate adaptation is given by

$$\bar{R}_i = \frac{3\left(\sum_{m=1}^{M} 1/m\right)}{2T_c(E_b/N_e)_o} \cdot E\left[\frac{1}{\mathcal{I}}\right] \tag{14}$$

where

$$\mathcal{I} \triangleq \sum_{k=1}^{\mathcal{K}_I} G_k^{(1)} + \frac{3N_0}{2S_T T_c}. \tag{15}$$

Since all $G_k^{(1)}$'s are assumed to be independent, identically distributed (i.i.d.) random variables,

$$\begin{aligned} E\left[\left.\frac{1}{\mathcal{I}}\right|_{\mathcal{K}_I=n}\right] &= \int_a^\infty \frac{1}{x} P_{\mathcal{I}}(x) dx \\ &= \frac{1}{2\pi} \int_a^\infty \frac{1}{x} \int_{-\infty}^{\infty} \varphi^n(\omega)\, e^{-j\omega(x-a)}\, d\omega\, dx \end{aligned} \tag{16}$$

where

$$a \triangleq \frac{3N_0}{2S_T T_c} \tag{17}$$

and $P_{\mathcal{I}}(x)$ is the pdf of $\mathcal{I}$. $\varphi(\omega)$ is the characteristic function of $G_k^{(1)}$, given by

$$\begin{aligned} \varphi(\omega) &= \int_0^\infty P_{G_k^{(1)}}(x)\, e^{j\omega x}\, dx \\ &= M \sum_{k=0}^{M-1} (-1)^{k-M+1} \binom{M-1}{k} \frac{1}{(M-k-j\omega)}. \end{aligned} \tag{18}$$

It follows from (12), (14), and (16) that

$$\begin{aligned} \bar{R}_i = \frac{3\left(\sum_{m=1}^{M} 1/m\right)}{4\pi T_c (E_b/N_e)_o} \sum_{\mathcal{K}_I=0}^{K-1} \Bigg[ \binom{K-1}{\mathcal{K}_I} p^{\mathcal{K}_I} (1-p)^{K-1-\mathcal{K}_I} \\ \times \int_a^\infty \int_{-\infty}^{\infty} \frac{1}{x} \varphi^{\mathcal{K}_I}(\omega)\, e^{-j\omega(x-a)}\, d\omega\, dx \Bigg]. \end{aligned} \tag{19}$$

Since $1/\mathcal{I}$ in (14) is convex $\cup$, a lower bound on $\bar{R}_i$ can be obtained by using the *Jensen's inequality* [14]:

$$\begin{aligned} \bar{R}_i &\geq \frac{3\left(\sum_{m=1}^{M} 1/m\right)}{2T_c (E_b/N_e)_o} \cdot \frac{1}{E[\mathcal{I}]} \\ &= \frac{1}{(E_b/N_e)_o} \cdot \frac{1}{2p(K-1)T_c/3 + N_0/\left[S_T(\sum_{m=1}^{M} 1/m)\right]}. \end{aligned} \tag{20}$$

Setting the lower bound (20) equal to $1/T$, we get a lower bound on $(E_b/N_e)_o$ as

$$\left(\frac{E_b}{N_e}\right)_o \geq \left[\frac{2p(K-1)}{3N} + \frac{N_0}{S_T T(\sum_{m=1}^{M} 1/m)}\right]^{-1} \tag{21}$$

and an upper bound on the average BER with the proposed joint power and rate adaptation scheme as

$$\bar{P}_b \leq Q\left(\left[\frac{p(K-1)}{3N} + \frac{N_0}{2S_T T(\sum_{m=1}^{M} 1/m)}\right]^{-1/2}\right). \tag{22}$$

## 4 Numerical Results and Discussions

In order to compare the performance of the proposed joint power and rate adaptation scheme in MC DS/CDMA system to that of power and rate adapted SC DS/CDMA systems with RAKE reception, the followings are assumed:

- The total bandwidth of the SC system is the same as that of the MC system. That is, $T_c^{sc} = T_c/M$, where $T_c^{sc}$ is the chip time for SC-RAKE systems.
- In the SC system, the number of resolvable path $L$ is equal to the number of sub-bands $M$ in the MC system.
- In the SC system, the channel gain for user $k$ on the $l$th path, denoted by $G_{k,l}^{sc}$, are assumed to be i.i.d. exponential random variables. Namely, constant multipath intensity profile (MIP) is assumed in SC-RAKE systems.

In [7], the average data rate $\bar{R}_i$ that meets adequate BER requirements was analyzed for power and rate adaptation schemes in SC DS/CDMA systems. By setting $\bar{R}_i = 1/T$, the (average) BER performance of power and rate adaptation schemes in SC-RAKE systems can be obtained.

Fig. 2 depicts pdf of the channel gain at the combiner output in MC-ST scheme. For comparison purpose, the pdf of the combined channel gains in SC-RAKE scheme is also shown. In the SC system, MRC is employed in combining the output of the correlators in RAKE receiver. Denoting

$$G_i^{sc} \triangleq \sum_{l=1}^{L} G_{i,l}^{sc}\,, \tag{23}$$

where $G_i^{sc}$ is the combined channel gain for user $i$ after MRC in the SC-RAKE system. For fair comparisons, we normalize $E[G_i^{sc}] = 1$ in what follows. We note that MRC in MC-ST scheme corresponds to selection combining at the receiver since the DS waveform is transmitted over only the sub-band with the largest

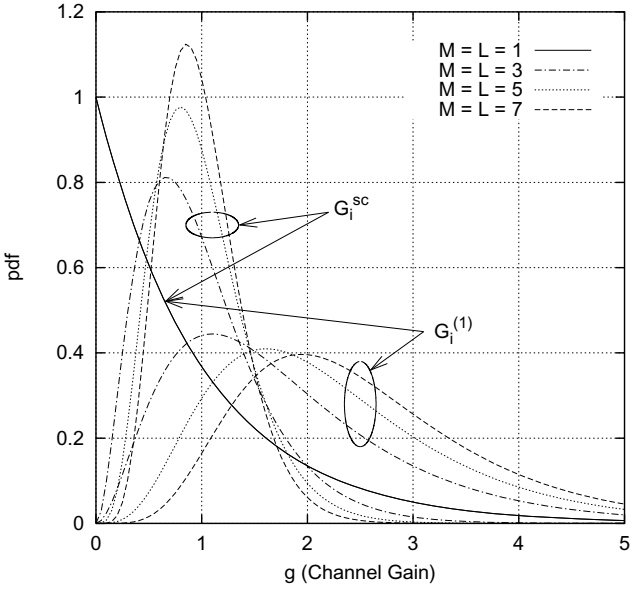



**Fig. 2.** Comparison of pdf for MC and SC-RAKE

channel gain. We can see that the probability of deep fading in SC-RAKE scheme is reduced as the number of resolvable path $L$ in RAKE receiver increases, and also the variance of the combined channel gain gets smaller as $L$ increases. When $L$ approaches to infinity, the pdf for SC-RAKE scheme becomes an impulse (no fading). For the case of MC-ST scheme, the probability of deep fading is also reduced as $M$ increases. The variance, however, is not reduced even for higher $M$. Notice that the mean value of the channel gain $G_i^{(1)}$ in MC-ST scheme becomes larger as $M$ increases, while the mean value of $G_i^{sc}$ in SC-RAKE scheme does not vary with $L$.

In Fig. 3, average BER versus $M(= L)$ is plotted for $S_T T/N_0 = 20$ [dB]. It is shown from Fig. 3 that the proposed joint adaptation scheme in MC DS/CDMA yields superior BER performance to power and rate adaptation schemes in SC-RAKE. This is because the joint power and rate adaptation in MC DS/CDMA adapts its data rate rather than the transmission power relative to the channel variations such as $G_i^{(1)}$ and $K_I$, in order to maintain a desired transmission quality. Besides, as we can see from (14) and [7, eq. (15)], the mean value of the combined channel gain has a significant influence on the performance of rate adaptation, since the achievable average data rate is immediately related with this mean value. Therefore the proposed adaptation scheme in MC DS/CDMA, in which the mean value of the combined channel gain is much larger than that in SC-RAKE as $M$ increases, is more effective transmission scheme than SC-RAKE scheme under the same frequency bandwidth, when power and/or rate adaptation is employed.

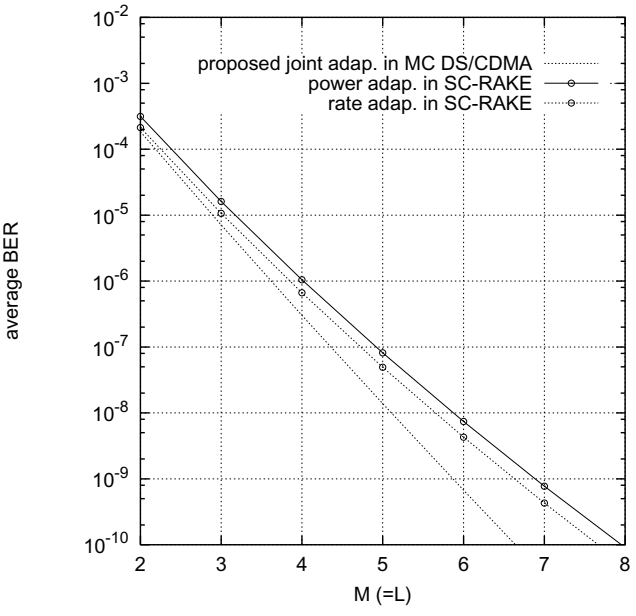



**Fig. 3.** Average BER versus $M(= L)$; $S_T T/N_0 = 20\,[\mathrm{dB}]$, $K = 30$, $N = 64$, $f_d T = 10^{-2}$

## 5 Conclusion

We considered jointly adapting transmission power and data rate in the frequency-time domain in MC-DS/CDMA communications. As a frequency domain power adaptation, we allocated the transmission power to only a sub-band with the largest channel gain. We then considered rate adaptation, where the data rate is adapted such that a desired transmission quality is maintained. We compared the performance of the proposed combined power and rate adaptation scheme in MC DS/CDMA systems to that of power and rate adaptations in SC-RAKE systems. The proposed joint rate adaptation employed in MC DS/CDMA systems was shown to have better performance than other adaptation schemes in SC-RAKE systems.

**Acknowledgments.** This research was supported in part by the MKE (The Ministry of Knowledge Economy), Korea, under the ITRC (Information Technology Research Center) support program supervised by the NIPA (National IT Industry Promotion Agency) (NIPA-2011-C1090-1121-0007) and in part by the project of development of local tactical technology which was conducted by the Ministry of Knowledge Economy of the Korean Government (project no. 70007243).

## References

1. Hara, S., Prasad, R.: Overview of multicarrier CDMA. IEEE Commun. Mag., 126–133 (December 1997)
2. Yee, N., Linnartz, J.P., Fettweis, G.: Multicarrier CDMA in indoor wireless radio networks. In: Proc. IEEE PIMRC, pp. D1.3.1–D1.3.5 (September 1993)
3. Fazel, K., Papke, L.: On the performance of convolutionally-coded CDMA/OFDM for mobile communication system. In: Proc. IEEE PIMRC, pp. D3.2.1–D3.2.5 (September 1993)
4. Kondo, S., Milstein, L.B.: Performance of multicarrier DS CDMA systems. IEEE Trans. Commun. 44, 238–246 (1996)
5. Sourour, E.A., Nakagawa, M.: Performance of orthogonal multicarrier CDMA in a multipath fading channel. IEEE Trans. Commun. 44, 356–367 (1996)
6. Gilhousen, K.S., et al.: On the capacity of a cellular CDMA system. IEEE Trans. Veh. Technol. 40, 303–312 (1991)
7. Kim, S.W.: Adaptive rate and power DS/CDMA communications in fading channels. IEEE Commun. Lett., 85–87 (April 1999)
8. Kim, S.W., Lee, Y.H.: Combined rate and power adaptation in DS/CDMA communications over Nakagami fading channels. IEEE Trans. Commun., 162–168 (January 2000)
9. Lee, Y.H., Kim, S.W.: Generalized joint power and rate adaptation in DS-CDMA communications over fading channels. IEEE Trans. Veh. Technol. 57, 603–607 (2008)
10. Chen, Q., Sousa, E.S., Pasupathy, S.: Multicarrier CDMA with adaptive frequency hopping for mobile radio systems. IEEE J. Select. Areas Commun. 14, 1852–1858 (1996)
11. Kim, Y.H., Song, I., Yoon, S., Park, S.R.: A multicarrier CDMA system with adaptive subchannel allocation for forward links. IEEE Trans. Veh. Technol. 48, 1428–1436 (1999)
12. Brennan, D.G.: Linear diversity combining technique. Proc. IRE 47, 1075–1102 (1959)
13. David, H.A.: Order Statistics, 2nd edn. John Wiley and Sons, Chichester (1981)
14. Feller, W.: An Introduction to Probability Theory and Its Applications, 2nd edn., vol. II. John Wiley and Sons, Chichester (1971)

# Worst-Case Traversal Time Modelling of Ethernet Based In-Car Networks Using Real Time Calculus

Kasper Revsbech, Henrik Schiøler, Tatiana K. Madsen, and Jimmy J. Nielsen

Dept. of Electronic Systems, Aalborg University, Denmark
{kar,henrik,tatiana,jjn}@es.aau.dk

**Abstract.** The paper addresses performance modelling of safety critical Ethernet networks with special attention to in-car networks. A specific Ethernet/IP based in-car network is considered as use-case. The modelling is based on the analytical method Real-Time Calculus (RTC), providing hard bounds on delay and buffer utilization. We show how RTC can be applied on the use-case network. Furthermore, we develop a network simulation, used to evaluate the overestimation. It is found that the delays from RTC is significantly overestimated. The bounds from RTC, however, are guaranteed bounds, which is not the case for the simulated.

**Keywords:** Real-Time Calculus, Vehicular Networks, Performance Modelling, Ethernet Modelling, Delay Bounds.

## 1 Introduction

Recently the car industry has started investigating the feasibility of merging all, or subsets of, in-car networks to Ethernet/IP. As an example [1] proposes Ethernet/IP as a bridging network, facilitating the variety of bus technologies and protocols currently used. Ethernet is currently widely used in a broad range of applications, however in safety-critical networks its applicability is still under question. As in-car networks are used for communication of critical date, e.g. wheel sensor information, there is a need of verifying its performance and reliability, as accidents and ultimately loss of human life can occur upon malfunction.

In this work we propose the analytical method of Real-Time Calculus (RTC) [2], to model Ethernet/IP based in-car networks. As RTC is an analytical method, it is appropriate, also in the early phases of the network design. Furthermore, RTC models will provide hard bounds on delay and buffer utilization, which are suitable dealing with safety critical systems, where deadline violations can have a crucial impact.

In this work we show how to model an Ethernet/IP based in-car network, to illustrate the applicability of RTC to model in-car networks. Furthermore, we contribute models that are generally applicable when modelling Ethernet networks, and in particular safety critical networks based on Ethernet.

The theory of RTC has been designed specifically for modelling/analyzing real-time systems. RTC is a derivative/flavor of Network Calculus (NC), which was first proposed by Cruz in [3] and [4], and later formalized by Boudec &

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 219–230, 2011.

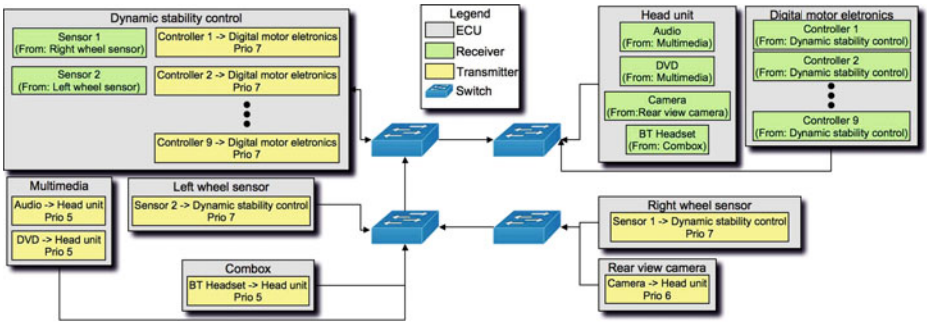



**Fig. 1.** The topology of the use-case network

Thiran in [5], and by Chang in [6]. Fidler recently provided an extensive survey [7] as well. In [8] Wandeler Et al. shows RTC applied in a case study, studying a distributed in-car radio navigation system.

The remainder of this paper is structured as follows: First we present an in-car network example, which serves as use-case network. In Sec 3 we present fundamentals of RTC, which is followed by a description of how RTC is applied on the use-case network. In Sec 5 we present a simulation model of the use-case network, in order to compare the RTC based model with a simulation model. The results of the RTC model, the simulation, and the comparison are given in Sec 6, followed by the conclusion and outlooks.

## 2 An Ethernet Based In-Car Network

To investigate the feasibility of applying the theory of RTC to Ethernet based in-car networks, an use-case network has been constructed. The constructed use-case network is a realistic sub-set of an in-car network, facilitating a sub-set of flows, present on a full scale in-car network. The link layer functionality is based on Ethernet (IEEE 802.3), where prioritization is achieved by use of VLAN headers (IEEE 802.1Q), and the prioritization specified in IEEE 802.1P. The Network layer is based on IPv4 and the Transport layer on UDP. The topology of the use-case network is depicted in Fig. 1. As seen from the figure, the topology is made such that two switches are placed in the back of the car, and two in the front. As seen, the communication lines are directed hence the model only consider one way traffic. The traffic flows in the use-case network are listed in Table 1. The two flows from the *Right wheel sensor*, and the *Left wheel sensor* to the *Dynamic stability control*, and the flows from the *Dynamic stability control* to the *Digital motor electronics*, are hard real-time sensor flows, and are assigned the highest priority (7) as they are safety-critical. The flow from the *Rear view camera* to the *Head unit* is a soft real-time video flow, and is assigned priority 6.

**Table 1.** Traffic flow specification, where **Min CT** is the minimum cycle time is in ms, **PS** the payload size in Bytes and **P** the priority

| **Flow** | **Min CT** | **PS** | **P** | **Flow** | **Min CT** | **PS** | **P** |
|---|---|---|---|---|---|---|---|
| *Right wheel sensor → Dynamic stability control* | | | | | | | |
| **Sensor 1** | 2.5 | 15 | 7 | | | | |
| *Left wheel sensor → Dynamic stability control* | | | | | | | |
| **Sensor 2** | 2.5 | 15 | 7 | | | | |
| *Dynamic stability control → Digital motor eletronics* | | | | | | | |
| **Controller 1** | 10 | 8 | 7 | **Controller 2** | 20 | 13 | 7 |
| **Controller 3** | 20 | 12 | 7 | **Controller 4** | 5 | 12 | 7 |
| **Controller 5** | 100 | 3 | 7 | **Controller 6** | 1000 | 3 | 7 |
| **Controller 7** | 100 | 7 | 7 | **Controller 8** | 200 | 7 | 7 |
| **Controller 9** | 10 | 8 | 7 | | | | |
| *Rear view camera → Head unit* | | | | | | | |
| **Camera** | 1.59 | 1472 | 6 | | | | |
| *Combox → Head unit* | | | | | | | |
| **BT Headset** | 1.25 | 10 | 5 | | | | |
| *Multimedia → Head unit* | | | | | | | |
| **Audio** | 8.4 | 1472 | 5 | **DVD** | 2.5 | 1472 | 5 |

The flow from the *Combox* to the *Head unit* is a Bluetooth audio flow. There are two flows from *Multimedia* to the *Head unit*, an audio and a DVD flow. These three flows are assigned the lowest priority (5), as they are not safety critical.

The definition of the use-case network also comprises a number of assumptions and delimitations, which are elaborated in the following:

- **No Address Resolution Protocol (ARP) requests.** ARP requests are not present in the network. It is fair to assume that the relation between the IP and the MAC addresses will be hardcoded, as the network does not have nodes entering and leaving.
- **Full duplex, switched network.** By use of a switched network cable collisions are avoided.
- **0.003 ms processing delay in the switch relay unit.** Device specific delay, in this case selected to 0.003 ms as we find this realistic.
- **Unlimited buffers.** It has been chosen not to model the effect of packet drops due to filled buffers, hence the buffers in the network are assumed to be unlimited [1].
- **No processing stack processing delays, except for a 0.01 ms delay in the IP stack.** Also a device specific delay, chosen to a realistic value.
- **No propagation delay.**
- **100 Mbit/sec Ethernet.**

[1] The theory of RTC provide methods to calculate bounds on needed buffer size.

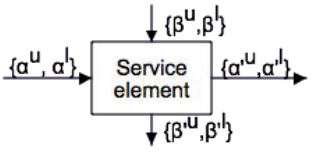



**Fig. 2.** The input and output bounds of a service element

## 3 RTC Theory

RTC is an analytical method allowing the network designer to calculate hard bounds on delay, and queue utilization of a system. In RTC every service element is modelled individually, where a service element can be in an abstract form, e.g. a whole switch, or more specific, e.g. a specific input port. The inputs and outputs of service elements, are then connected to model the whole network. Fig. 2 shows a service element with inputs and outputs. As shown in Fig. 2 the inputs of a service element is an upper and lower arrival bound $\{\alpha^u, \alpha^l\}$, and an upper and lower service bound $\{\beta^u, \beta^l\}$. The upper and lower arrival bounds, are use to model the input traffic to the service element. If we let $R(t)$ be a cumulative function, defined as the number of events from a particular traffic flow in the time interval $[0, t)$. Then we provide an upper and a lower arrival function $\alpha(\Delta) = [\alpha^u(\Delta), \alpha^l(\Delta)]$, for which the upper arrival curve $\alpha^u(\Delta)$ is an upper bound of the number of events that can occur in the traffic flow, in any time interval of length $\Delta$. Similarly the lower arrival function $\alpha^l(\Delta)$ is the lower bound of the events that will occur in a time interval $\Delta$. This leads to the following definition:

**Definition 1 (Arrival curves [8]).** *Let $R(t)$ denote the number of events that arrive on an event stream in the time interval $[0, t)$. Then, $R, \alpha^u$ and $\alpha^l$ are related to each other by the following inequality:*

$$\alpha^l(t-s) \leq R(t) - R(s) \leq \alpha^u(t-s), \qquad \forall s < t \tag{1}$$

*with $\alpha^u(0) = \alpha^l(0) = 0$.*

As described, a service element (Fig. 2) also has an upper and a lower service bound as inputs, bounding the service provided by the element. In analogue to the cumulative arrival function $R(t)$, we define a cumulative function $C(t)$ denoting the available processing resources of the element in the time interval $[0, t)$. Similarly to the arrival curves, we define an upper and lower service function $\beta(\Delta) = [\beta^u(\Delta), \beta^l(\Delta)]$. The upper service function $\beta^u(\Delta)$ denotes the maximum service available by the element in any time interval of length $\Delta$. The lower service function $\beta^l(\Delta)$ denotes the guaranteed minimum service, provided by the service element in any time interval of length $\Delta$. The formal definition of the upper and lower service curves is as follows:

**Definition 2 (Service curves [8]).** *Let $C(t)$ denote the number of processing or communication cycles available, from a resource over the time interval $[0, t)$, then $C$, $\beta^u$ and $\beta^l$ are related by the following inequality:*

$$\beta^l(t-s) \leq C(t) - C(s) \leq \beta^u(t-s), \qquad \forall s < t \tag{2}$$

*with $\beta^l(0) = \beta^u(0) = 0$.*

In this work we have chosen to confine ourselves to the following functions, to model arrival and service curves: The *affine* function, the *peak rate* function and the *rate latency* function. The affine function is defined in Eq. (3) [5], where $r$ defines the rate over time, and $b$ the burst that can arrive at any instance of time. Fig 3(a) shows an example of the affine function.

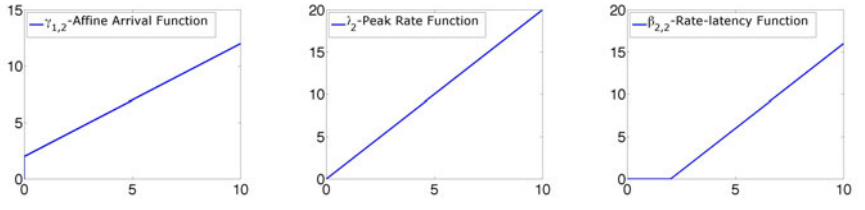
$$\gamma_{r,b}(t) = \begin{cases} rt + b & \text{if } t > 0 \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

The peak rate as defined in Eq. (4) [5] only takes on parameter: $R$, which is the rate over time. An example of the peak rate function is shown in Fig. 3(b).

$$\lambda_R(t) = \begin{cases} Rt & \text{if } t > T \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

The rate latency function is defined in Eq. (5) [5], where $R$ denotes the rate over time, and $T$ the latency. Fig. 3(c) depicts and example of the rate latency function.

$$\beta_{R,T}(t) = R[t-T]^+ = \begin{cases} R(t-T) & \text{if } t > T \\ 0 & \text{otherwise} \end{cases} \tag{5}$$




(a) Affine function with: $r = 1$ and $b = 2$ ($\gamma_{1,2}(t)$) (b) Peak rate function with: $R = 2$ ($\lambda_2(t)$) (c) Rate latency function with: $R = 2$ and $T = 2$ ($\beta_{2,2}$)

**Fig. 3.**

As seen from Fig 2, the outputs from a service element are: An upper and lower bound of the traffic flow, leaving the service element $\{\alpha'^u, \alpha'^l\}$ and the upper and lower service bounds $\{\beta'^u, \beta'^l\}$ of the service remaining, after serving the traffic flow traversing the element. For a scenario as depicted in Fig 2 where

only one flow traverses the service element, the output bounds: $\{\alpha'^u, \alpha'^l\}$ and $\{\beta'^u, \beta'^l\}$ are found according to Eq. (6)-(9) [8], where the definitions of $\otimes, \oslash, \overline{\otimes}$ and $\overline{\oslash}$ follows the same definitions in as regular NC and can be found in [5].

$$\alpha'^u = \min\left\{(\alpha^u \otimes \beta^u) \oslash \beta^l, \beta^u\right\} \tag{6}$$

$$\alpha'^l = \min\left\{\left(\alpha^l \oslash \beta^u\right) \otimes \beta^l, \beta^l\right\} \tag{7}$$

$$\beta'^u = \left(\beta^u - \alpha^l\right) \overline{\oslash} 0 \tag{8}$$

$$\beta'^l = \left(\beta^l - \alpha^u\right) \overline{\otimes} 0 \tag{9}$$

In Fig. 2 only one traffic flow traverses the scheduling element. However it is often the case that the service element serves a number of flows. This is done according to some predefined scheduling discipline. In this work we confine ourselves to Fixed Priority (FP) scheduling, and First In First Out (FIFO) scheduling. For FP scheduling in a preemptive scheduling setting the service curves $\left\{\beta^u_{FP,\alpha^x}, \beta^l_{FP,\alpha^x}\right\}$, available for a particular flow $\alpha^x$ can be found according to Eq. (10) and (11) [8].

$$\beta^u_{FP,\alpha^x} = \left(\beta^u - \alpha^l_{AGG}\right) \overline{\oslash} 0 \tag{10}$$

$$\beta^l_{FP,\alpha^x} = \left(\beta^l - \alpha^u_{AGG}\right) \overline{\otimes} 0 \tag{11}$$

$\alpha^u_{AGG}$ and $\alpha^l_{AGG}$ are the aggregate i.e. the sum of the upper and lower arrival bounds in set $\mathcal{P}$. In this case set $\mathcal{P}$ contains the flows which have higher priority than $\alpha^x$. $\alpha^u_{AGG}$ and $\alpha^l_{AGG}$ are found according to Eq. (12) and (13).

$$\alpha^u_{AGG} = \sum_{i \in \mathcal{P}} \alpha^u_i \tag{12}$$

$$\alpha^l_{AGG} = \sum_{i \in \mathcal{P}} \alpha^l_i \tag{13}$$

Similarly the service available for flow $\alpha^x$ in the case of FIFO scheduling can be found according to Eq. (10) and (11). The aggregates of the cross flow are found according to Eq. (12) and (13), however here the set $\mathcal{P}$ contains all cross flows. Note that this is using the principle of Blind Scheduling [5].

The delay bound for traversing a single service element can be found according to Eq. (14) [8]. Similarly the bound of needed buffer capacity for a single service element can be found according to Eq. (15) [8].

$$d_{max} \leq \sup_{\phi \geq 0}\left\{\inf\left\{\tau \geq 0 : \alpha^u(\phi) \leq \beta^l(\phi + \tau)\right\}\right\} \stackrel{\text{Def}}{=} \text{Del}(\alpha^u, \beta^l) \tag{14}$$

$$b_{max} \leq \sup_{\phi \geq 0}\left\{\alpha^u(\phi) - \beta^l(\phi)\right\} \stackrel{\text{Def}}{=} \text{Buf}(\alpha^u, \beta^l) \tag{15}$$

In the case where the delay and the utilized buffer bounds should be for the whole system or a specific part of the system, the bounds are found according to Eq. (16) and (17). Here the lower service bounds $(\beta^l_1, \beta^l_2, \ldots \beta^l_n)$ of the of the service provided to the flow in question are concatenated by use of the min-plus convolution ($\otimes$). As shown in [5] the concatenation gives more tight delay bounds that simply summing the delays experienced in every traversed service

element. It can be shown that the buffer bound is similar, whether or not the concatenation is applied.

$$d_{max} \leq Del\left(\alpha^u, \beta_1^l \otimes \beta_2^l \otimes \cdots \otimes \beta_n^l\right) \tag{16}$$

$$b_{max} \leq Buf\left(\alpha^u, \beta_1^l \otimes \beta_2^l \otimes \cdots \otimes \beta_n^l\right) \tag{17}$$

## 4 Ethernet/IP Based In-Car Network RTC Model

Having outlined the fundamentals of RTC, we now describe how RTC can be used to model the network described in Sec. 2. In the following we will describe how to model: A node, the traffic in the network, and a switch. Note that the time units of the model has been chosen as ms, and the data units as Bytes.

### 4.1 Node Modelling

The end nodes in the network are all 100 Mbit/s full-duplex Ethernet nodes. As described in Sec. 2, we have chosen to introduce a 0.01 ms processing delay at the IP layer. The link-speed constraint and the processing delay constraint, are used as input to a single service element, modelling a node. The upper service curve ($\beta^u$) is modelled as a peak-rate service curve ($\lambda_R(t)$), where the rate ($R$) is: 100 Mbit/s = 12500 Byte/ms. The lower service curve is modelled as a rate-latency service curve $\beta_{R,T}(t)$, with a rate ($R$) similar to the upper service bound, as the Ethernet link guarantees 100 Mbit/sec. The delay part ($T$) of the service curve is set to 0.01, as we have chosen a 0.01 $ms$ processing delay. Hence, the upper and lower service bounds of a node in the network become: $\left\{\beta^u = \lambda_{12500}, \beta^l = \beta_{12500,\ 0.01}\right\}$.

### 4.2 Traffic Modelling

In this work we have chosen to confine ourselves to affine arrival bounds, bounding the output of the physical layer from the nodes. As the bound is given for the physical layer, the flow data in Table 1 has been converted to affine arrival bounds, by adding the sum of the headers appended along with the Ethernet inter-frame gap, which is also considered as a part of a packet. The affine arrival bounds are presented in Table 2. Note that the minimum physical size of an Ethernet frame, including all headers and the inter-frame gab, is 88 Bytes. Packets are zero padded, if they are smaller. Also note that the "Controller aggregate" flow in Table 2 is an aggregate of the Controller flows, which has been created to simplify the model.

### 4.3 Switch Modelling

As described in Sec. 2, the network is prioritized by use of the methods defined in IEEE 801 P and Q. By use of the prioritization field (3 bit field), the switch can prioritize the Ethernet frames. As seen from Table 1, the modelled network

**Table 2.** Affine arrival bounds of the flows, where the rate is in Bytes/ms and the Burst in Bytes

| Flow | Rate | Burst | Flow Bound | Flow | Rate | Burst | Flow Bound |
|---|---|---|---|---|---|---|---|
| *Right wheel sensor → Dynamic stability control* | | | | | | | |
| **Sensor 1** | 35.2 | 88 | $\gamma_{35.2,88}$ | | | | |
| *Left wheel sensor → Dynamic stability control* | | | | | | | |
| **Sensor 2** | 35.2 | 88 | $\gamma_{35.2,88}$ | | | | |
| *Dynamic stability control → Digital motor eletronics* | | | | | | | |
| **Controller 1** | 8.8 | 88 | $\gamma_{8.8,88}$ | **Controller 2** | 4.4 | 88 | $\gamma_{4.4,88}$ |
| **Controller 3** | 4.4 | 88 | $\gamma_{4.4,88}$ | **Controller 4** | 17.6 | 88 | $\gamma_{17.6,88}$ |
| **Controller 5** | 0.88 | 88 | $\gamma_{0.88,88}$ | **Controller 6** | 0.088 | 88 | $\gamma_{0.088,88}$ |
| **Controller 7** | 0.88 | 88 | $\gamma_{0.88,88}$ | **Controller 8** | 0.44 | 88 | $\gamma_{0.44,88}$ |
| **Controller 9** | 8.8 | 88 | $\gamma_{8.8,88}$ | | | | |
| *Rear view camera → Head unit* | | | | | | | |
| **Camera** | 969.812 | 1542 | $\gamma_{969.812,1542}$ | | | | |
| *Combox → Head unit* | | | | | | | |
| **BT Headset** | 70.4 | 88 | $\gamma_{70.4,88}$ | | | | |
| *Multimedia → Head unit* | | | | | | | |
| **Audio** | 183.572 | 1542 | $\gamma_{183.572,1542}$ | **DVD** | 616.8 | 1542 | $\gamma_{616.8,1542}$ |

only utilize three of these values: 7,6, and 5. Note that 7 is the highest priority. As described in [9] a switch can be modelled in multiple ways. In this work we have chosen to abstract the switch model into a model of the output ports only. This means that we neglect the input buffers, and the backplane in the model. We assume that the processing delay in the input ports is insignificant, since we neglect them. Note that we assume that the switch does not apply cut-through, hence we have to account for the Store-and-Forward (STF) delay. Our modelling of the delay due to STF, is explained in the end of this section. The output ports of a switch are modelled as depicted in Fig. 4, where a chain of service elements, each representing a priority, is used. Each service element uses Blind scheduling (As described in Sec. 3) to model FIFO scheduling among the flows of same priority. As seen from Fig. 4 the remaining upper and lower service bounds $\{\beta'^u, \beta'^l\}$ of the priority 7 service element, are used as input service bounds for the priority 6 service element. Thus the flows of priority 6 get the "service available", after serving the flows of priority 7. Equally the input service bounds of the priority 5 service element, is the remaining service of the priority 6 element.

As the link-speed is 100 Mbit/sec, and we have chosen to introduce 0.003 ms delay in the switch (See Sec. 2), the upper and lower service curves provided to the highest priority service element (Priority 7) become: $\beta^u = \lambda_{12500}$ and $\beta^l = \beta_{12500,0.003}$.

As Ethernet switches are non-preemptive ([10]) all flows, except those served in the lowest priority service element, must expect Head-of-line Blocking (HOL) from a lower priority flow. This is modelled conservatively by adding a delay corresponding to the worst case waiting time of a lower priority flow $\frac{1542 \text{ Bytes}}{12500 \text{Bytes/ms}} = 0.1234$ ms, to flows exposed to HOL per scheduler.

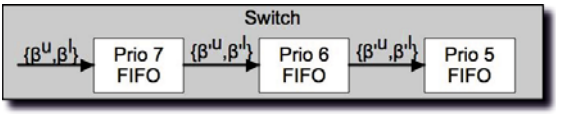



**Fig. 4.** Switch model

As mentioned above, the service curves in the model do not incorporate the delays due to STF. As these only depend on the frame length, and not on delays due to cross flows[2], the delay per receiver where STF occur can be calculated, and then added manually, according to Eq. (18), where $FS$ is the size of the transmitted frame (denoted as the burst in Table 2).

$$t_{stf} = \frac{FS \text{ Bytes}}{12500 \text{ Bytes/ms}} \tag{18}$$

## 5 Ethernet/IP Based In-Car Network Simulation Model

To compare the results of previously described RTC model with commonly used methods to evaluate network performance, the calculated bounds are compared with network simulation results of the use-case network. The simulation model is created in OMNET++ (V. 3), using the INET Framework.

The nodes in the use-case network are modelled in the following way: On the application layer, the *UDPApp* from the INET framework is used to represent a flow; meaning that if a node has multiple flows, it has multiple entities of the *UDPApp*. Each application is configured with a packet size according to Table 1. The cycle time is uniformly distributed between $t_{min}$ and $t_{max}$, where $t_{min}$ is the minimum cycle time from Table 1, and $t_{max} = 1.5 \cdot t_{min}$. This jitter is introduced to ensure simulation of non-synchronized flows. In reality the jitter might be smaller, however as we want to investigate the worst case, we are interested in this jitter to "seek" the worst/most congested state of the network.

The link layer is configured to be 100 Mbit/s Ethernet with VLAN based prioritization. The ability to append VLAN headers is not at part of the standard INET framework, but has been made possible due to the work in [11]. From this work it has also been made possible to simulate VLAN based switches. The switches are modelled as 100 Mbit/s switches, using the VLAN prioritization from [11], and the switch components from the INET framework. On top of Ethernet, the simulated network uses IPv4 and UDP according to the specifications in Sec. 2.

As we seek to investigate the maximum delay of every flow, we have to ensure it is observed in the simulation. In this work we assume that a simulation time of 5 hours is sufficient to observe the a realistic maximum delay. Hence we have configured the simulation with a simulation time of 5 hours. As we introduced the jitter of every flow, and each flow is appended to its own random generator, a long simulation time is as sufficient as many repetitions with different seeds.

[2] As the flows already are interleaved when they are transmitted.

## 6 Results

In Table 3 we present the delays obtained from the RTC model, along with the maximum observed delays obtained by network simulation. As seen from the table, the flows of high priority (*Sensor 1, Sensor 2* and *Controller aggregate*) have the smallest delays. This is as expected since they are not interfered by any cross flows. Furthermore as expected, the *Audio* and the *DVD* flows have the highest delays as they are of the lowest priority.

To compare the results from the RTC model and the simulation, the overestimation from the RTC model against the simulation is shown in Fig. 5. As seen from the figure, the overestimation by the RTC model seems significant, the mean overestimation is 179%. However the results from the RTC model are hard-bounds derived from an analytical model, thus one can be confident that no delays will arise greater than the calculated bound. This is a fundamental property of the RTC method. Regarding simulation one has to realize that it is impossible to know when the absolute worst case delay has been seen. As explained in Sec. 5, we assume that a simulation time of 5 hours is enough to reveal the true worst case of the system, however we cannot guarantee this.

The significant difference in the overestimation between the last three flows and the remaining four, is due to the fact that the three flows (*BT Headset, Audio* and *DVD*) have high bursts, and traverses the same service elements multiple times, thus they are multiplexed several times. In reality when flows are multiplexed once and traverses the same path, they do not interfere each other any more. In the domain of NC this problem is referred to as Pay Multiplexing Only Once (PMOO), and is addressed in e.g. [12,13,14,7]. However to the knowledge of the authors, the methods derived have not (yet) been proved within the domain of RTC. Hence, methods can be applied to reduce the overestimation.

**Table 3.** End to end delay bounds of the flows in ms. The delays presented are those obtained from the RTC model, and the maximum observed in the simulation.

| **Flow** | **RTC** | **Maximum observation from simulation** |
|---|---|---|
| *Right wheel sensor → Dynamic stability control* | | |
| **Sensor 1** | 0.512 | 0.303 |
| *Left wheel sensor → Dynamic stability control* | | |
| **Sensor 2** | 0.38 | 0.181 |
| *Dynamic stability control → Digital motor electronics* | | |
| **Controller aggregate** | 0.418 | 0.177 |
| *Rear view camera → Head unit* | | |
| **Camera** | 1.601 | 0.78 |
| *Combox → Head unit* | | |
| **BT Headset** | 2.575 | 0.547 |
| *Multimedia → Head unit* | | |
| **Audio** | 2.689 | 0.776 |
| **DVD** | 2.448 | 0.778 |

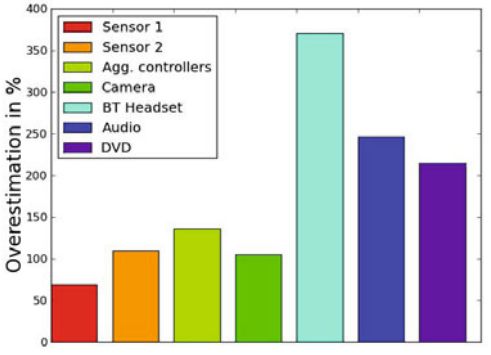



**Fig. 5.** The overestimation, per flow, for the RTC model compared to the simulation

## 7 Conclusion and Outlook

An RTC model of an Ethernet based use-case in-car network has been proposed. The delays obtained by the calculations in RTC, have been compared with delays obtained by performing simulation of the same use-case network. It can be concluded from the comparison that the overestimation in RTC for this scenario is significant, the level depends on the nature of the flow, and the path it traverses. As explained in Sec. 6, there are known optimizations that, (when proven valid in the domain of RTC), can be applied. Even though the overestimation is significant it is important to emphasize that, the maximum observed delay from simulation, is not a bound and delays greater than the observed can occur. This means that as we know the delays obtained from RTC are overestimated, and those from simulation might be "underestimated", the "true delays" are found in between.

Due to the overestimation in RTC the price of over-dimensioning a network can be high. However as in-car networks are safety critical, hard bounds are needed, and bounds need to be found by methods like RTC. Therefore, it is of a high importance to find methods to reduce the overestimation in RTC, by improving the bounding methods. As mentioned in Sec. 6, one way would be to investigate how the principles of PMOO can be applied within the domain of RTC.

**Acknowledgment.** The authors would like to thank the Research and Technology department of the BMW, for participating in the development and publication of this paper.

## References

1. Steffen, R., Bogenberger, R., Hillebrand, J., Hintermaier, W., Winckler, A., Rahmani, M.: Design and realization of an ip-based in-car network architecture. In: The First Annual International Symposium on Vehicular Computing System, Dublin, Ireland, July 22-24 (2008)

2. Thiele, L., Chakraborty, S., Naedele, M.: Real-time calculus for scheduling hard real-time systems. In: Proceedings of the 2000 IEEE International Symposium on Circuits and Systems, ISCAS 2000, Geneva, vol. 4, pp. 101–104 (2000)
3. Cruz, R.: A calculus for network delay. i. network elements in isolation. IEEE Transactions on Information Theory 37, 114–131 (1991)
4. Cruz, R.: A calculus for network delay. ii. network analysis. IEEE Transactions on Information Theory 37, 132–141 (1991)
5. Le Boudec, J.-Y., Thiran, P.: Network calculus: a theory of deterministic queuing systems for the internet, May 10. Springer-Verlag New York, Inc., Heidelberg (2004)
6. Chang, C.-S.: Performance Guarantees in Communication Networks. Springer, London (2000)
7. Fidler, M.: A survey of deterministic and stochastic service curve models in the network calculus. IEEE Communications Surveys and Tutorials 12, 59–86 (2010)
8. Wandeler, E., Thiele, L., Verhoef, M., Lieverse, P.: System architecture evaluation using modular performance analysis: a case study. International Journal on Software Tools for Technology Transfer (STTT) 8 (October 2006)
9. Georges, J., Divoux, T., Rondeau, E.: Comparison of switched ethernet architectures models. In: Proceedings of the IEEE Conference Emerging Technologies and Factory Automation, ETFA 2003, vol. 1, pp. 375–382 (2003)
10. Seifert, R.: The Switch Book: The Complete Guide to LAN Switching Technology. John Wiley & Sons, Inc., New York (2000)
11. Rahmani, M., Tappayuthpijarn, K., Krebs, B., Steinbach, E., Bogenberger, R.: Traffic shaping for resource-efficient in-vehicle communication. IEEE Transactions on Industrial Informatics 5(4), 414 (2009)
12. Schmitt, J., Zdarsky, F.: The disco network calculator: a toolbox for worst case analysis. In: Proceedings of the 1st International Conference on Performance Evaluation Methodolgies and Tools (January 2006)
13. Schmitt, J., Zdarsky, F., Martinovic, I.: (technical report) performance bounds in feed-forward networks under blind multiplexing (January 2006), http://disco.informatik.uni-kl.de/publications/SZM06-1.pdf
14. Bouillard, A., Gaujal, B., Lagrange, S., Thierry, E.: Optimal routing for end-to-end guarantees: the price of multiplexing. In: Proceedings of the 2nd International Conference on ... (January 2007)

# Infrastructure-Assisted Probabilistic Power Control for VANETs

Dmitri Moltchanov, Jakub Jakubiak, and Yevgeni Koucheryavy

Department of Communication Engineering,
Tampere University of Technology,
P.O.Box 553, Tampere, Finland
moltchan@cs.tut.fi

**Abstract.** In this paper we propose a new way to control power allocations for vehicle transmissions in vehicular networks (VANET). Our proposal is based on usage of infrastructure nodes along the roads that can be equipped with special hardware advertising information about distance between successive vehicles on the road. We analyze our system exploring the trade-off between density of infrastructure nodes and accuracy of power allocation. Our numerical results demonstrate that the proposed approach may improve accuracy of power allocation. Finally, the proposed approach does not necessarily replaces those proposed to date but may coexist with them and improve their performance.

## 1 Introduction

A variant of CSMA/CA mechanism is accepted as a MAC protocol for communication in vehicular ad hoc networks [1]. In CSMA/CA-based systems the probability of correct delivery of a packet strongly depends on the number of stations competing for transmission resources (see e.g. [2]). Practically, the bigger the amount of stations in the local contention environment the higher the probability of collision. To decrease the amount of stations simultaneously contending for resources stations need to transmit using as less power as possible. From the other hand, when density of cars is rather small to increase the connectivity between stations along the road stations are often required to transmit using their maximum possible power. As a result, optimal power allocation is one of the most important features of forthcoming VANET networks.

There are a number of ways to get information about distances between cars on the road. First of all, vehicles are expected to use navigation systems and include their current position in their outgoing messages. Thus, positioning of neighboring cars can be obtained as a result of decoding of received messages. The biggest advantage of this approach is that it is accurate. However, vehicles may not communicate frequently enough such that there is always up-to-date information about location of neighboring nodes. Furthermore, those vehicles serving as relay nodes may not be allowed to supplement their location information in relayed messages.

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 231–237, 2011.

According to IEEE 802.11p protocol neighboring stations are expected to exchanging presence information using beaconing [1]. As a result, those stations that are in coverage area of the transmitting one hear beacons and may use the received signal strength to obtain the information about the distance till the transmitting node. There are a number of disadvantages associated with this approach. First of all, this method is not accurate, especially, in crowded environment, where the received signal strength is affected by many obstacles moving in the radio channel. Secondly, the forward channel could be significantly different compared to the backward one due to differences in the speed of mobiles. Finally, beacons transmitted by stations may also collide causing long pauses without updates.

Another approach to get positioning information for power allocation is to use infrastructure nodes that could be located along the roads. Currently, these nodes are already installed along many roads in Europe and North America provides drivers with information about the current weather situation, notifying about the current speed, monitoring speed regime of vehicles, etc. Furthermore, infrastructure nodes installed along the roads are anyway expected to participate in communication supplementing drivers with additional information using communication channels [link]. As a result, it is straightforward to assume that these stationary nodes may also instructed to provided information about distances between cars on the road. Taking into account that the distance is measured locally this approach could be exceptionally accurate. In this paper we explore potential gains obtained this information. We note that the approach proposed in this paper is not expected to fully replace those based on beaconing and GPS-assisted.

The rest of the paper is organized as follows. We discuss the concept of infrastructure assisted power control in Section 2. Next, we analyze accuracy of this approach in Section 3. Numerical result are presented in Section 4. Finally, conclusions are drawn.

## 2 The Concept

Let us fix a certain time instant $t$ and take a snapshot of cars moving along the road as shown in Fig. 1, where circle dots denote moving objects (e.g. cars), crosses marks stationary infrastructure nodes. We assume that infrastructure nodes are separated by well-known constant distance $m$. At a certain time instant $t$ cars are separated by unknown distances $d_{ij}$, $i = 0, 1, \ldots$, $j = i + 1$.

Consider a car $i$ and concentrate on one-hop forward connectivity with next next car $i + 1$. In order for transmission of car $i$ to reach car $i + 1$ with non-negligible probability we need to know the distance till the next car, $d_{ii+1}$. Assuming that cars are moving st constant speeds and infrastructure measures and provides distance till the next car on the road $d_{ij}$ are immediately known and can be used to estimate the required power level to reach the next car or possibly decide not to transmit due to the next care is out of coverage.

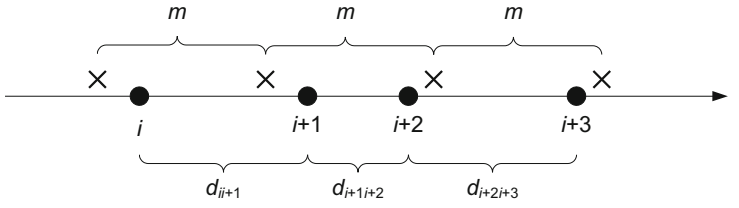



**Fig. 1.** A snapshot of cars moving along the road

In reality, given relatively big $m$ speed or cars is variable. Thus, there is trade-off between $m$ describing the density of infrastructure nodes and accuracy of estimation. In what follows, we explore it using simple probabilistic framework.

## 3 The Decision Statistics

Let $d_{ii+1}(t)$ be the distance between cars $i$ and $i+1$ at the time instant $t$ when car $i$ obtains information from a certain infrastructure node. Observe that at this time instant this information is already not absolutely accurate. This uncertainty is further accumulated till the next infrastructure node separated by $m$. Depending on the type of the road segment this uncertainty may contain both deterministic and stochastic component. Deterministic components arises when there is a certain traffic sign further along the road (e.g. speed limit). Another reason for deterministic component is current speed or mobile $i$ and $i+1$, $v_i(t)$ and $v_{i+1}(t)$. This information can be communicated to the car by the infrastructure node. Since these reasons affecting the distance between cars are easy to take into account we assume that the road is homogenous and $v_i(t) = v_{i+1}(t)$.

Another type of uncertainty is purely stochastic. It is primarily caused by drivers' habits and is mostly unpredictable. In this paper we propose to model it using time-dependent gaussian process with mean 0 and variance $s^2(l)$, $l \in (0, m)$. Note that $s^2(0)$ is a certain measure of "starting" uncertainty at the point when car $i$ receives information about the distance to car $i+1$. Given the constant long-time averaged speed of mobiles $v$, $s^2(im) = s^2(0)$, $i = 0, 1, \ldots$. The process of uncertainty accumulation between infrastructure nodes on the road is illustrated in Fig. 2.

As one may observe, in addition to the current distance from the previous infrastructure point $l$, $l \in (0, m)$ $\sigma^2(l)$ should also depend on the speed of mobiles on the road. Indeed, the higher the average speed of car flow on the road the higher maximum value of uncertainty just accumulated just before the infrastructure node. Although this dependency may have quite complicated behavior, it is fair to assume that uncertainty accumulated linearly faster when speed increases.

Once the current distance from the previous infrastructure point, $l$, is known we propose to use percentiles to determine the current distance with a certain

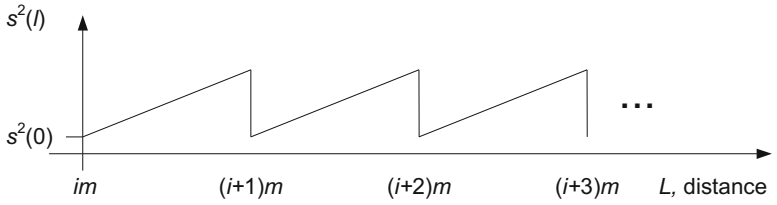



**Fig. 2.** Increase of uncertainty between infrastructure nodes

level of assurance. Recall that $x$-percentile is the value of a variable below which $x$ percents of observations fall. Thus, choosing $x$ to be close enough to 100% (e.g. 95%) we almost surely determine the maximum distance till the next node. Formally, since we model the uncertainty at the $l$ separation distance from the previous infrastructure point using Gaussian random variable with mean 0 and variance $s^2(l)$ we have

$$Q_{x*0.01}(l) = N^{-1}(0, s^2(l)), \qquad l \in (0, m), \tag{1}$$

where $N^{-1}$ denotes inverse Normal distribution, $Q_{x*0.01}(l)$ is the corresponding quantile function, and $x$ is the value we are looking for. The typical increase of uncertainty as cars moves between infrastructure nodes is shown in Fig. 3.

There are a number of reasons for choosing percentiles as the decision statistics. Firstly, percentiles are less susceptible to long-tailed distributions and outliers. Even if actual data deviate from assumed distribution percentiles provide fairly reliable description of statistical data. Secondly, percentiles are widely-used in networking and engineers this statistics. Percentiles are also very flexible

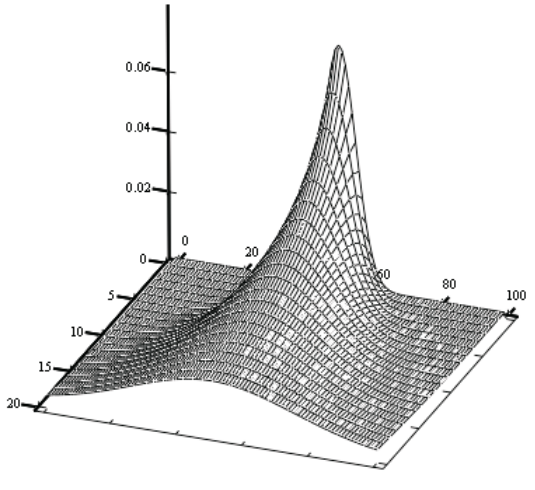



**Fig. 3.** Increase of uncertainty in terms of normal distribution

in terms of making a system more or less conservative. Finally, estimation of percentiles can be performed even when the distribution of interest does not pose analytical expression for its quantile function (inverse of its cumulative distribution function (cdf) see e.g. [3]).

Note that for Normal distribution there are no analytical expression for quantile function. However, one may get numerical algorithm for arbitrarily high accuracy constructing and solving non-linear ordinary differential equation for the normal quantile [4].

## 4 Results and Discussion

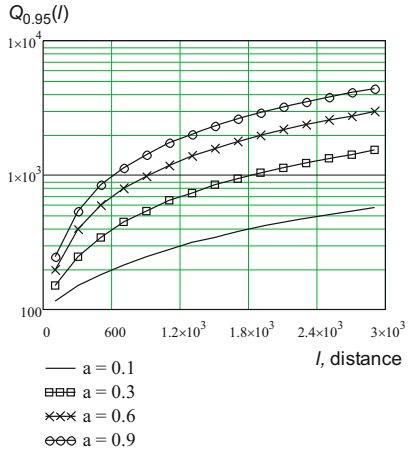
Observing (1) one may notice that there is dependence between the amount of uncertainty accumulated at the distance $l$ and the expected distance till the next car. It could be the case that the estimate distance for a certain relatively high assurance is too big to provide constructive information. To understand under which values of $s^2(0)$ and for which uncertainty increase rate the proposal is sensible we need to visualize $s^2(l)$ as a function of $l$. To do so we represent $s^2(l)$ as a function of $s^2(0)$ and certain $a$ in terms of linear function $s^2(a, l) = s^2(0) + al$. Further using these dependencies, we obtain the maximum possible distance between infrastructure points that provide improvement for a certain maximum communication range $r$.

Fig. 4 demonstrates values of $Q_{0.95}(l)$ and $Q_{0.90}(l)$ as a function of the amount of uncertainty represented by $a$, and the distance between infrastructure nodes. The initial distance for these illustrations, $d_{i,i+1}$, was set to 100 units. In addition, we assumed that $s^2(0) = 0.0$. First of all, as we expected quantiles preserve linear property of the uncertainty increase model. Basically, the higher the stochastic uncertainty of the car position the higher the distance that need to be covered by the transmission to achieve a certain reception probability [1]. Next, it is obvious that the increase in $a$ degrades the performance of the algorithm drastically. This is especially true taking into account that the maximum coverage range of transmitters in VANETs is rather limited. However, when $a$ is rather small good performance can be achieved even for fairly large distances between infrastructure nodes. As a result, given a certain value of $a$ and $r$, illustrations presented in Fig. 4 allows to estimate the required transmission power to reach the next car on the road. They could also be used to estimate the optimal distance between infrastructure nodes along the road.

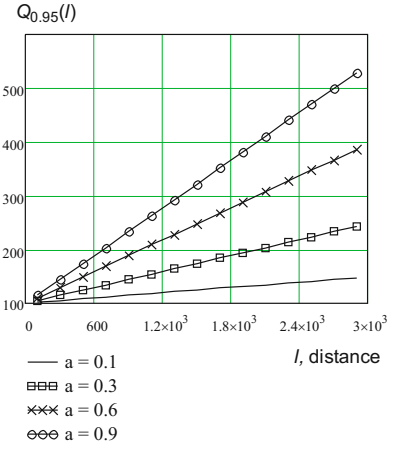
Recall that we do not visualize the effect of $s^2(0)$ and $d_{ii+1}$. The reason is that the effect of these two parameters is quite straightforward, e.g. $s^2(0)$ adds a constant component, while $d_{ii+1}$ determines the mean of the normal distribution around which the position of a car spreads out. Adding these factors to the resulting expression results in simple scaling of graphs presented in Fig. 4.

We would like to note that the proposed approach can also be used in conjunction with other information delivery mechanisms. For example, when information

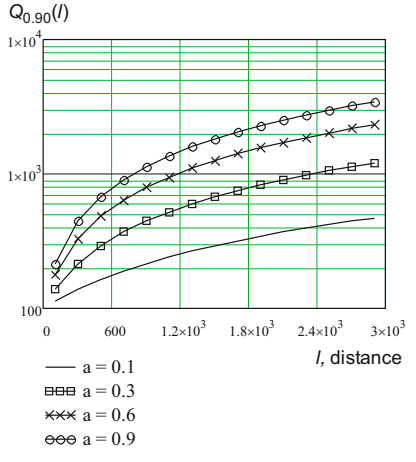
[1] The correct packet reception probability is a function of the distance between cars.



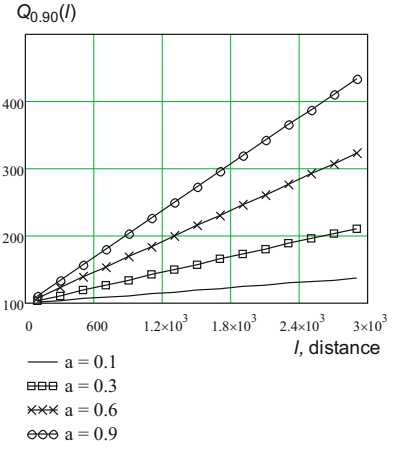

(a) 0.95-quantile




(b) 0.95-quantile




(c) 0.90-quantile




(d) 0.90-quantile

**Fig. 4.** $Q_{0.95}(l)$ and $Q_{0.90}$ as a function of $a$, and $l$

about the next car on the road is obtained using beaconing or location information in the body of the received message, the proposed approach can be used to estimate the required power level to reach this station. Once this information is update using any available mechanism, the algorithm is re-initialized. In this case time instants when information is updated become random in Fig. 4. As a result, the simple concept proposed in this work can co-exist with all three possible mechanisms.

## 5 Conclusions

In this paper we proposed probabilistic algorithm estimating the required power level to reach the next car on the road using information provided by infrastructure nodes. In absence of other information refining our approximation (e.g. those obtained using beaconing or reception of messages from neighbors with GPS information included) our approach would provide approximate results to ensure connectivity between neighboring nodes on the road. However, its accuracy strongly depends on the value of position uncertainty of cars on the road. It is expected that these values can be quite different for different types of the roads and possibly, long-term averaged velocities of a car flow. One of the topic of further studies is to obtain real statistics of cars flows and test the proposed algorithm in realistic environment.

We would like to specifically stress that the proposed concept of using stochastic uncertainty statistics and quantiles to estimate distance till the next car with a certain probability can also be used in conjunction with other neighbor location discovery algorithms. This could be especially helpful when location information is not delivered timely on regular intervals due to possible collisions, random channel fades, etc.

## References

1. IEEE standard for information technology part 11: Wireless medium access control (MAC) and physical layer (PHY) specifications. Technical report, IEEE
2. Bianchi, G.: Perfromance analysis of the IEEE 802.11 distributed coordination function. IEEE JSAC 18(3), 535–547 (2000)
3. Gilchrist, W.: Statistical Modelling with Quantile Functions. CRC Press, Boca Raton (2000)
4. Steinbrecher, G., Show, W.: Quantile mechanics. IEEE Trans. Veh. Tech. 19(2), 87–112 (2008)

# Node Mobility Modeling in Ad Hoc Networks through a Birth and Death Process

Carlos A.V. Campos[1], Luis F.M. de Moraes[2],
Eduardo M. Hargreaves[2], and Bruno A.A. Nunes[2]

[1] Department of Applied Informatics,
Federal University of State of Rio de Janeiro – UNIRIO, Rio de Janeiro, Brazil
beto@uniriotec.br
[2] Laboratory of High Speed Networks – RAVEL/COPPE,
Federal University of Rio de Janeiro – UFRJ, Rio de Janeiro, Brazil
{moraes,hargreaves,bruno}@ravel.ufrj.br

**Abstract.** Mobility models are used to represent the movement behavior of mobile devices in ad hoc networks simulations. As a consequence, the results obtained via simulations for specific characteristics related to mobile ad hoc networks are expected to be significantly dependent upon the choice of a particular mobility model under consideration. In this context, we present in this work[1] a new mobility model, based on *Birth-Death* stochastic processes, which allows us to adjust mobility parameters according to the movement profile intended to be represented. The impact of mobility models in the simulation of ad hoc networks is observed through the performance evaluation of metrics related to the AODV routing protocol, by comparing the results obtained under the Birth-Death framework used in this paper with those calculated under the Generic Individual Mobility Markovian and the Random Waypoint models.

**Keywords:** MANETs, mobility model, birth-death process.

## 1 Introduction

Mobile Ad Hoc Networks (MANETs) consist of a set communication devices (nodes) which are able to exchange data without the need of a preexisting infrastructure. In general, these networks are self-configuring, self-organizing and present a dynamic topology – characteristics that make them an important subject of analysis, as they can provide great flexibility and are adding new areas of applications in the field of wireless communication networks.

In order to support mobility of ad hoc networks, one must face a series of new challenges, such as the design of specific routing protocols, quality of service provision, improving and measuring maximum channel capacity and the reducing the energy consumption by client nodes, to mention a few. In addition, each developed mechanism or algorithm must be very well evaluated before being

[1] This work was sponsored partially by CAPES, CNPq and FINEP.

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 238–250, 2011.

implemented. This paper is mainly concerned about the problem of choosing a modeling capable of representing specific mobility patterns for the mobile users.

When representing the motion of nodes in a mobile network one can choose between two alternatives [4]: (1) the use of real mobility traces, and (2) the use of mobility models. Despite being the best method to represent reality, the use of real traces is not common, because keeping track of nodes motion is not an easy task in ad hoc networks. Therefore synthetic mobility models have been extensively used as the main alternative to represent the mobility in MANETs.

This paper addresses mobility models for ad hoc networks and proposes a model based on Birth-Death processes, which is a special class of Markov chains. This model gives the network mobile nodes a sense of direction based on the knowledge of previous speed and angle of direction in the next transition. Furthermore, a comparison among some mobility models is performed and the results show that the proposed model presented fewer number of sharp turn and sudden accelerations, which are interesting characteristics when building more realistic mobility models [2,8].

The remainder of this paper is organized as follows. In Section 2 some mobility models used in ad hoc networks are briefly described. In Section 3 our model is presented and described analytically. In Section 4 we present a performance evaluation of this model. Section 5 concludes this paper.

## 2 Related Work

Mobility models are largely used in mobile networks simulation and performance evaluation studies. A mobility model can be defined as a set of rules applied in order to determine the location of mobile entities in time, also defining how and when its location is going to change. These models are used in network simulations to generate changes in network topology due to node movement.

In this work, we give focus to mobility models applied to ad hoc networks. These models generally represent the network node micromobility and describe the movement of each single entity independently. Moreover, in performance evaluation of such networks, sometimes it is interesting to include in the simulation, some kind of restrictions to movement such as streets, in case of outdoor application and walls and corridors for indoor simulation. It is important to notice that obstacles, walls and corridors have great influence over wireless signal radio propagation characteristics. For this reason, for indoor simulation and simulations considering obstacles, the appliance of propagation models and good physical layer simulators are necessary. However, there is no simulator yet, with a good physical layer and propagation model implementation that reflects real radio propagation conditions of the wireless network environment. We know that the physical layer have a significant impact in the final evaluation results, however, we did not treat this problem. In this work, we focus our research only on simulations of wireless network scenarios with no restrictions and obstacles. We also use the native implementation of the physical layer available on NS-2 [6].

In the following, we describe some individual mobility models for scenarios with no restrictions. We talk about several related work and finally present a discussion about the movement representation through synthetic models.

In the random waypoint (RWP) model, described in [4], the initial position of the nodes can be set randomly. Usually researches distribute wireless nodes initial positions, uniformly over the simulation area. The nodes remain in the same position during a random time interval called pause time. After the pause time, the node uniformly chooses a new position inside the simulation area, and moves towards it with constant speed $v$, also chosen uniformly between an interval $V_{min}$ and $V_{max}$. This characteristic eliminates any sense of direction in the movement, which can be considered an unwanted characteristic.

The generic individual Markovian mobility (GIMM) model, proposed in [5], uses discrete time Markov chains [10] in order to model nodes movement. Its main characteristic is to use a Markov chain to give the movement behavior for the $x$ direction and an other chain for the $y$ direction. This model allows movements in the same and adjacent directions, in speed changes and intervals of pause between consecutive movements. Besides, it avoid sharp turns and sudden stops, representing a more adequate movement than random models, specially the RWP model that is the most used.

Recently, several mobility models were proposed. For example, in [1], obstacle-based model is presented and, in [9,13], models based on human mobility are proposed. Another form of mobility representation is through social behavior of users, which is presented in [12]. However, these models do not evaluate the number of the sharp turns and sudden changes in speed.

A good mobility model should try to reproduce the movement pattern of a real world mobile node, in a way that variations in speed and direction of the movement occur in a similar manner as they occur in a real world scenario. In other words, movement modeling of mobile nodes should consist in the creation of models capable of reflecting real characteristics of the mobile user movement. According to [2,4,5], a mobility model that does not consider correlation between time $t$ and time $t + \Delta t$ speed and direction is said to be no-realistic. This means that real movement happens when the trajectories follow a certain sense of direction, where speed and direction are not chosen randomly. Moreover, in [8], authors assume that a realistic movement is a smooth one, where mobility models should avoid sharp turns and sudden changes in speed. Thus, in the next section a new model is proposed with this objective.

## 3 Proposed Mobility Model

Assuming that nodes in a mobile network should move according to some sense of direction and not randomly choosing destinations [4], we propose here a new continuous time Markovian mobility model. In the proposed model, what gives nodes sense of direction is the knowledge of previous speed and angle of direction in the next transition. This is similar to what happens in GIMM. The difficulties

in obtaining stationary distributions for GIMM is due to the fact that in this model there are many transition probabilities between states. Usually, if the Markov chain does not present any special structure that allow simplification of complex linear equations, the solution of this chain are only possible through numerical methods like, Gauss elimination or Gauss-Seidel iterations [7].

In the proposed model, we allowed transitions only between adjacent states. This way, our new model is similar to a birth-death stochastic processes, which are a very special class of Markov chains, normally used for modeling changes in population size. In our model, the speed behavior is ruled by a Markov chain. Once speed is a vector quantity, the probability distribution of the node movement direction is also a stochastic Markovian process, which means that nodes tend to maintain the same or similar direction.

### 3.1 Description of the Markovian Model

The model uses two continuous time Markov chains that rule the nodes movement behavior in directions $x$ and $y$. We assume that the transitions in the Markov chains ruling the movement for one direction is independent of those in the Markov chain used for the other direction. The state space of the Markov chain in direction $x$ is given by I={-N, -N+1, ... , -1, 0, 1, ... , N-1, N}. The state space in direction $y$ is given by J={-M, -M+1, ... , -1, 0, 1, ... , M-1, M}. Furthermore, if we define $n \in \mathbb{Z}$ as the index of the state in the chain that represents the movement in the $x$-axis, and $m \in \mathbb{Z}$ as the index of the state in the chain that models the movement in $y$, then from any state $n$ (or $m$), only transitions in the neighborhood of states $n-1$ ( or $m-1$) and $n+1$ (or $m+1$) are allowed.

Given certain state $n$ on direction $x$ (or an state $m$ on direction $y$), the rate in which this state changes to a state $n+1$ (or $m+1$) is given by $\lambda_n$ (or $\lambda_m$). Moreover, the rate that a state $n$ (or $m$) changes to a state $n-1$ (or $m-1$) is $\mu_n$ (or $\mu_m$). The rates $\lambda_n, \lambda_m, \mu_n$ and $\mu_m$ are defined as

$$\lambda_n = \begin{cases} 2\lambda, & n = 0 \\ \lambda, & -N < n < N, n \neq 0 \\ 0, & n = -N \quad \text{or} \quad N \end{cases} \tag{1}$$

$$\mu_n = \begin{cases} \mu, & -N \leq n \leq N, n \neq 0 \\ 0, & n = 0 \end{cases} \tag{2}$$

$$\lambda_m = \begin{cases} 2\lambda, & m = 0 \\ \lambda, & -M < m < M, m \neq 0 \\ 0, & m = -M \quad \text{or} \quad M \end{cases} \tag{3}$$

$$\mu_m = \begin{cases} \mu, & -M \leq m \leq M, m \neq 0 \\ 0, & m = 0. \end{cases} \tag{4}$$

The state transition diagrams of each chain are represented in the Figure 1.

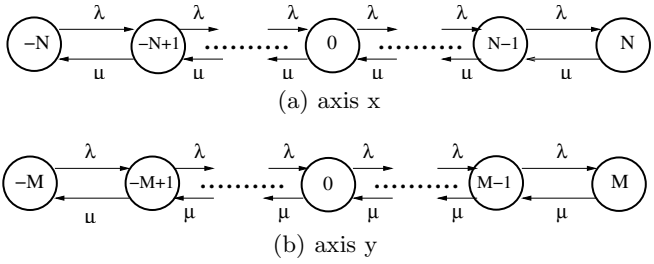



**Fig. 1.** Markov chains that rules movement in: (a)$x$ axis, (b)$y$ axis

### 3.2 Description of Speed and Direction of Movement

To obtain the speed of the node from the Markov chain state for each direction ($x$ and $y$), we make the following definitions

$$V_x(n) = \begin{cases} b^n, \text{ if } & n > 0, \\ -b^{|n|}, \text{ if } & n < 0, \\ 0, \text{ if } & n = 0, \end{cases} \tag{5}$$

$$V_y(m) = \begin{cases} b^m, \text{ if } & m > 0, \\ -b^{|m|}, \text{ if } & m < 0, \\ 0, \text{ if } & m = 0. \end{cases} \tag{6}$$

Similar to GIMM model [5], $b$ is a constant parameter, that can be tunned according to a desired mobility profile. For larger values of $b$, the range of values the speed can assume is also larger. Essentially, $b$ is the base of the number used to represent the mobile node speed. We make the speed exponentially related to the state of the chain to represent a broader range of speed without the necessity to make a Markov chain with too many states.

Using vectorial properties, we can obtain a speed vector, $\mathbf{V}(\mathbf{n}, \mathbf{m})$, as a function of $V_x(n)$ and $V_y(m)$

$$\mathbf{V}(n, m) = (V_x(n), V_y(m)). \tag{7}$$

The speed modulus in rectangular coordinates is given by

$$|\mathbf{V}|(x, y) = \sqrt{V_x(n)^2 + V_y(m)^2}. \tag{8}$$

Substituting (5) and (6) in (8) we have the speed modulus as a function of states $n$ and $m$ in the following equation

$$|\mathbf{V}|(n, m) = \begin{cases} \sqrt{b^{|2n|} + b^{|2m|}} & \text{if } n \neq 0 \text{ and } m \neq 0 \\ \sqrt{b^{|2n|}} & \text{if } n \neq 0 \text{ and } m = 0 \\ \sqrt{b^{|2m|}} & \text{if } n = 0 \text{ and } m \neq 0 \\ 0 & \text{if } n = 0 \text{ and } m = 0. \end{cases} \tag{9}$$

Once defined the speed modulus (9), the maximum speed modulus is given by $|V_{max}| = |\mathbf{V}|(N,M) = \sqrt{b^{|2N|} + b^{|2M|}}$. The movement direction as a function of states $n$ and $m$, is given by $\theta(n,m) = arctan\left(\frac{V_y(m)}{V_x(n)}\right)$. Negative values of $V$ denotes a negative direction sense. As the speed in $x$-axis varies between $-b^{|-N|}$ and $b^N$, we can say that the chain, in $x$-axis, has $2N+1$ states; and the speed in $y$-axis varies between $-b^{|-M|}$ and $b^M$, so the chain in $y$-axis, has $2M+1$ states.

Knowing that the system is Markovian, the probability that the system remains in the same state is only a function of the current state and is independent of how much time the system remains in that state. Thus, it is possible for us to say that the remaining time distribution in a given state is memoryless [10]. In continuous time, the only random variable with this property is the exponential, thus we use this random variable to represent the time between state transitions. When the speed component is maximum, it can only be reduced. This time is exponentially distributed with mean $\frac{1}{\mu}$, when $n = N$ or $m = M$, and $\frac{1}{\lambda}$, when $n = -N$ or $m = -M$. In all other states, we can say that the time that a mobile station moves in a given speed is a random variable exponentially distributed with average $\frac{1}{\lambda+\mu}$. In this way, the rate in which the station changes its speed, in a given direction, is given by $\lambda + \mu$.

Assuming that transitions in each directions are independent, we can merge the two independent chains resulting in a single Markov chain with two parameters, controlling the vectorial speed. The state transition diagram represented in Figure 2 illustrates how the model works.

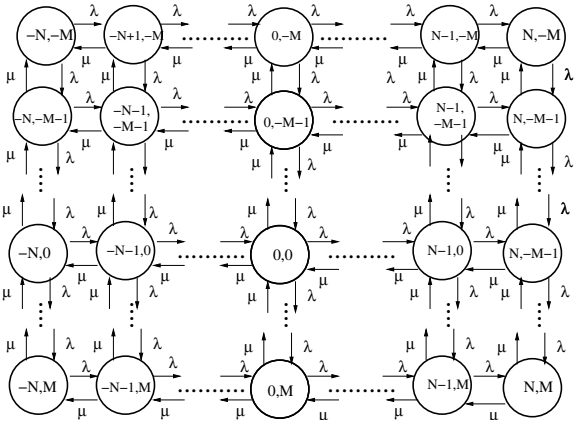



**Fig. 2.** State diagram with two-state variables

By adjusting the values of $b, \lambda, \mu, M$ and $N$ we can represent different mobility profiles. In Figure 3 we show two examples, where Figure 3a shows a profile with high speed and Figure 3b, shows a profile with high number of changes in direction. So, the proposed model was described and its characteristics were presented in this section. In the next section we present an evaluation of proposal.

## 4 Mobility Evaluation Results

As discussed in previous sections, it is not realistic for a mobile node to assume movement velocities and directions in a transition independently of movement velocities and directions adopted in previous transitions, in other words, realistic movement should occur with sense of direction and not randomly. Furthermore, according to the metrics defined in this section, we show that our model presented a much more realistic behavior than the other mobility models studied.

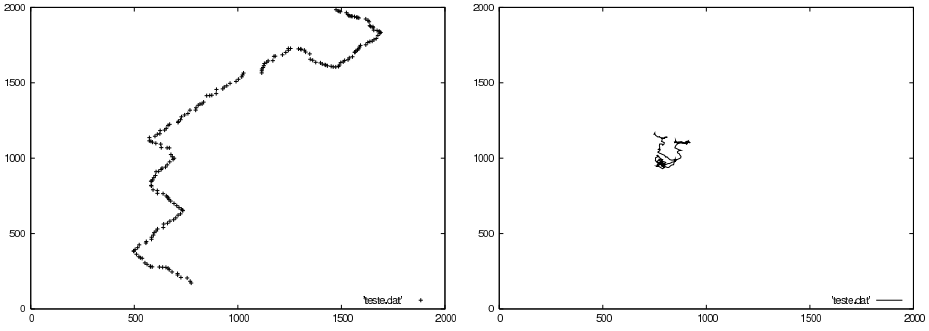



**Fig. 3.** b=2,N=M=4, Simulation Time=200s: (a) profile with high speed ($\lambda = 2$, $\mu = 1$), (b) profile with high changes in direction ($\lambda = \mu = 10$)

We will evaluate our proposed model by comparing it with only two of the mobility models mentioned in Section 2. We use the most used random model, the RWP, its version without the transient phase (RWP_PS), described in [11], and a Markovian mobility model that is able to drive a mobile node with some sense of direction, the GIMM.

### 4.1 Mobility Evaluation

We define four metrics for comparing and evaluating the mobility traces generated by the three studied models. **Total number of sharp turns** - a change in direction is considered a sharp turn, when the module of the difference in the movement direction angle in a given instant $t$ and the direction angle in the instant $t + dt$ is between the interval $[90^0, 180^o]$. **Normalized number of sharp turns** - is the total number of sharp turns divided by the total number of changes in direction occurred. **Total number of sudden accelerations** - a change in speed is considered high, if the module of the difference in nodes speed in a given instant t and the speed in the instant $t+dt$ is superior to 50% of $V_{max}$. This difference can be positive or negative, and if it falls into this restriction, it is considered a sudden acceleration. **Normalized number of sudden accelerations** - is the total number of sudden accelerations divided by the total number of sudden accelerations occurred.

The metrics concerning changes in movement direction are considered assuming that a user in the real world, rarely changes its direction, at once, with an angle superior to $90^o$. This is because we assume that, if the user is moving with a sense of direction, it will not change dramatically its current direction in the near future. In the same way, when a variation in speed occurs, it does not occur at once. Velocity varies gradually in the real world, and a realistic mobility model should avoid sudden accelerations.

The proposed model was implemented in C language. The simulation scenario implemented to perform this evaluation is composed by a 700 meters X 500 meters area where 50 nodes were initially uniformly distributed. There were 50 simulation runs at a confidence level of 95%. The comparisons were made between the RWP model, the RWP without the transient phase (RWP_PS), described in [11], the GIMM model and the proposed model. The RWP model was configured with an uniformly distributed speed between $[0.1m/s, 12.1m/s]$ and pause time equal to zero. The same parameters were used with the RWP without transient. In the GIMM model, the parameters were adjusted so that the maximum speed could be $12.1m/s$. In the proposed model, in order to achieve the same $12.1m/s$ maximum speed, we used the base $b$ equal to 1.43, with 6 states, $\lambda = 1$ and $\mu = 1$.

Figure 4 illustrates the results of the metric *total number of sharp turns*. According to the assumptions made in Section 3, in order to represent a more realistic behavior, nodes should move with a sense of direction, which means avoiding sharp turns. As we can notice in this figure, GIMM is the mobility model with more sharp turns. RWP is the model that presented the least number of sharp turns, while the proposed model presented an intermediary result. This is explained by the fact that, in the same period of time, the GIMM model performs much more changes in direction than the RWP. A node in a RWP simulation can choose a far way waypoint and a small speed value, and stay stuck in the same trajectory the whole simulation, never changing its direction. Thus, in absolute values, the RWP presents a smaller number of sharp turns, but when we consider the *normalized number of sharp turns* in Figure 5, we can

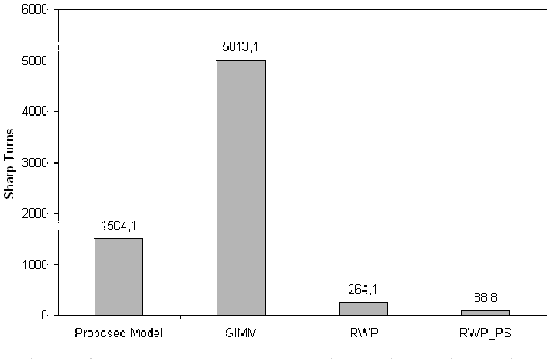



**Fig. 4.** *Total number of sharp turns* as a function of the mobility model

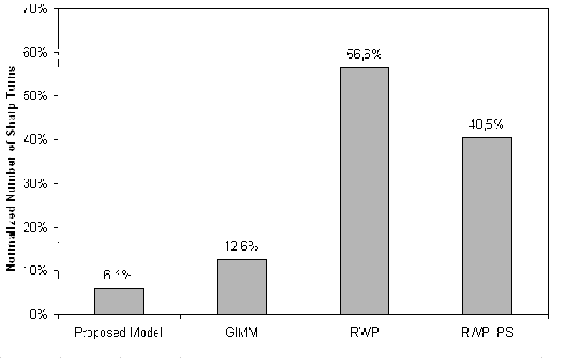



**Fig. 5.** *Normalized number of sharp turns* as a function of the mobility model

see that 56% of all the changes in direction when using RWP are sharp, a much larger value than the ones presented by the Markovian models.

The difference between the proposed model and the GIMM model is also big. This is due to the fact that the Markov chain that rules the motion behavior in the GIMM is in discrete time, while the chain in the proposed model is in continuous time. In a discrete time chain it is possible for two events to occur at the same instant of time, which cannot happen while using a continuous chain. Once more than one simultaneous transition is possible, it is more probable that much more changes are going to occur when using the GIMM model.

Figure 6 presents the results of the *total number of sudden accelerations* metric for the evaluated models. As discussed previously in Section 3, realistic mobility models should avoid sudden changes in speed. By analyzing this figure, we can see that GIMM is once again the mobility model presenting the highest number of sudden accelerations. RWP and the proposed model presents almost the same results. As done with the previous metric, it is necessary to look at the normalized quantity of sudden accelerations. Observing Figure 7 we can notice that

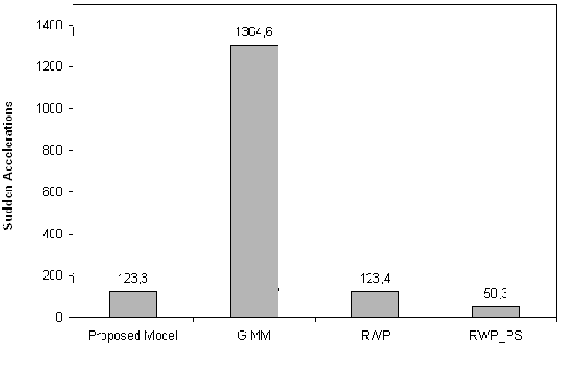



**Fig. 6.** *Total number of sudden accelerations* as a function of the mobility model

the RWP, once again, presented the higher value of sudden accelerations. The proposed model presents less sudden accelerations than GIMM and a similar value of this metric when comparing to the RWP in absolute values.

### 4.2 Application Evaluation

In order to evaluate the performance of an application running over an ad hoc network moving according our model, we considered another simulation scenario and two more metrics: packet delivery rate and routing overhead. **Packet delivery rate** is the ratio between the number of packets received by the destination and the total number of packets sent. **Routing overhead** is the ratio between the number of routing packets transmitted through the network and the total number of data packets sent by the source.

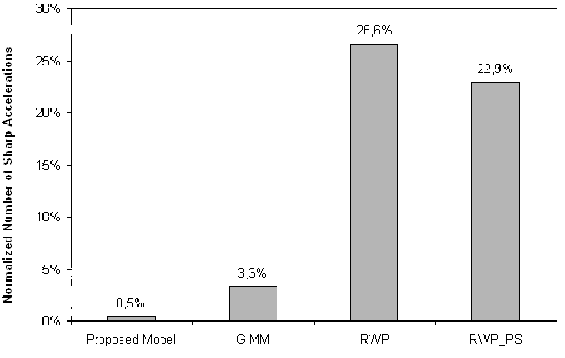



**Fig. 7.** *Normalized number of sudden accelerations* as a function of the mobility model

The simulation was performed using an area of $1500m \times 1000m$ with 50 nodes moving around according to a specific mobility model. The mobility models used were the RWP, GIMM and the proposed model. Simulations took $500seconds$ and the NS-2 simulator [6] was used. Nodes maximum velocities were fixed in $2m/s$. Parameters of the proposed model were $[\lambda,\ \mu,\ b,\ n,\ m] = [0.01,\ 0.01,\ 1.1,\ 4,\ 4]$ and for GIMM we used $[m,\ b,\ n] = [0.4,\ 1.1,\ 4]$. For the RWP model the parameters were $[Meanspeed,\ Minspeed,\ pausetime] = [1.5,\ 0.5,\ 0]$. For traffic variations, we used 10, 20 and 40 simultaneous CBR connections, where 4 packets with 512 bytes were generated each second. The transport protocol used was the UDP, to avoid the influence of TCP the congestion control, which could affect the generated traffic.

The routing protocol used in this evaluation was the AODV [3], because it is a well known and widely used routing protocol. Instead of using various routing protocols, we decided to use just the AODV and vary the traffic so that we could have a better comprehension of the influence of the proposed model in routing for the specific case of the AODV. The medium access control protocol used was the standard IEEE 802.11 in DCF mode with RTS/CTS. Transmission range was

adjusted for 250 meters. To model the physical layer, the propagation model used in the simulation was the Two-Ray-Ground. A total of 30 simulations were performed for each number of active connections, using each mobility model. The obtained results have a confidence level of 95%.

Figure 8 shows a big variation when changing the used mobility model. With 10 active connections, the proposed model presented a routing overhead 5 times higher than the RWP and 3 times higher than GIMM. This big difference is due to the mobility profile configured that made the nodes using the proposed model roam around the simulation area more than the nodes moving according to RWP and GIMM. This behavior influences the spatial density. More roaming, means a less dense network, which allows more route disruption, and consequently, large routing overhead.

It is also interesting to observe how the routing overhead metric varies as the number of active connections grow, when the used mobility model is the one proposed in this work. For GIMM and RWP models this overhead is almost constant, no matter how many active connections.

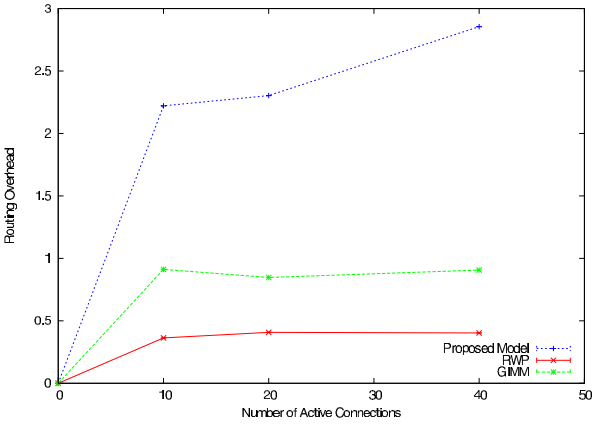



**Fig. 8.** Impact of mobility model over the *routing overhead* when the maximum speed is limited to 2m/s

In Figure 9 it is possible to observe that the packet delivery rate metric remains constant for the GIMM and RWP models, no matter what number of active connections. When the proposed model is used, a little drop in the delivery rate is computed, as the number of active connections increase. However, this variation is negligible when compared to the variation presented by the routing overhead. As the AODV routing protocol is a reactive one, there are only routing messages exchanged when there is the necessity of discovering new routes. This means that the AODV adjusts to the great number of routes loss that occurs when using the proposed model, without losing to much packets. The high routing overhead is justified by the delivery rate maintenance.

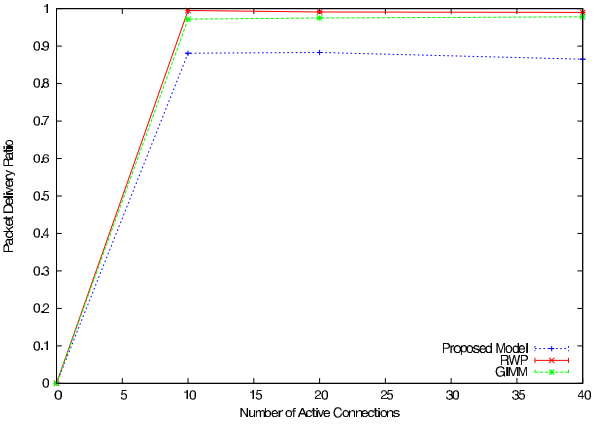



**Fig. 9.** Impact of mobility model over the *packets delivery ratio* when the maximum speed is limited to 2m/s

## 5 Conclusions and Future Work

In this work a new Markovian mobility model based on a Birth-Death process was proposed. This model assumes that the nodes move with a direction sense and not randomly. Thus, the model allows smoother movements in the same and adjacent directions, speed changes and pauses.

In order to evaluate the proposed mobility model, we presented a comparison among mobility models through some performance metrics related to sharp turn in the movement. This evaluation showed that our model achieved better results, when compared to RWP and GIMM models, considering the metrics: normalized number of sharp turns and normalized number of sudden accelerations. We also present a performance evaluation of AODV routing protocol when applying some mobility models: the proposed model, the GIMM model and the RWP model, in which the packet delivery rate and the routing overhead were computed.

As future work, we intend to perform the following: (i) to derivate closed formulas for the steady-state distributions of movement speed and direction; (ii) to derivate the steady-state distribution of nodes density for the proposed model, and, (iii) to capture real traces of pedestrian movement through GPS devices and to compare this real mobility profile with the proposed model.

## References

1. Baumann, R., Legendre, F., Sommer, P.: Generic Mobility Simulation Framework GMSF. In: ACM MobilityModels. Hong Kong, China (2008)
2. Bettstetter, C.: Smooth is better than sharp: a random mobility model for simulation of wireless networks. In: MSWIM'01. pp. 19–27 (2001)
3. Broch, J., Maltz, D.A., Johnson, D.B., Hu, Y.C., Jetcheva, J.: A performance comparison of multi-hop wireless ad hoc network routing protocols. In: ACM/IEEE MOBICOM'98. pp. 85–97 (1998)

4. Camp, T., Boleng, J., Davies, V.: A survey of mobility models for ad hoc network research. Wireless Communications & Mobile Computing 2(5), 483–502 (2002)
5. Campos, C.A.V., de Moraes, L.F.M.: A markovian model representation of individual mobility scenarios in ad hoc networks and its evaluation. EURASIP Journal on Wireless Communications and Networking 2007 (jan 2007)
6. Fall, K., Varadhan, K.: The NS manual, `http://www.isi.edu/nsnam/ns/ns-documentation.html`, last access in 15/04/2011
7. Haverkort, B.R.: Performance of Computer Communication Systems: A Model-Based Approach. John Wiley & Sons, New York, USA (1998)
8. Jardosh, A., Belding-Royer, E.M., Almeroth, K.C., Suri, S.: Towards realistic mobility models for mobile ad hoc networks. In: MobiCom'03. pp. 217–229 (2003)
9. Kim, S., Lee, C.H., Eun, D.Y.: Super-diffusive Behavior of Mobile Nodes from GPS Traces. In: ACM Mobicom Poster Abstract 2007 (2007)
10. Kleinrock, L.: Queuing Systems, vol. 1. John Wiley & Sons Publishers (1975)
11. Le Boudec, J.Y., Vojnovic, M.: Perfect Simulation and Stationarity of a Class of Mobility Models. In: INFOCOM 2005 (2005)
12. Musolesi, M., Mascolo, C.: Designing Mobility Models Based on Social Network Theory. ACM M2CR 11(3), 59–70 (2007)
13. Rhee, I., Shin, M., Hong, S., Lee, K., Chong, S.: On the Levy-walk Nature Human Mobility. In: IEEE Infocom'08. Phoenix, USA (2008)

# Lossy Data Aggregation with Network Coding in Stand-Alone Wireless Sensor Networks

Tatiana K. Madsen

Networking and Security Section, Dept. of Electronic Systems,
Aalborg University, Denmark
tatiana@es.aau.dk
http://es.aau.dk/netsec

**Abstract.** This work focuses on a special type of wireless sensor networks (WSNs) that we refer to as a stand alone network. These networks operate in harsh and extreme environments where data collection is done only occassionally. Typical examples include habitat monitoring systems, monitoring systems in chemical plans, etc. Given resource constrained operation of a sensor network where the nodes are battery powered and buffer sizes are limited, efficient methods for in-network data storage and its subsequent fast and reliable transmission to a gateway are desirable. To save scarse resources and to prolong the life time of the whole network, the lossy data aggregation method can be applied. It is especially viable in the networks where several sensors are measuring the same physical phenomenum and only average values of sensor readings are of interest. In this paper, we present a method for effiecient lossy data aggregation using a network coding technique. We demonstrate that with low communication and processing overhead, a gateway can retrieve data fast from the network and the retrieved values are close to the sample means of the collected measurements. This method requires a special choice of coefficients for linear data encoding: the condition number of the coefficient matrix should be small and close to one.

**Keywords:** wireless sensor network, lossy data aggregation, network coding.

## 1 Introduction

Wireless sensor networks (WSNs) consist of nodes that have both sensing and communication capabilities [1]. Nodes are deployed over an area monitoring certain events or physical phenomena. The measured data collected by individual nodes should be transmitted to some central entity for further processing and analysis. Therefore, nodes send their data to common sinks that act as gateways to other networks. These many-to-one communication flows are typical for WSNs.

Some WSNs operate in a stand-alone fashion. This is typically the case for networks deployed in harsh environments or in remote regions where constant data retrieval is not possible or is costly. In such situations a gateway (or a sink) is not

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 251–261, 2011.

always present in a network but is available for communication only occasionally, be it with some regular time interval or at times not specified beforehand. Fig. 1 illustrates the network topology. Examples of such system can be a sensor field located in a desert or on an ocean surface where data readings are done from a helicopter or from a boat. The absence of a sink in a network leads to a need for temporal data storage until the measurements can be transmitted to a gateway. In such WSNs nodes are saving measurement data in their buffers. These settings also lead to the problem with reliability of data storage due to both limited buffer size and potential failure of individual nodes. To overcome these problems cooperative behavior of the nodes can be employed where we allow for inter-node communication and pre-processing of information received from neighboring nodes before storing it.

Approaches for in-network data aggregation are also proposed for WSNs for the purpose of energy efficiency. Instead of transmitting raw data to sink, energy can be saved and communication overhead and network interference can be reduced if partially aggregated data is transmitted [2]. Two different types of aggregation are possible: *lossless* aggregation and *lossy* aggregation [3]. Lossless aggregation refers to the situation when no data is lost and aggregation means concatenating individual readings into larger packets. In case of a stand-alone WSN it means that data packets from neighboring nodes are exchanged and stored without any modifications. Lossless aggregation is effective only if the load on the system is small.

Lossy aggregation, as the name indicates, is done with some loss of information. This is unavoidable in cases when the total communication load to a gateway exceeds system capacity or in cases when the amount of information to be stored exceeds available buffer sizes. In both cases the amount of communicated / stored data must be forcibly reduced. An example of lossy aggregation is averaging of sensor values that is a natural choice in many applications where only an average value is of interest. Only the averaged value is communicated to the gateway, thus reducing the time and energy needed for sensor nodes-to-gateway communication.

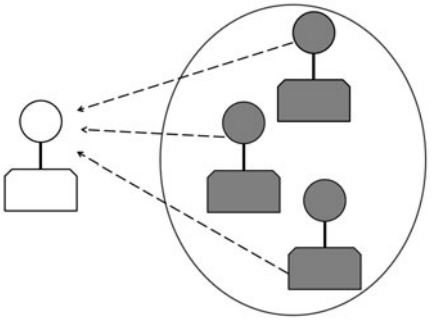


**Fig. 1.** Gateway receives data transmissions from sensors in a WSN

Averaging can be done either temporally or spatially. For temporal averaging, data from the same sensor is averaged over time. This does not require inter-node communication and thus, does not introduce an additional communication burden on the system. However, this type of averaging might not be suitable for many applications where a value change in time is of key interest. In many cases a number of identical nodes will be distributed over an area all measuring the same value. As each node's measurements will have some error due to equipment imprecision, and supposing that the measurements are independent, identically distributed normal random variables, the maximum likelihood estimator for the mean will be the sample mean. Averaging data over measurements from multiple sensors will give us a sample mean, however the clear disadvantage of this approach is the fact that it requires a large amount of inter-network communication as measurement data should be exchanged between the nodes before averaging can be made.

The problem becomes worse as the number of nodes in a network grows. If something is known about the spatial correlation of data, extensive inter-network communication can be avoided. This can be done by applying Distributed Source Coding (DSC), a compression technique that removes data redundancy when each sensor has a priori knowledge of the correlation structure that depends on the distances between the sensors. Different methods for application of DSC for data aggregation in WSNs have been developed in [4], [5]. It has been demonstrated that this technique can be successfully applied in cluster-based WSNs. However, it requires that the spatial correlation structure of the network is given in advance that is not always a viable assumption.

In this paper we propose a method for spatial averaging that does not require any pre-knowledge about network structure and uses a very limited data exchange between sensor nodes. It is based on the *network coding* technique [6] that has already shown its benefits in different areas of communication. Here we will use it in a completely new way that to the best of our knowledge has not been yet considered.

Network coding can be viewed as a generalization of the conventional routing method based on the store-and-forward principle. Network coding allows for individual packets to be combined and re-combined at any point in the network, thus encoding previously received input data. At a point of data reception (it can be both at an end point and at an intermediate node in a network) the decoding procedure is performed to reconstruct the original data. For WSNs, network coding has been introduced in [7] for distributed storage. We will furthermore extend the usage of network coding in WSNs and show that it can be successfully applied for lossy data aggregation.

The rest of the paper is organized as follows. Section 2 introduces the details of the considered scenario. Section 3 provides the neccessary information about network coding to allow the reader to understand how using linear data combinations can improve the performance of a WSN. A novel approach for lossy data aggregation is presented in Section 4. This section also shows how

the coefficients for encoding matrix should be chosen and illustrates this with an example. Section 5 gives concluding remarks and gives an outlook for further work.

## 2 Scenario

We consider a sensor network deployed in a remote and inaccessible environment where sensors are measuring and storing data in the network over long time intervals. A gateway that have a function of a data collector, may appear at any location and time in the network and will try to retrieve as much data as possible. In the following, we refer to this settings as a *stand-alone* WSN. We assume that several different phenomena should be observed, e.g. temperature, pressure and humidity. Thus, the sensors are divided into groups, all sensors in a specific group perform the same kind of operation. For simplicity, we assume that each group contains the same number of nodes, that is we have $n$ groups with $m$ nodes in each group. This assumption can easily be relaxed, however we will use it throughout the paper in order to keep the presentation simple.

Furthermore, let us assume that all nodes are within communication range of each other. This assumption allows us to abstract from the problem of routing information between the nodes and the gateway. The measurements are done by each node periodically and the recorded data is saved in memory until a gateway arrives and there is a possibility to transmit the collected data and empty the buffer. We are interested in temporal behavior of the observed phenomena. As readings of different sensors measuring the same value might be imprecise, we are interested in knowing an average value at each time instant when the measurements were done.

After saying all this, the thing that is left to be defined is an algorithm specifying how the collected data is stored and how it is transmitted to a gateway. Designing such an algorithm, which is a goal of our work, one should keep in mind the following criteria:

- Reliability: if some sensors in the network have stopped functioning before the gateway arrives, their measurement data will be lost, unless it is distributed to other nodes in the network. Thus, inter-network communication for reliability purposes is desirable.
- Energy efficiency: this is a key parameter for any WSN as nodes are battery powered and the life time of the network should be as long as possible.
- Fast transmission of data to gateway: a gateway might be not available in the network for a long period, therefore data aggregation before a gateway appears in the network is desirable. Additionally, encoding of information to be transmitted to a gateway might reduce the time needed to transmit all available data. Additionally, the energy cost of transmission from a sensor node to a gateway is typically larger compared to in-network communication due to a large distance to the gateway.
- Limited buffer size: buffer utilization should be optimized. Additionally, a method to remove obsolete data should be applied.

The straightforward approach for lossy aggregation is when a node receives all measurements from its group via sensor-to-sensor communication, computes and stores only an average value. By the request from a gateway, this value is transmitted to it. The disadvantage of this approach is a large in-network communication overhead where a node has to receive $m-1$ messages before it can make the calculations. In our approach presented in Section 4 the amount of in-network overhead is significantly reduced. The nodes across different groups will cooperate to store the collected information. A node should receive just one measurement from any other group. The received data is encoded using network coding principles and stored. Before we proceed with the detailed description of our approach, we introduce the basics of network coding theory.

## 3 Network Coding Basics

In this section we provide the reader with some basic information to understand network coding. We give the necessary details about linear network coding and show its application in WSNs.

Network coding was first introduced in [6] to improve the performance of multicast routing. Since then, many analytical studies and simulation evaluations advocate the usefulness of the network coding approach in terms of network throughput improvement. An example that has already become classical is shown in Fig. 2. It demonstrates that the load on a shared node can be reduced when coding of information is applied. Nodes $s_1$ amd $s_2$ want to exchange data using node $s_3$ as relay. In the traditional way of implementing a relaying procedure, shown on the left side of Fig. 2, 4 transmissions are needed. However, if packets from $s_1$ and $s_2$ are combined at $s_3$ using a simple XOR operation, both end nodes will be able to recover the data and only 3 transmissions are needed.

Generally, let us denote as $x_i$ the packet generated by a node $i$. The information can be "mixed", thus giving us a new coded packet $b = f(x_1, \ldots, x_k)$ where $k$ is the number of "old" packets. It has been shown [8] that linear functions suffice in many cases.

In *linear network coding* function $f$ is linear and one works with coding (or coefficient) vectors $c_j$ that presents coefficients used for linear combinations of packets:

$$c_{1,j}x_1 + \ldots + c_{k,j}x_k = b_j \tag{1}$$

Notice that the size of $b_j$ remains equal to $x_i$ when the operations reside in a Galois field.

At least $k$ encoded packets $b$ are required to invert the matrix and solve equations (1) for $x_i$. Thus, the coefficients should be pre-defined in advance in a way such that the coefficient matrix is invertible and the vectors $c_j$ should be distributed in the network. Often centralized management of the coefficient matrix is difficult to implement or it requires a lot of communication overhead. The way to overcome the central control is to allow each node to select by itself randomly coefficients from a finite field. This way of encoding is called *random linear network coding*. It is shown [9] that if the finite field is chosen to be large

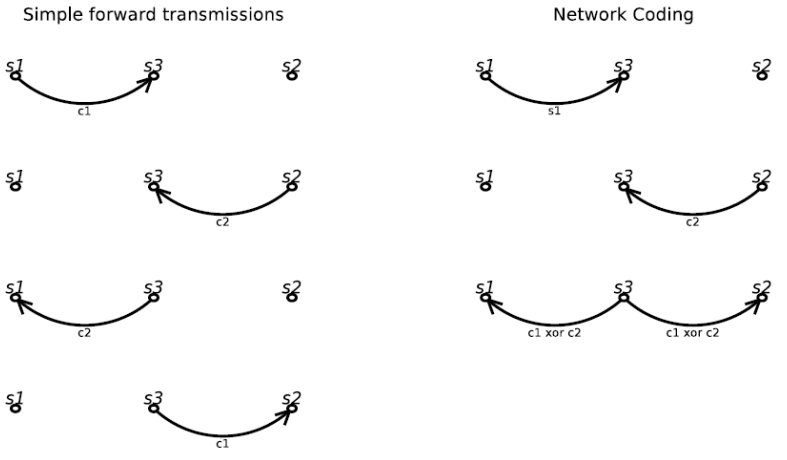



**Fig. 2.** On the left, an intermideate node works as a relay and emplys a simple store-and-forward mechanism for data transmission. On the right, it combines 2 data packets by using XOR operation and then broadcasts it. Both end nodes are able to recover the data from the combined packet.

enough, then the linear dependency of the coefficients is negligible. In this case, the coefficent vector should be transmitted together with the encoded packet, since the coefficiemts are not known at the destination side. This increases the size of the packet, but does not require communication for a central set up of the coefficient matrix.

One of the problems of applying network coding in WSNs is the lack of support for removing obsolate data. In many practical applications, the new data has higher value than the old ones. If a gateway does not arrive, the sensor network needs to remove old data in order to accomodate for the new data. In case of no coding this problem is non-existing as the oldest packet can just be overwritten with the new one. However, when network coding is applied, the coded packets have to be decoded first; the oldest packets removed and then the remaining packets re-encoded into new packets. This procedure is computationally expensive. To overcome this problem, *partial network coding* is presented in [10] that generalizes the network coding paradigm and allows efficient storage replacement of old data with new data without the need for re-encoding. This requires a slight increase in the number of coded packets needed to extract the original data. The details of this approach and its implementatin on a wireless sensor board can be found in [10], [11].

Applying linear network coding in WSNs can basically bring the following 2 benefits:

– Efficient distributed data storage: a single sensor node is not capable of storing all data generated in the network. It can neither perform complicated computational operations. Computing linear combinations is visible

on ordinary, commercially available, sensor boards (see, e.g., [11]). Such an operation combines all data segments from other nodes in a network. One should note that the coded packets are equivalent to each other in decodeability, i.e. in order to decode the original data, consisting of $k$ packets, *any* $k$ linear combinations of these packets can be used.
– Fast and load balanced data collection: As long as we are interested in receiving any $k$ coded packets at a gateway, we do not have to keep track of what packets are stored at which node, but the gateway will contact any sensors until it will receive the sufficient number of coded packets.

At the end of this section, we would like to illustrate how network coding can be applied to our scenario. Let us assume that each node makes a measurement once per day and these measurements are exchanged among the nodes. Let $x_{ij}$ denote a value measured by a node belonging to the group number $i$ and the node number in the group is $j$. Each node computes a linear combination of all $x_{ij}$ using either in advance pre-defined coefficients or selecting random coefficients. The gateway has to collect $n \times m$ coded packets and the system of linear equations that should be solved looks like:

$$\begin{cases} \alpha_{11}x_{11} + \ldots + \alpha_{1\ n\times m}x_{mn} & = b_1 \\ \vdots & \\ \alpha_{n\times m\ 1}x_{11} + \ldots + \alpha_{n\times m\ n\times m}x_{mn} & = b_{n\times m} \end{cases} \tag{2}$$

This way of encoding information will correspond to implementation of lossless data aggregation. When the number of groups $n$ and the number of nodes per group $m$ are small, this is a viable approach that allows extraction of complete information from the network. However, when $n$ and/or $m$ grows, communication overhead for in-network information exchange and processing time grow as well, making this approach less and less attractive. Given constrains on resources in a WSN, for large $n \times m$ an efficient way for lossy data aggregation should be found.

## 4 Lossy Data Aggregation with Network Coding

In this paper we suggest the following approach for lossy data aggregation: each node keeps its own measurement and receives one measurement from other groups. Thus, in total each nodes have $n$ data segments for coding. This is illustrated in Fig. 3. For simplicity of notations we assume that a node $s_{ij}$ receives data from nodes with number $j$. We obtain the following system of linear equations:

$$\begin{cases} a_{11}x_{11} + \ldots + a_{n1}x_{n1} & = b_1 \\ \vdots & \\ a_{1m}x_{1m} + \ldots + a_{nm}x_{nm} & = b_n \end{cases} \tag{3}$$

This approach reduces the amount of communication overhead. Additionally, it allows for implementation of fast transmission to a gateway as only $n$ packet

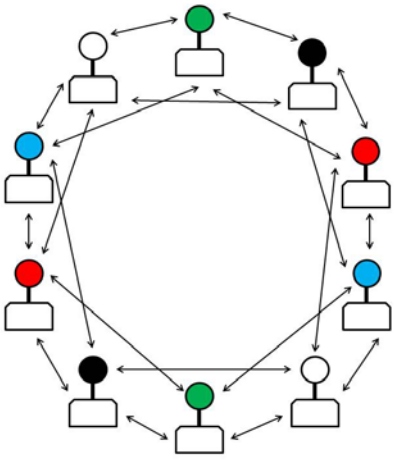


**Fig. 3.** The system shown in the figure consists of 10 sensors, 5 groups with 2 sensors in each. The arrows indicate which sensors exchange data with each other. Every sensor saves a single measurement from every other group.

tramnsmissions are needed. The system (3) as it is written is underdefined and the unique solution can not be found. However, assuming that all unknown $x$ with the same first index are equal, we have $n$ linear equations with $n$ unknowns that can be solved if the coefficient matrix $A$ is invertible. Now, the question is, if the solution vector $\mathbf{x}$ is found as described above, how close it will be to the average value of $x_{ij}$. We will prove that if the condition number of matrix $A$ [1] is small, then the solution is close to the average value.

Let $\bar{\mathbf{x}}$ be the vector of average values, and the right-hand side vector $\bar{\mathbf{b}}$ is calculated as $\bar{\mathbf{b}} = A\bar{\mathbf{x}}$. If $det(A) \neq 0$, then $\bar{\mathbf{x}} = A^{-1}\bar{\mathbf{b}}$. We can also write it in the matrix form:

$$\begin{bmatrix} a_{11} & \dots & a_{n1} \\ & \ddots & \\ a_{m1} & \dots & a_{m1} \end{bmatrix} \quad \begin{bmatrix} \bar{x_1} & \dots & \bar{x_1} \\ & \ddots & \\ \bar{x_n} & \dots & \bar{x_n} \end{bmatrix} \quad = \quad \begin{bmatrix} b_1 & \dots & b_1 \\ & \ddots & \\ b_n & \dots & b_n \end{bmatrix}, \tag{4}$$

or in a short form $A\bar{X} = \bar{B}$.

Let us introduce a matrix $B$ with elements $b_{ij}$ where $b_{ij} = a_{1i}x_{1j} + \cdots + a_{ni}x_{nj}$. Matrix $X = [x_{ij}]$ composed from measured values can be represented as $X = \bar{X} + \Delta X$. In a similar way we can write $B = \bar{B} + \Delta B$. By definition, $AX = B$. Thus,

$$A(\bar{X} + \Delta X) = \bar{B} + \Delta B \quad \Leftrightarrow A\Delta X = \Delta B \quad \Leftrightarrow \Delta X = A^{-1}\Delta B \tag{5}$$

[1] The condition number for a matrix $A$ is defined as $\|A\| \left\|A^{-1}\right\|$ where $\|\cdot\|$ is the matrix norm. It is denoted $\kappa(A)$. Note that $\kappa(A) \geq 1$ for all $A$.

**Statement.** If we choose matrix $A$ in such a way that its condition number is small (i.e. close to one), then the relative error in $X$ is not much larger than the relative error in $B$ and the solution of (3) is close to the measured average values.

*Proof.* Using equation (5) we can write that

$$\|\Delta X\| = \left\|A^{-1}\Delta B\right\| \leq \left\|A^{-1}\right\| \|\Delta B\| \tag{6}$$

From $\bar{B} = A\bar{X}$ we have

$$\|A\| \left\|\bar{X}\right\| \geq \left\|\bar{B}\right\| \quad \Leftrightarrow \quad \left\|\bar{X}\right\| \geq \left\|\bar{B}\right\| \|A\|^{-1} \tag{7}$$

Thus,

$$\frac{\|\Delta X\|}{\left\|\bar{X}\right\|} \leq \frac{\left\|A^{-1}\right\| \|\Delta B\|}{\|A\| \left\|\bar{B}\right\|} \leq \|A\| \left\|A^{-1}\right\| \frac{\|\Delta B\|}{\left\|\bar{B}\right\|} = \kappa(A) \frac{\|\Delta B\|}{\left\|\bar{B}\right\|} \tag{8}$$

This completes the proof.

Note that the presented proof is somewhat similar to the one in [12], e.g., however instead of working with vectors, we deal with matrixes.

Describing the proposed method, we have made the assumption that a node receives a packet from a node with the same number from another group. In practice it might be easier to allow a node to save the first measurement it received from another group. This will simplify the in-network distribution process and if the packet broadcasting is done at sufficiently random time of day, then every measurement will be stored by some other node with high probability. As one can see the proof stays the same, as well as the main conclusion that the proposed method will work if encoding coefficients are choosen in a certain way. The easiest way to ensure that the condition number of matrix $A$ is small, is to assign encoding coefficients in advance to different sensor nodes.

Finally, we would like to present an example that illustrates how important it is to choose a small correlation coefficient for matrix $A$. Let us assume that $n = 3$ and 3 sensors have received the following measurements of phenomena $Ph_1$, $Ph_2$ and $Ph_3$:

$$sensor_1 : Ph_1 = 2.32, Ph_2 = 35.83, Ph_3 = 79.21$$

$$sensor_2 : Ph_1 = 2.61, Ph_2 = 34.98, Ph_3 = 80.61$$

$$sensor_3 : Ph_1 = 3.57, Ph_2 = 33.53, Ph_3 = 78.48$$

This gives the following sample mean: $P\bar{h}_1 = 2.83$, $P\bar{h}_2 = 34.78$ and $P\bar{h}_3 = 79.43$. Let us consider two coefficient matrixes:

$$A_1 = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 5 & 8 \\ 9 & 3 & 98 \end{bmatrix} \quad A_2 = \begin{bmatrix} 2.5 & 7 & 2 \\ 2 & 2.5 & 7 \\ 4.5 & 2 & 2.5 \end{bmatrix}$$

With the matrix $A_1$ the solution is $(-12.29, 42.45, 79.66)$; the solution corresponding to $A_2$ is $(1.35, 35.75, 80.69)$. The reason for the first solution to deviate

a lot from the average and the second solution being resonable close lies in the condition numbers of matrices $A_1$ and $A_2$: it is 512.28 for $A_1$ and 3.7 for $A_2$. Thus, the matrix $A_1$ is ill-conditioned and can not be used in our algorithm, while matrix $A_2$ is well-conditioned and is suitable as a matrix for encoding coefficients.

## 5 Conclusions

This paper focuses on lossy data aggregation and we have presented a method for lossy aggregation that requires little computational and communication efforts to collect data from the network that is close to the average values of the monitoring events. To achieve this we employ the network coding technique that has already demonstrated its benefits in many application areas of communications. Generally, the usage of network coding is not limited to a specific layer, but could be used at the application, network or physical layer. Here, we use this method at the application layer. The usage of network coding for WSNs already has been considered for distributed storage. Partial network coding has been specifically developed for use in WSNs to solve the problem of removal of obsolete data. We furthermore extend the application area of network coding in WSNs and show that it can be used for lossy data aggregation.

The main contribution of the paper is the identification of in which cases decoding of linear combinations of data segments gives us values close to the sample mean of the original data. We have proven that the condition number of the coefficient matrix should be chosen to be small. It can be easily done when the encoding vectors for all nodes are defined in advance and distributed before the network is put into operation. It still remains an interesting and challenging question if these coefficients can be chosen in a distributed way without any need for central coordination, but still guarantiing that the condition number is small. This is a topic of future work.

## References

1. Mhatre, V., Rosenberg, C.: Design Guidelines for Wireless Sensor Networks: Communication, Clustering and Aggregation. Elsevier J. of Ad Hoc Networks 2, 45–63 (2004)
2. Dai, G., Zhang, J., Tang, S., Shen, X., Lv, C.: Lossy Data Aggregation in Multihop Wireless Sensor Networks. In: 5th International Conference on Mobile Ad-hoc and Sensor Networks, pp. 156–163 (2009)
3. Abdelzaher, T., He, T., Stankovic, J.: Feedback Control of Data Aggregation in Sensor Networks. In: 43rd IEEE Conference on Decision and Control, pp. 1490–1495 (2004)
4. Wang, P., Zheng, J., Li, C.: Data Aggregation Using Distributed Lossy Source Coding in Wireless Sensor Networks. In: IEEE Global Telecommunication Conference, pp. 908–913 (2007)
5. Vuran, M.C., Akan, O.B., Akyildiz, I.F.: Spatio-temporal Correlation: Theory and Applications for Wireless Sensor Networks. Computer Networks Journal 45(3), 245–261 (2006)

6. Ahlswede, R., Cai, N., Li, S.Y.R., Yeung, R.W.: Network Infomration Flow. IEEE Transactions on Infomration Theory 46(4), 1204–1216 (2000)
7. Dimakis, A., Godfrey, P.B., Wu, Y., Wainwright, M., Ramchandran, K.: Network Coding for Distributed Storage Systems. IEEE Transactions on Information Theory 56(9), 4539–4551 (2010)
8. Koetter, R., Medard, M.: An Algebraic Approach to Network Coding. IEEE/ACM Trans. Netw. (2003)
9. Medard, M., Acedanski, S., Deb, S., Koetter, R.: How Good is Random Linear Coding Based Distributed Networked Storage? In: Proc. of NETCOD 2005 (2005)
10. Wang, D., Zhang, Q., Liu, J.: Partial Network Coding: Theory and Application fro Continuous Sensor Data Collection. In: 14th IEEE International Workshop on Quality of Service, pp. 93–101 (2006)
11. Jacobsen, R., Jakobsen, K., Ingtoft, P., Madsen, T., Fitzek, F.: Practical Evaluation of Partial Network Coding in Wireless Sensor Networks. In: 4th ICTS/ACM International Mobile Multimedia Communications Conference (2008)
12. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press, Cambridge (2004)

# An Asynchronous Scheduler to Minimize Energy Consumption in Wireless Sensor Networks

Luca Anchora[1], Antonio Capone[2], and Luigi Patrono[2]

[1] IMT Institute for Advanced Studies Lucca, Piazza San Ponziano 6, 55100 Lucca, Italy
[2] Dept. of Innovation Engineering, University of Salento, Via per Monteroni, 73100 Lecce, Italy

**Abstract.** Next-generation networks are witnessing a convergence of wired and wireless technologies towards a ubiquitous computing paradigm. Wireless Sensor Networks are leading this revolution, but their deployment is delayed mainly because of energy constraints. Sensor nodes are typically equipped with non-rechargeable batteries, thus energy consumption must be minimized. In this paper, an energy-efficient communication protocol, compliant with the ZigBee standard, is proposed and validated. In particular, a new algorithm is discussed that efficiently tunes the duty cycle of a node, i.e., the cycling between an awake and a sleep state of the radio transceiver. After a preliminary communication, each node knows the transmission times of its neighbors and thus when it can switch its radio off since it is not going to receive any message. The solution is robust and has low processing and storage capacity requirements. Simulation results have demonstrated substantial improvements in power consumption compared to the ZigBee standard.

**Keywords:** Wireless Sensor Network, Energy saving, MAC scheduler, Duty cycle, Performance evaluation.

## 1 Introduction

The next-generation Internet aims to integrate heterogeneous communication technologies, both wired and wireless, in order to contribute substantially to assert the concept of *Internet of Things*. Wireless Sensor Networks (WSNs) represent a fundamental alternative able to effectively support many applicative scenarios such as environment monitoring, agriculture, healthcare, and smart buildings.

A WSN is made of many sensor nodes distributed in a zone of interest and equipped with some sensors (e.g., for temperature, pressure, humidity) and with a radio transceiver to communicate with other nodes. In most of the cases, sensed data are gathered by a special entity called *sink*, typically a node with higher capabilities. A very important feature of WSNs is the use of multi-hop communications to allow even nodes very far from the sink to reach it. The only requirement to guarantee network scalability is that the WSN must not be partitioned, i.e., there always exists a path to the sink.

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 262–273, 2011.

Commonly, sensor nodes are battery powered and the battery has a limited capacity. In many cases, due to environmental or cost reasons, batteries can be neither replaced nor recharged and their exhaustion results in the permanent unavailability of the node itself. This event has a twofold impact. On one hand, data collected by that node from the environment are lost. On the other hand, the node is no longer able to forward messages of others and this could imply some route changes in the network. If the number of failures is high, the network might become partitioned. Therefore, energy consumption is a primary issue and effective solutions to increase the battery duration are fundamental to make the lifetime of a WSN high enough to allow its use in real applications. Indeed, in most of the scenarios, network requirements are mainly focused on reliability and lifetime instead of, for example, on delay.

The main activities of a WSN node are sensing, data processing and communication (transmission and reception). The last one is the most remarkable in terms of energy consumption and includes the following phenomena: *message collision*, *overhearing* (i.e., reception of messages addressed to another node), *control packet overhead*, *idle listening* (i.e., listening of the channel when there is no activity on it), and *over-emitting* (i.e., transmission of data towards a node that cannot receive them). For these reasons, most of the works about energy minimization have investigated possible improvements in communication protocols at every layer [1, 2]. A very interesting strategy based on a well-performing mechanism of *duty cycling* is reported in [3, 4], where nodes periodically switch between *awake* (transceiver turned on) and *sleep* (transceiver turned off) states according to a predefined duty cycle, thus saving energy during the sleep times. However, most of these solutions still suffer from problems related to node synchronization, high overhead, high energy consumption at non-intended receivers. Therefore, there is still room for improvements.

In this paper, a novel asynchronous scheduler is defined that exploits the duty cycle approach. A local scheduler, managed by each node, determines node's wake up times by using information about other nodes. Each node communicates to its neighbors the time when it will start transmitting its data, so that they can set their wake up time accordingly. In this way, every node has the list of the transmission times for all of its neighbors and knows in advance when it is supposed to be awake to receive data or can switch to the sleep mode to save energy. Unnecessary awakenings are prevented yet guaranteeing the reception of messages. The solution is able to face topology changes since the information stored by each node is updated as soon as it is not able to listen a neighbor's transmission or is aware of the presence of a new one. Possible desynchronizations due to clock drift are addressed as well. The effectiveness of the proposed scheme has been validated by means of simulations. A performance comparison between the defined Medium Access Control (MAC) protocol and the current ZigBee [5] standard solution has been carried out. The simulation results show a substantial reduction in power consumption, at the cost of a slight increment in delay. But, since many application scenarios are not delay constrained (delay tolerant networks - DTN), this is not a significant problem.

The paper is organized as follows. Section 2 summarizes the state of the art for energy consumption minimization in a WSN. The defined scheduler is described in Section 3. The simulation model is given in Section 4, while in Section 5 numerical results are discussed. Conclusions are drawn in Section 6.

## 2 Related Works

In the literature, several MAC protocols that exploit the duty cycle mechanism have been proposed. These works fit into three main categories: *preamble-sampling*, *scheduling* and *hybrid* approaches.

Preamble-sampling MAC protocols [6, 7, 8] exploit the technique of Low Power Listening (LPL) [9] for sampling the preamble of the packets. LPL minimizes the duty cycle when there are no packet exchanges, but requires a preamble longer than the wake up interval to guarantee that the receiver can detect the channel activity. BMAC [6] uses unsynchronized duty cycling and uses a preamble longer than a sleep period to wake up receivers This approach introduces latency at each hop and suffers from excessive energy consumption at non-intended receivers. WiseMAC [7] uses non-persistent CSMA and preamble sampling to reduce the idle listening. Nodes have independent schedulers and put the information about their next awake time in the data acknowledgement frame. This value is used to dynamically determine the length of the preamble. C-MAC [8] avoids synchronization overhead while allowing operations at very low duty cycles. Three mechanisms are used to improve latency and energy efficiency: aggressive RTS (for channel assessment), anycast (to wake up forwarding nodes) and convergent packet forwarding (to reduce the anycast overhead).

Scheduling MAC protocols [10, 11, 12, 13] use periodic synchronization messages to schedule duty cycling and packet transmissions. Such message exchanges can cause high overhead and consume significant energy even when no traffic is present. S-MAC [10] was the first duty cycle MAC protocol for WSNs. It requires that all nodes in a neighborhood simultaneously wake up and listen to the channel. Nodes remain awake during the entire awake period even if they are neither sending nor receiving data. The static sleep-listen periods lead to high latency and low throughput. T-MAC [11] tries to reduce the long wake up time of S-MAC by shortening the awake period if the channel is idle. Anyway, the wake up time is much longer than the LPL. DMAC [12] uses a data gathering tree structure to achieve both energy efficiency and low packet delivery latency. It assumes that nodes adjust the duty cycles adaptively according to the traffic load in the network. RMAC [13] exploits cross-layer routing information in order to avoid latency. A setup control frame is used to schedule the upcoming data packet delivery along that route so that an upstream node can send the data packet to intermediate relay nodes and these can immediately forward it to the downstream node.

Hybrid approaches [14, 15] combine some features of preamble sampling with scheduling techniques. SCP-MAC [14] exploits the synchronization of the wake up time of neighboring nodes to reduce the length of the preamble and to minimize the cyclic wake up time of an LPL system. It is effective in energy consumption reduction, especially at very low duty cycles, but it does not avoid the overhearing problem. AS-MAC [15] asynchronously coordinates the wake up times of neighboring nodes to reduce overhearing, contention and delay. It employs a duty cycling that periodically puts the radio in a sleep state to avoid idle listening. Furthermore, it exploits LPL to minimize the periodic wake up time. Since nodes store the wake up schedules of their neighbors, they know when these become active. One of the main disadvantages of the asynchronous wake up interval is the inefficiency in broadcasting since AS-MAC has to transmit every packet once per each neighbor.

The current MAC for WSNs is standardized in IEEE 802.15.4 [16], which refers to both MAC and PHY layers. On top of such standard, the most widely used technology for WSNs is Zigbee, which is able to meet the needs of low power, low cost and low maintenance of this kind of networks. As stated in [17], node's energy consumption is mainly due to the high number of retransmissions caused by the CSMA/CA schema of 802.15.4 and a reduction is possible by using innovative schedulers for a better management of the sleep periods. The ZigBee specification permits the introduction in the ZigBee protocol stack of the Low Power Routing feature, which allows a multi-hop mesh networking without requiring router nodes free from energy constraints.

The previous description highlights that synchronization in large multihop networks can be complex because of clock drifts and low duty cycles. Moreover, even though preamble sampling approaches give the sender guarantee that the receiver stays in the awake state for the reception, they suffer from excessive energy consumption at non-intended receivers. The coordination of the wake up times of neighboring nodes and the opportunity to enable networks to self-configure are relevant topics to be addressed, as is done by the solution hereafter introduced.

## 3 The Proposed MAC Scheduler

The main idea of the defined scheduler is that nodes wake up at scheduled times either to transmit packets or to receive messages from their neighbors. According to this model, when the radio is on a node either transmits data and receives acknowledgment messages (ACKs) or receives data from other nodes and sends ACKs.

For the sake of clearness, before describing the new scheduler, the main parameters and concepts used in the discussion are described below:

- $T_0$ is the time interval (in seconds) between two subsequent transmissions. Its value is the same for every node of the WSN and is preconfigured.
- *WakeTime* is the time interval (in seconds) in which a node can transmit the local buffered data or receive data from its neighbors.
- *SleepTime* is the time interval (in seconds) inside which a node can turn off the radio.
- *Announce Packet* ($Pkt_{ANN}$) is a signaling packet used by each node to advertise its presence and its next transmission time.
- *Wake up Table* ($W_{TBL}$) is managed by each node and contains the awakening times for transmission of all the neighbors. Each table entry stores the following information: (a) the ID of the neighbor, (b) the awakening time offset, $O_S$, computed with respect to the first event in the table, and (c) the number of cycles of length $T_0$ during which no data has been received from that node.

In the following more details about the proposed solution are reported.

### 3.1 Start Up of the Network

In the network initialization phase, nodes exchange information about their transmission time by sending a $Pkt_{ANN}$. On the reception of such a message from a new node, everyone updates its $W_{TBL}$ by inserting a new entry. The reception time is added

to the value specified in the packet in order to estimate the next transmission time of the sender. The offset stored in the entry of the table is obtained as difference between the estimated transmission time and the receive time of the first packet received by the node. The entries in the table are in increasing order of offset.

In order to reduce the channel access contention, every node chooses its own transmission time as a random value in a proper range also taking into account the choice done by its neighbors. If the $W_{TBL}$ is empty, then this value is randomly selected in the range

$$[\ T_C,\ T_C + T_0 - (WakeTime + 2*TurnAroundTime)\ ]\ , \tag{1}$$

where $T_C$ is the current time, *WakeTime* is the transmission time and *TurnAroundTime* is the amount of time the radio needs to change its state. If the $W_{TBL}$ is not empty, then every node tries to set its transmission time to a different value with respect to those of its neighbors in order to avoid collisions due to simultaneous transmissions. The value is chosen so that the interval of time reserved for the transmission (*WakeTime*) does not overlap with the transmission time of any neighbor. The node checks if there are two consecutive entries in the table, *i* and *i+1*, whose offsets' difference is greater than a threshold λ, where

$$\lambda\ = 2*WakeTime + 4*TurnAroundTime\ . \tag{2}$$

If this is the case, then the transmission time is chosen within the range

$$[\ offset[i] + D,\ offset[i+1] - D\ ]\ , \tag{3}$$

where $D = WakeTime + 2*TurnAroundTime$, while *offset[i]* and *offset[i+1]* are the offsets associated to entries *i* and *i+1* respectively. If there is only one entry in the $W_{TBL}$, the transmission time is chosen within the range

$$[offset[0]+D,\ T_0 - D]\ . \tag{4}$$

If no entry exists, then the node selects a random value in the range specified in (1). The duration of the start up phase depends on the number of nodes that are switched on at the same time. However, if $T_0$ is great enough to fit all the nodes, it will converge in a finite time.

### 3.2 Steady State

The most important events in this phase are: the periodic wake up for a transmission, the scheduled wake up for the reception of packets from a neighbor, and the arrival of a new node into the WSN. The proposed scheduler manages only the transceiver device, whose state can be either ON or OFF. These should not be confused with the possible states of the MAC layer, i.e., IDLE (inactive MAC), CCA (channel contention), TX (transmission) or RX (reception). Fig. 1 shows the state diagram of the proposed MAC protocol and highlights the main actions that trigger a state change.

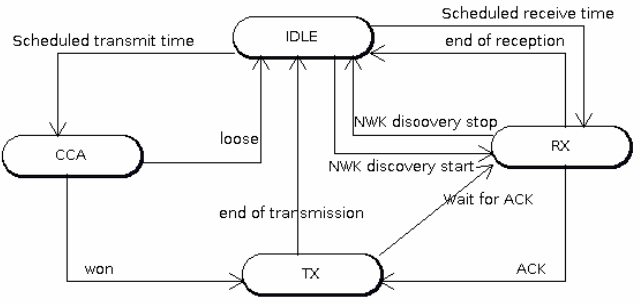



**Fig. 1.** A state diagram of the proposed MAC protocol

The activity of each node is ruled by its $W_{TBL}$. For every scheduled event the node switches its radio on, handles the correspondent event and then switches the transceiver off. There are only two kinds of event to handle: (i) the transmission of a buffered packet (every $T_0$ seconds) and (ii) the reception of packets from a neighbor that has a scheduled transmission.

When the transmission timer of a node expires, the node wakes up and checks the presence of packets to be transmitted in its queues. If there is any, the node starts the contention for the channel access (MAC in CCA state) and, if this is won, it begins to transmit (MAC in TX state). When the transmission ends, the node switches to the RX state and waits for an ACK. If this is not received, then the message must be sent again. On the reception of the ACK, the node goes back to the TX state and schedules the next transmission at $T_0$ seconds after the current time. Then it moves from the TX to the IDLE state and, finally, switches the radio off.

The reception times are determined by checking the $W_{TBL}$: when a neighbor has a scheduled transmission, then the node switches its radio on and starts listening to the channel (MAC in RX state). When the reception is finished, it sends an ACK (switching to the MAC TX state), goes back to the IDLE state and switches the radio component off. If nothing is received, then the node updates the counter in the corresponding $W_{TBL}$ entry about the number of times that no packet has been received from that sender. After a fixed number *m, m>0,* of consecutive failed receptions, the entry is removed so as to avoid useless awakenings. In this way, if a node fails then all its neighbors will remove the corresponding entry from their table after a limited amount of time. Because of this mechanism, it may happen that a node deletes all the entries of its $W_{TBL}$. This situation is the same as when the node has just joined the network, thus the neighbor discovery process must be started again to get an updated view of the neighborhood.

When a new node joins the network, it first listens to the channel for a time interval slightly larger than $T_0$ with the aim of detecting the transmissions of its current neighbors (MAC in RX state). For each packet received from any unknown node, it adds an entry in its $W_{TBL}$. Then, it announces its presence by sending a $Pkt_{ANN,}$, whose payload contains the transmission time selected according to the procedure explained in the start up phase. To transmit such a packet, it starts a contention (MAC in CCA state) with the other neighbors at the end of which the transmission is performed (MAC in TX state).

Since there might not be a perfect synchronization between the timers of the nodes, to compensate the clock drift a receiver brings its reception time forward by a quantity $I_G = 2*C_{DRIFT}$ with respect to the scheduled time, where $C_{DRIFT}$ is the maximum clock drift that may occur. The desynchronization is a negative phenomenon that can generate idle listening and over-emitting. To compensate for this problem a node

- after a fixed number *p*, *p>0*, of consecutive transmissions without ACK, can execute a procedure similar to that one for joining the network also exploiting the $W_{TBL}$ already built or, if this one is empty, doing again the neighbor discovery and announcement procedure, or
- performs only the neighbor discovery procedure if its $W_{TBL}$ is empty but it is still receiving ACKs.

## 4 Simulation Model

To evaluate the effectiveness of the proposed scheduler, an extensive simulation campaign has been carried out by using the tool OMNET++ [18].

The purpose of simulations is to evaluate the benefits that the scheduler can introduce in terms of energy consumption and to assess its impacts on communication delay. A performance comparison between the proposed MAC scheduler and the MAC protocol adopted by the current version of ZigBee has been carried out. Several simulations were run to evaluate the impact of some structural parameters as $T_0$, transmission power and packet rate.

The network layer has been implemented according to the ZigBee specifications [5]. The MAC and PHY layers are compliant with the IEEE 802.15.4 specifications. The proposed schema has been introduced at the MAC layer as a manager for the IEEE 802.15.4 MAC protocol.

To allow the auto-configuration of networks, some new network commands have been defined. The *Announce Packet* ($Pkt_{ANN}$) has been defined as a ZigBee MAC Command Frame (Fig. 2) by exploiting one of the reserved values, while the packet payload includes the next awakening time for transmission. In this way, an open feature of the standard has been exploited without the need of introducing a new packet format.

| Octets: 2 | 1 | 2/8 | 0/5/6/10/14 | 1 | variable | 2 |
|---|---|---|---|---|---|---|
| Frame Control | Sequence Number | Addressing Field | Auxiliary Security Header | Command Frame Identifier | Command Payload | FCS |
| MHR | | | | MAC Payload | | MFR |

**Fig. 2.** ZigBee MAC command Frame format

The topology considered is a multi-hop chain. The mutual distances are 50 m and 20 m respectively for a transmission power of 0 dBm and -10 dBm. This choice allows to compare the power consumption in the same conditions of number of nodes and hops to the sink. It allows to evaluate if it is better, for covering a fixed area, to reduce the transmission power and increase the number of nodes or vice versa. As known, a lower

transmission power implies a lower nodes' transmission range and requires a higher number of nodes to cover a fixed area.

Each node has an on-board sensor that generates data packets with a fixed size of 60 bytes at the application layer. A Constant Packet Rate (CPR) traffic has been chosen. The buffer size at the network layer is assumed infinite. The frequency band used is 2.4 GHz and antennas are omnidirectional. Packets are discarded by the receiver with a packet error rate equal to 0.1%. All the main simulation parameters are reported in Table 1.

All the simulation results are characterized by a 95% confidence interval with a 5% maximum relative error.

**Table 1.** Simulation parameters

| Parameter | Value |
|---|---|
| Network topology | Chain network |
| Number of nodes | 6 |
| Sensors per node | 1 |
| Frequency | 2.4 GHz |
| Transmission Power | 0, -10 dBm |
| Packet Rate | 0.1, 0.5, 1 Hz |
| Packet Error Rate | 0.1 % |
| $T_0$ | 5, 10, 20, 40 s |
| Packet Size | 60 bytes |
| TurnAround Time | 1.92e-4 s |
| Modeled System on Chip | ST/Ember EM250 |

## 5 Simulation Results

Fig. 3 shows the average power consumption versus the distance between sender and receiver (in number of hops) for several values of $T_0$. The transmission power and packet rate are, respectively, 0 dBm and 0.1 Hz. The curves clearly show that, whatever the value of $T_0$, the proposed scheduler reaches a substantial reduction of the power spent by each node, as expected since idle listening and collisions are avoided. For $T_0$ = 5 s the minimum power gain is around 36%, while for $T_0$ = 40 s it reaches more than 90% of the ZigBee solution. It can be noted that the power consumption is a decreasing function of $T_0$, which is a consequence of its impact on the activity of a node. On one hand, a small value of this interval means a high communication activity of the node and thus for its neighbors, which are required to be awake every time a transmission is going to happen and, if necessary, to process the received messages. On the other hand, a large value indicates a small amount of traffic generated by each node and thus a reduced number of awakenings for its neighbors. Another effect can also be noted in Fig. 3. Since nodes closer to the sink forward messages generated by others too, a higher level of power consumption would be expected for them. But this is not the case since farther nodes show a greater consumption. This is due to the different number of neighbors, which determines the number of awakenings in the $W_{TBL}$. Nodes at one hop from the sink have only one neighbor (the sink is not included) while the others in the chain have two and so are awake for more time.

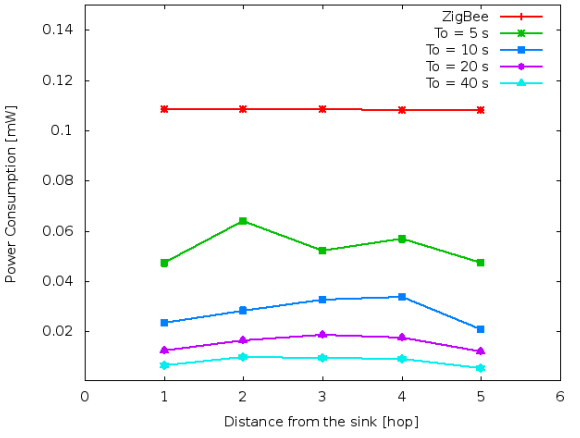



**Fig. 3.** Performance comparison between the proposed MAC and current ZigBee MAC protocols in terms of power consumption versus distance and $T_0$

Figure 4 shows a performance comparison in terms of packet delay. As expected, higher values of $T_0$ result in a higher message delivery time. In particular, this effect is more significant for nodes that are farther from the sink since their messages need to cover more hops to reach the final destination, and at each hop an additional delay is introduced. Here the retransmission is not performed immediately but is deferred of an amount of time related to $T_0$. The delay is accumulated at each hop and this can be negative in the case of delay-sensitive applications, while for delay-tolerant networks the effect is practically negligible and is offset by a high improvement in power consumption.

Further simulations have been carried out to evaluate the influence of the packet rate on the effectiveness of the defined scheduler. When the packet rate is increased,

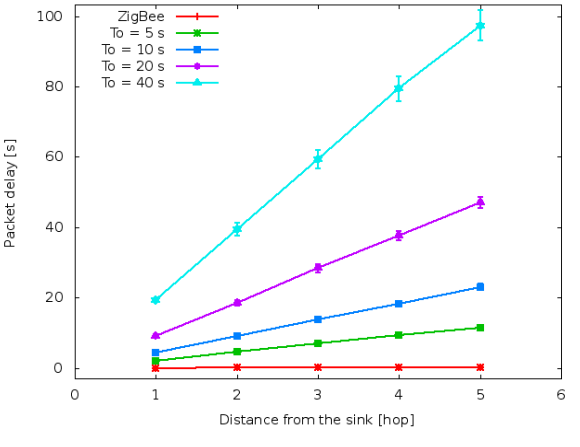



**Fig. 4.** Performance comparison between the proposed MAC and current ZigBee MAC protocols in terms of packet delay versus distance and $T_0$

the activity of each node and of the whole network increases too and greater values of the consumed power are expected. The load for relay nodes increases too. Fig. 5 shows the impact of the defined scheduler on power consumption for a packet rate of 0.1, 0.5 and 1 Hz. The variation in power consumption is quite small, while the packet delay, not shown here, has a behavior similar to the previous case. The same consideration is valid for the other values of $T_0$ and transmission power, which have been evaluated as well for the sake of completeness but are not reported here.

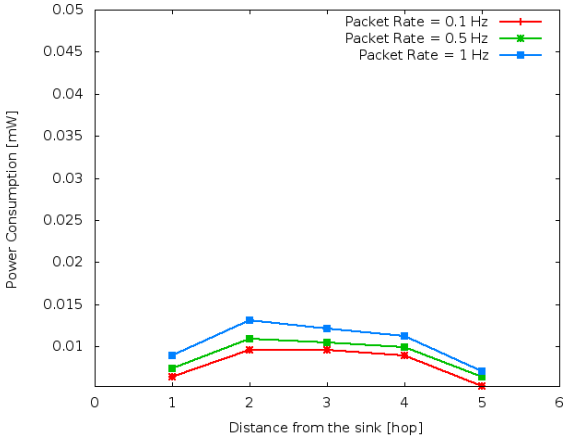



**Fig. 5.** Impact of the defined scheduler on the power consumption by varying the packet rate and setting $T_0$=40 s and TxPower= -10 dBm

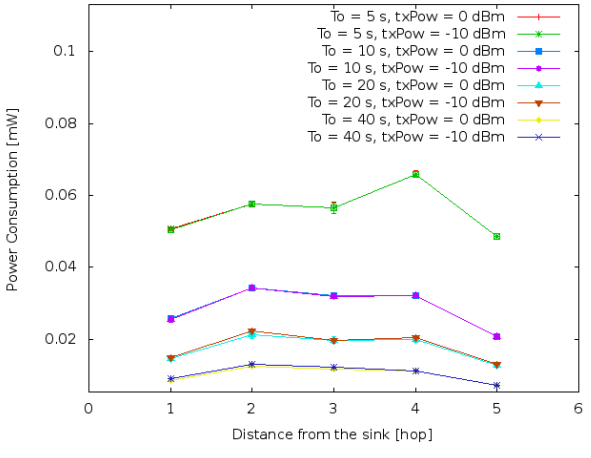



**Fig. 6.** Power consumption for the defined scheduler with a packet rate of 1 Hz , for various combinations of $T_0$ and TxPower

Finally, Fig. 6 shows the power consumption for several combinations of $T_0$ and transmission power. The packet rate is set to 1 Hz. For a fixed value of $T_0$ there is no relevant power consumption reward in reducing the transmission power since the couple of curves are practically superimposed. This is due to the characteristics of the considered antenna [19], whose consumption in reception is greater than that in transmission for low power levels (e.g., -10 dBm or 0 dBm). In other words, the reduction of transmission power has a minimum effect because what impacts more is the length of the interval of time the radio is on to receive something. Therefore, the conclusion is that it is worth using a lower number of nodes transmitting with a higher power, at least in the scenario here analyzed.

## 6 Conclusions

An asynchronous, energy-efficient, scheduled MAC scheme for WSNs has been presented where each node exploits information about the periodic transmissions of its neighbors to schedule its own wake up intervals and avoid useless awakenings. The solution is robust to network changes since the information stored by each node is updated every time a variation is detected. Even though each node needs to store a table about its one-hop neighbors, information content per entry is minimum and the total memory occupation depends mainly on the average density of the network.

Simulations results have highlighted an improvement in power consumption with respect to the current ZigBee MAC protocol, although at the cost of a small increment in the delay. But for most of the practical applications of WSNs delay is not a tight constraint.

The evaluation of the proposed protocol in a more complex and dynamic scenario is a natural evolution of this study. The interaction with routing protocols and the combination with other power saving techniques will characterize future works.

**Acknowledgments.** The authors want to thank Eng. D. Blasi, M. Rizzello, and V. Cacace of the Ultra Low Power Radio & Networks laboratory of STMicroelectronics (Lecce, Italy).

## References

1. Ehsan, S., Hamdaoui, B.: A Survey on Energy-Efficient Routing Techniques with QoS Assurances for Wireless Multimedia Sensor Networks. IEEE Communications Surveys and Tutorials (2011)
2. Tang, Q., Sun, C., Wen, H., Liang, Y.: Cross-Layer Energy Efficiency Analysis and Optimization in WSN. In: International Conference on Networking, Sensing and Control (2010)
3. Cho, K.-T., Bahk, S.: Duty Cycle Optimization for a Multi Hop Transmission Method in Wireless Sensor Networks. IEEE Communications Letters 14(3) (2010)
4. Cohen, R., Kapchits, B.: An Optimal Wake-Up Scheduling Algorithm for Minimizing Energy Consumption While Limiting Maximum Delay in a Mesh Sensor Network. IEEE/ACM Transactions on Networking 17(2) (2009)
5. ZigBee Alliance, ZigBee Specification Document 053474r17

6. Polastre, J., Hill, J., Culler, D.: Versatile low power media access for wireless sensor networks. In: Proc. of the Second ACM Conference on Embedded Networked Sensor Systems, pp. 95–107 (2004)
7. Enz, C.C., El-Hoiydi, A., Decotignie, J.-D., Peiris, V.: WiseNET: An Ultralow-Power Wireless Sensor Network Solution. IEEE Comp. 37(8) (2004)
8. Liu, S., Fan, K.W., Sinha, P.: CMAC: An Energy Efficient MAC Layer Protocol Using Convergent Packet Forwarding for Wireless Sensor Networks. In: 4th Annual IEEE Comm. Soc. Conference on Sensor, Mesh and Ad Hoc Communications and Networks (2007)
9. Merlin, C.J., Heinzelman, W.B.: Duty Cycle Control for Low-Power-Listening MAC Protocols. IEEE Transactions on Mobile Computing 9(11) (2010)
10. Ye, W., Heidemenn, J., Estrin, D.: An energy efficient MAC protocol for wireless sensor networks. In: 21st Intemational Annual Joint Conference of the IEEE Computer and Communications Societies (2002)
11. Van Dam, T., Lengendoen, K.: An adaptive energy efficient MAC protocol for wireless sensor networks. In: 1st ACM Conference on Embedded Neworked Sensor Systems, pp. 171–180 (2003)
12. Lu, G., Krishnamachari, B., Raghavendra, C.: An Adaptive Energy-efficient and Low-Latency MAC for Data Gathering in Sensor Networks. In: International Workshop on Algorithms for Wireless, Mobile, Ad Hoc and Sensor Networks (2004)
13. Du, S., Saha, A.K., Johnson, D.B.: RMAC: A Routing-Enhanced Duty-Cycle MAC Protocol for Wireless Sensor Networks. In: INFOCOM (2007)
14. Ye, W., Silva, F., Heidemann, J.: Ultra-Low Duty Cycle MAC with Scheduled Channel Polling. In: Proc. of the Fourth ACM SenSys Conference (2006)
15. Jang, B., Lim, J.B., Sichitiu, M.L.: AS-MAC: An asynchronous scheduled MAC protocol for wireless sensor networks. In: 5th IEEE International Conference on Mobile Ad Hoc and Sensor Systems (2008)
16. IEEE 802.15.4-2006. IEEE Standard for Information technology – Specific requirements Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (LR-WPANs)
17. Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E.: Wireless sensor networks: a survey. IEEE Comm. Magazine (2002)
18. OMNET++, `http://www.omnetpp.org/`
19. `http://www.ember.com/pdf/120-2001-000_EM250_RCM.pdf`

# Mobile Agents Model and Performance Analysis of a Wireless Sensor Network Target Tracking Application

Edison Pignaton de Freitas[1,2], Bernhard Bösch[1], Rodrigo S. Allgayer[3], Leonardo Steinfeld[4], Flávio Rech Wagner[2], Luigi Carro[2], Carlos Eduardo Pereira[2,3], and Tony Larsson[1]

[1] School of Information Science, Computer and Electrical Engineering, Halmstad University, Halmstad, Sweden
[2] Institute of Informatics, Federal University of Rio Grande do Sul, Brazil
[3] Electrical Engineering Department, Universidade Federal do Rio Grande do Sul, Brazil
[4] Electrical Engineering Institute, Universidad de la República, Uruguay
{edison.pignaton,tony.larsson}@hh.se, berbos09@student.hh.se, leo@fing.edu.uy, {flavio,carro}@inf.ufrgs.br, {allgayer,cpereira}@ece.ufrgs.br

**Abstract.** Advances on wireless communication and sensor systems enabled the growing usage of Wireless Sensor Networks. This kind of network is being used to support a number of new emerging applications, thus the importance in studying the efficiency of new approaches to program them. This paper proposes a performance study of an application using high-level mobile agent model for Wireless Sensor Networks. The analysis is based on a mobile object tracking system, a classical WSN application. It is assumed that the sensor nodes are static, while the developed software is implemented as mobile agents by using the AFME framework. The presented project follows a Model-Driven Development (MDD) methodology using UML (Unified Modeling Language) models. Metrics related to dynamic features of the implemented solution are extracted from the deployed application, allowing a design space exploration in terms of metrics such as performance, memory and energy consumption.

**Keywords:** Wireless Sensor Networks, Multi-agents, Overhead, Energy Consumption.

## 1 Introduction

Wireless sensor nodes are embedded systems used to implement a large number of applications in different areas including those where wired solutions are not suitable. Their potential usage is increased when they form a network of cooperating nodes, i.e. a Wireless Sensor Network (WSN). A WSN is an ultimate technology for applications resembling environmental and area monitoring to acquire different types of measurements. WSNs are used for monitoring wildlife, water resources, security perimeter or collect data for disaster relief and prevention systems [1].

In spite of its enormous potential uses, the real deployment of WSN-based systems presents big challenges, particularly in relation to energy resource usage. Usually,

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 274–286, 2011.

sensor nodes have their energy supply provided by batteries, which represent limited resources. Even in the best cases in which the sensor nodes are able to harvest energy from the surrounding environment, like piezoelectric sensor nodes, the available energy is still a concern. Moreover, their processing capabilities are also limited, which makes it hard to run very complex applications in a single node. Based on that and on the inherent distributed nature of sensor network applications, distributed programming is highly recommended and used for WSN [2].

Another important aspect of WSN applications is related to their operation environment. Usually WSNs operate in harsh and very dynamic scenarios, in which constant changes imply in disturbances of the previous network topology or sensing conditions, by appearance of obstacles or occurrence of communication interferences, for instance. Moreover, the events of interest that have to be monitored per se might be highly dynamic, which requires flexible capabilities in terms of behaviors and functionalities. This dynamicity demands additional features from the WSN, such as reprogrammability, adaptability and autonomy in order to change the sensor nodes' behavior according to the current needs [3]. However, besides the original complexity came from the distributed nature of WSNs applications, these additional desirable features increase even more the overall system programming complexity.

As a consequence of this growing complexity of WSN applications, the deployment of such applications becomes even more challenging. Thus, the development of new programming models is an essential step towards solutions that address such complexity but at the same time present an acceptable level of energy efficiency. In spite of the existence of models used for distributed systems for ordinary computing platforms that could solve the problems related to the distributed nature of WSN applications, they do not address the needs related to the constrained energy and processing resources presented by WSN nodes. Recognizing such situation, research is being developed towards adaptation of such models to the WSN reality.

A promise model to be used in WSN is based on mobile software agents [4]. By the inherent distributed nature of the computation performed by mobile agents, they fit to the distributed processing needs of WSN applications. An example of such usage is presented in [5], in which firefighters use WSN with mobile agents to monitor the progression of fire. However, a drawback of approaches as [5] is the ad hoc nature of the solution and the low level programming paradigm used to implement it, which makes it hard to evolve or adapt such systems. Thus, there is a need for higher level programming languages and methodologies that allow the adoption of flexible mobile agents' solutions. This alternative is becoming possible with the appearance of virtual machines for embedded systems with scarce resources, such as Squawk [6], allowing the use of programming languages with higher levels of abstraction such as Java. These advances combined with the usage of methods with higher abstraction methods, like Model-Driven Development (MDD) [7], allows the systems designers consider different alternatives for implementation and deployment in early stages of the software lifecycle. Moreover, these methods allow the traceability of changes across the design and implementation, which provides a smoother transition between different software versions, when changes are required. However, again the concern about resource usage has to be considered, as higher level programming alternatives for agents may impose high overhead depending on the

software design and how the implementation is carried on. Thus, a study about these issues is required to indicate how the adoption of this type technology with high level abstraction can be improved in WSN domain.

This paper aims to analyze the performance of a WSN system programmed with mobile software agents using a MDD methodology. To carry out this analysis, the classical mobile object tracking application for WSN was chosen [8]. The goal is to evaluate the overhead imposed by an agent support platform in relation to the application. Metrics are extracted from the implemented application to allow the analysis of its performance and the influence of the supporting platform. The outcome of these metrics provides valuable information for design space exploration, making the system designers aware about their decisions and choices during system design and implementation. The implementation was performed using AFME [9], which provides the support for the agents. It is a framework to develop agent-based systems using Java programming language. The implemented application was deployed on a network of SunSPOT sensor nodes [10].

The rest of the paper is organized as follows: Section 2 describes the methodology used to perform this study. In Section 3, the mobile object application is presented and the agent system is detailed. Section 4 presents the application models. Implementation details of the performed simulations along with acquired results are presented in Section 5. Finally, Section 6 presents the conclusion and future work.

## 2 Study Methodology

The road map for the proposed study is composed by three major steps: 1) Application modeling; 2) Application implementation; and 3) Metrics assessment and evaluation.

In the first step, the mobile object tracking application was abstractly modeled using UML use case diagrams to study the system requirements and to identify its functionalities. Then, based on the information achieved by this first analysis, the application was modeled according to the features provided by AFME framework [9]. The resulting model is composed by class and sequence diagrams.

AFME is a low scale agent framework, which was developed to enable the creation of planned agents for mobile devices and resource constrained devices. It is designed to handle the Constrained Limited Device Configuration (CLDC)/Mobile Information Device Profile (MIDP) subset of the Java Micro Edition (J2ME) specification. AFME is based on Agent Factory, a large framework for the deployment of multi-agent systems. The framework is complaint with FIPA specification enabling its interoperability with other FIPA-compliant environments. AFME uses a rule-based concept similar to expert systems [11] to represent the agents' behaviors and maintain a reduced set of meta-information about itself and its surrounding environment as the agents' belief. Rules' operations over the belief set determine the agents' commitments, which finally provide the actions that the agents should perform.

Based on the developed model, the second step could take place and the application was developed in Java programming language using AFME framework for execution on SunSPOTs.

The third step was achieved by acquiring the dynamic metrics of developed application, by instrumented code to retrieve performance measurements during the system execution on the SunSPOT platform. Finally, the acquired data is reported and discussed.

## 3 Mobile Object Tracking Application

The mobile object tracking application used in this study is based on the experiment described in [8], which uses the paradigm of software agents for location and tracking. However, unlike this referred study where agents were static, in the approach here presented the agents migrate among the sensor nodes enabling location and tracking with the cooperation among the nodes. This approach can reduce the amount of data exchanged among nodes, thus lowering the energy consumption.

The network is composed by distributed nodes, which have ability to perform sensing, processing and communication with their neighbor nodes. It is assumed that the sensor nodes have sensor devices which are able to measure the distance to the target. It is possible to use various types of sensors (sound, light and radiofrequency) for this purpose, which may be active or passive to measure the distance to the target, depending on the characteristics of the target object. This work considers that the target object emits radiofrequency signals, called beacons, which are detected by receivers located at each sensor node in the network. Depending of the signal strength received by the nodes, the software calculates the distance between the sensor and the target object based on an omnidirectional model of electromagnetic waves propagation. This model provides that the signal power decays quadratically with the distance between transmitter and receiver.

There are three types of nodes in the network: sensor node, coordinator node and target object. Each node has a distinct function in the application. The target object is characterized by the emission of beacons on the network. The sensor node is responsible for performing the sensing of signals sent by the target object. These nodes are in a greater number in the network. The coordination node is responsible to manage the entry and exit of agents in the network and stores a database with the trajectory of the target object. Figure 1 illustrates this scenario.

The algorithms for location and tracking of the target object in the network are performed by software agents which can be of two types: Resident Agent (RA) and Collaborative Agent (CA). The RA agents can be found fixed in a sensor node and can be a RA_Coordinator (RAC) when the node is a coordinator node or RA_SensorNode (RAS) when it is in a sensor node. They communicate with CA agents when these agents are on the same node. The CA agents have the ability to move among the nodes of the network, being responsible for performing tracking and calculating the position of target objects. They can be a CA_Master (CAM) or a CA_Slave (CAS). As CAM they have the task of coordinating a cluster formed by the sensor nodes that are closer to the target to calculate the location of the target object. This is done by requesting data from others agents and checking when an agent should migrate to another node. As CAS they have the function of cooperating with the agent CAM by sending data.

The injection of agents in the network is accomplished by the coordinator node. When a sensor node detects a beacon from a target object for the first time, it informs the coordinator node (Figure 1 (a)). The coordinator node waits to receive notification of at least three nodes to inject the CAM in network. The sensor node which receives the CAM will be chosen by the shortest distance between the sensor node and the target object (Figure 1 (b)). When the CAM is initialized in the sensor node, it notifies the resident agent about its presence in the sensor node and that it is able to cooperate. Then the resident agent begins to send the distance information to the collaborative agent. This node makes two clones of the agent generating the CASs, and sends these clones to neighboring nodes (Figure 1 (c)). The CASs are initialized in the sensor nodes that received them and inform the respective resident agent, which initiate to send the measured distances. Then CASs send the distance to the CAM.

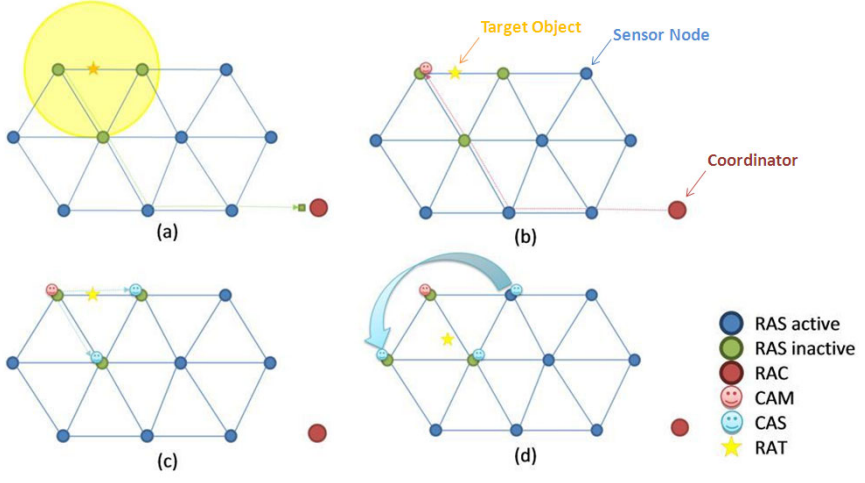



**Fig. 1.** Agents movement in the sensor network for locations and tracking of target object

Aiming to limit the complexity of application, some characteristics were identified and restrictions applied for the implementation:

- Only one target object is presented within the network at a time;
- The sensor nodes and the target object are on a flat surface (2D). Thus, it is possible to determine the position of the target based on the distance measured by at least three sensor nodes. For the case of a 3D surface, not addressed in this work, it would be required the distance values from at least four sensor nodes;
- The relation of the distances between nodes and the communication range assures direct communication between at least three nodes, i.e. an error free communication is assumed;
- The routing protocol allows sending messages to the coordinator, and its current position is known by the network nodes.

### 3.1 Calculation of the Target Object Position

The calculation of the target object position is accomplished by triangulation among three neighboring sensor nodes in the network. The network has a regular shape where the nodes are arranged in a triangular fashion equidistant from one another. The formula for determining the position of the target object ($x_0$, $y_0$) is represented by (1) where it is necessary to know the position of three sensor nodes ($x_i$, $y_i$) and the distances of these nodes to the target object ($r_i$) for i = 1, 2, 3.

The CAM agent is responsible for execute the triangulation algorithm. It has the knowledge of the position of two neighboring nodes and obtains the distance measured by the CAS agents in relation to the target object.

$$p_0 = \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} = A^{-1} \cdot b \tag{1}$$

where:

$$A = 2 \cdot \begin{bmatrix} x_3 - x_1 & y_3 - y_1 \\ x_3 - x_2 & y_3 - y_2 \end{bmatrix} \tag{2}$$

and

$$b = \begin{bmatrix} (r_1^2 - r_3^2) \\ (r_2^2 - r_3^2) \end{bmatrix} - \begin{bmatrix} (x_1^2 - x_3^2) + (y_1^2 - y_3^2) \\ (x_2^2 - x_3^2) + (y_2^2 - y_3^2) \end{bmatrix} \tag{3}$$

The position calculation can degrade the performance of sensor nodes which have a limited processing power and energy sources like traditional sensor network nodes. Thus, optimizations can be performed in the algorithm for calculating the relative position of the target object given by (1)-(3). The matrix A, given by (2), will remain constant until one of the collaborating agents have to perform a migration. Then, the result of the inverse operation of matrix A can be stored until a migration occurs, which will reduce the number of operations performed by the algorithm. The same can be applied to a part of the vector b, represented by (3).

### 3.2 Calculation of Agent Movement (Migration)

The collaborating agents have the characteristic of mobility between the sensor nodes to follow up the target object, minimizing the amount of messages exchanged between network nodes which is one of the goals of distributed processing and collaborative information processing performed by the multi-agents. This feature requires that agents have knowledge about the network topology and to which node they have to migrate in order to be closer to the target object. In Figure 1 (d) depicts the migration of agents between network nodes.

The CAM determines if a CAS agent, or itself, needs to migrate by comparing the distance between the current node that hosts the agent and the target object to a predetermined threshold value. Then, due to the equilateral triangular nodes distribution in the network, the CAM agent can calculates the position of the node

which the agent has to migrate, by using (4). In this case the agent located at ($x_2$, $y_2$) has to migrate to the node located at ($x_4$, $y_4$).

$$\begin{aligned} x_4 &= x_1 + x_3 - x_2 \\ y_4 &= y_1 + y_3 - y_2 \end{aligned} \tag{4}$$

## 4 Application Model and Implementation

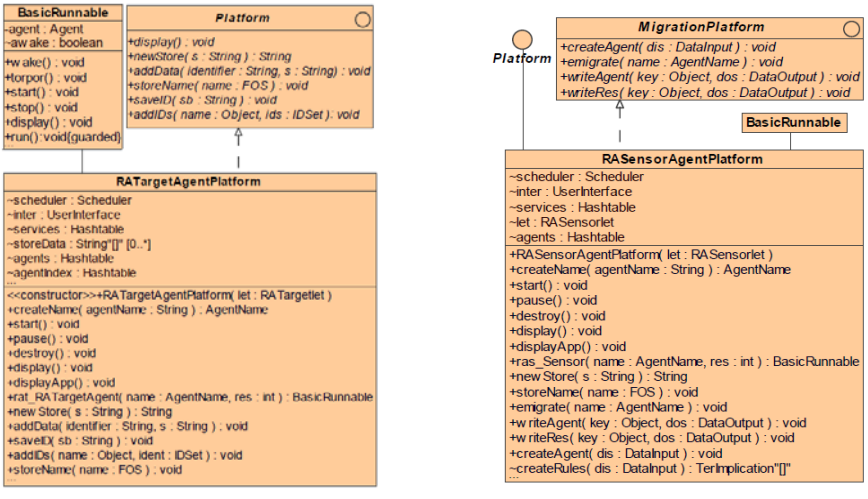
The software was developed based on models created for the application. This software development approach based on the creation and specification of models for the describing application is called Model Driven Development (MDD) [7]. This approach uses specifications (called models) developed in high-level domain languages for software specification, in which UML [12] is a widely used standard and the one chosen for this work.

Some issues needed special attention. First, the application considered in this work, as any collaborative one, involves different parts that run independently in each node of the network, so each of them need to be modeled. Another important issue is the interaction between the network nodes through communication messages may be modeled in two different ways: modeling the communication through a "network" object, where all network messages are sent to and by this object; or modeling the communication channel as an association between the objects that need to interchange messages, so they directly call the appropriate method of each other. This second option translates better the distributed nature of WSN applications, such as the tracking application studied in this work, besides it is simpler to be implemented than the first one, due to the fact of avoiding an intermediary element in the system. For these reasons the second alternative was the selected one.

The Class Diagram is one of the main structural diagram in the UML, in which the classes, interfaces and their relationships are represented. The developed model was divided in three main class diagrams, one for each type of node, namely sensor node, coordinator node and target object. Additionally other class diagrams were created for the CAs. The model includes the classes from the application itself plus the essential ones from the AFME framework. Notice that the framework has many other classes, which are not presented in this paper because they were not used in this project or they are not essential for the overall understanding of the software structure. Parts of the developed class diagrams are presented in the following, which were selected in order to show the relationship between the most important classes from the framework and those from the application. Figure 2 shows the class diagrams for the classes that provided the support to run the agents in the three different kind of nodes, i.e. target object (Figure 2a), sensor node and coordinator node (Figure 2b).

The class diagram of Figure 2a show the class `RATargetAgentPlatform` which implements the interface `Platform` from the AFME API, which provides the basic functionalities to the agents that are hosted in a node. This class contains an instance of the `BasicRunnable` class, which provides the basic functionalities to execute an agent, updating its beliefs and proceeding the agent's control process. The instance of this class that represents an agent is the `RATarget` in the case of target

node. Figure 2b presents a similar diagram for the sensor and coordinator nodes. In the case of these two nodes, besides the implementation of the `Platform` interface, the interface `MigrationPlatform` is also implemented by the class `RASensorAgentPlatform`, which will instantiate an object `RASensorNode` for the sensor nodes and an object `RACoordinator` for the coordinator node from the class `BasicRunnable`. The interface `MigrationPlatform` provides the functionalities that are need for both types of nodes, sensor and coordinator, to send and receive the mobile agents in the application, i.e. the CA_Master and CA_Slave.



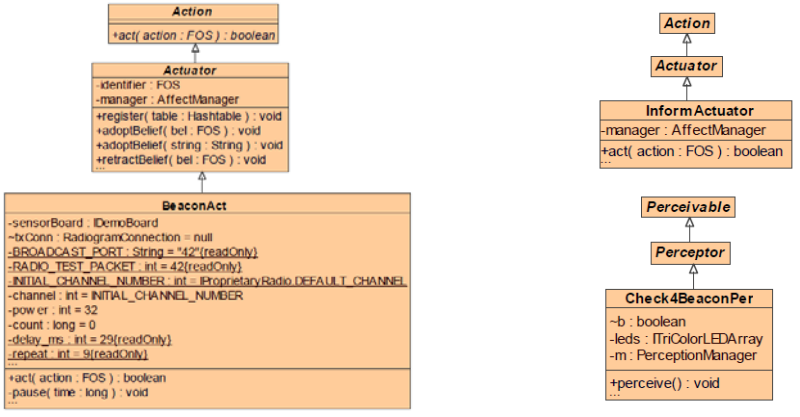

(a) (b)

**Fig. 2.** Class diagram for the basic classes supporting the agents in each node: (a) Target Object; (b) Sensor and Coordinator Nodes

The classes shown so far do only provide the support to run the agents in the nodes. In AFME, the real semantics of the agents are expressed classes representing the agents' perceptions, the Perceptors, and the classes that implement their actions, the Actuators. Figure 3a presents the classes that model the actuator for the RA_Target, while Figure 3b presents the actuator and perceptor of the RA_Sensor.

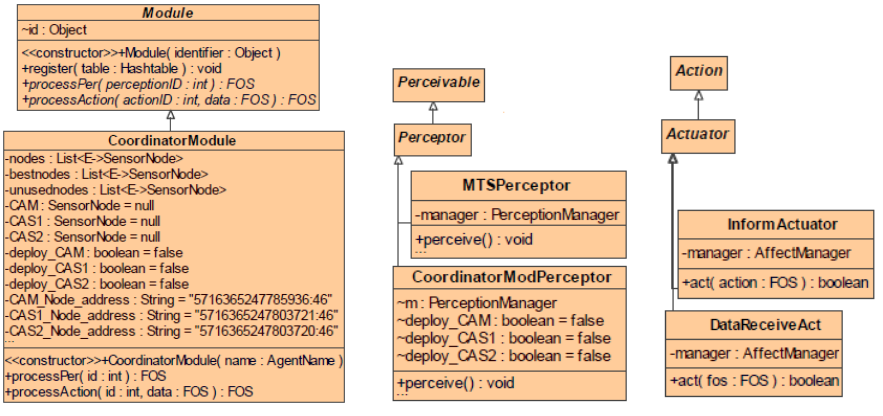
Notice that the RA_Target has no perceptor and just an actuator, the `BeaconAct` in Figure 3a, which is responsible for the action related to sending beacons which will be perceived by the RASensorNode, by means of the functionality implemented in the class `Check4BeaconPer` presented in Figure 3b. This perception will be stored in the agent's belief and informed to other agents resident in the node by means of the functionality in the implemented in the `InformActuator` class (Figure 3b).

Figure 4 present the class diagram with the perceptors and actuator for the RA_Coordinator.




(a) (b)

**Fig. 3.** Class diagram: (a) RA_Target Actuator; (b) RA_Sensor Perceptor and Actuator



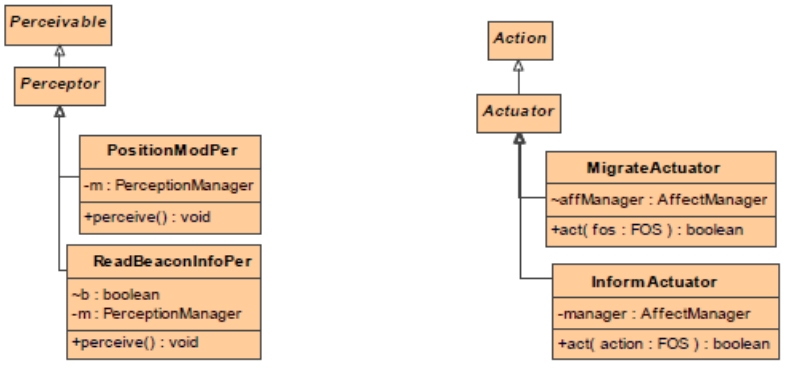

(a) (b) (c)

**Fig. 4.** Class diagram for RA_Coordinator: (a) Module; (b) Perceptors; (c) Actuators

The `CoordinatorModule` presented in Figure 4a extends the class `Module` from the AFME API, and it is mainly responsible for deploying the CA agents in the network. The result of this deployment is perceived by the `CoordinationModPerceptor` (Figure 4b), which keeps track of the CA agents after the deployment, updating the RACoordinator beliefs. This deployment of the CA agents is done after the RASensorNodes have sent the information about the first beacon to the RACoordinator, which handles this information by the actuator `DataReceiveAct` (Figure 4c).

Figure 5 presents perceptors and actuators of the CA_Master agent. The `ReadBeaconInfoPer` class is responsible for the update of the belief state upon a

received beacon from the target node, while the `PositionModPer` updates the target position information (Figure 5a). The `MigrateActuator` is responsible for the agent migration when the target leaves the range of sensing of the current node in which the agent is hosted, while the `InformActuator` is responsible to send messages to the CA_Slave in the neighbor sensor nodes.




(a) (b)

**Fig. 5.** Class diagram for CA_Master: (a) Perceptors; (b) Actuators

The class diagram for the CA_Slave has similar classes as presented for the CA_Master, but with differences in the semantic implemented in its actuators and perceptors, such as sending of information about the target position to the CA_Master and the migration upon receiving a message from the CA_Master.

## 5 Results: Metrics Analysis

The developed software was evaluated by means of extraction of dynamic metrics. These metrics represent information about the application execution, which were achieved during the system runtime by instrumented code. The measurements provide information about the cost in running the application in terms percentage of CPU time utilization, average energy consumption and memory usage.

The testbed was deployed in a network of six SunSPOT nodes: one represent the target object to be detected by other four sensor nodes and the last one was used as a reference node. The reference node was without any agent running on it during the whole time of the experiment, namely Configuration 1. The node representing the target object ran the RA_Target agent, Configuration 2. The other sensor nodes were initially running the RA_SensorNode agent to perform the target detection which was out of the sensors' range, Configuration 3. Then, the nodes were classified during the experiment according to its workload (processing and communication) and depending on the agents they were hosting. When the target entered to the network, three of them performed the detection, Configuration 4. Finally, they performed the target tracking,

two nodes running a CA_Slave agent, Configuration 5, and one a CA_Master, Configuration 6. The RA_Coordinator was running in a PC that represented a base station.

Table 1 presents the results for CPU utilization and current consumption, directly related to the energy drain. Configuration 2 does not consume much processing resources, but energy, which is explainable because the target node periodically sends beacons to the network and wireless communication module demands significant energy to send data. Configuration 3 has a very small CPU utilization if compared with the reference Configuration 1, which is explainable by the fact that it is just waiting for receiving beacons. Its energy consumption is not much high either. However, by the appearance of the target in the network, the sensor nodes in the target range will be in Configuration 4, which presents higher energy consumption, which is explainable by the first beacon message that they send to the coordinator node. The CPU utilization is increased, but not much significantly. With the injection of the mobile CA agents and consequently move to Configurations 5 and 6, it is noticed a significant increase in the CPU utilization, as well as in the energy consumption, which are explained by the execution of the calculations presented in Section 3, as well as the communication among the CA agents.

**Table 1.** Results for the CPU Utilization and Current Consumption

| Sensor Node Configuration | CPU Utilization (%) | Current Consumption (mA) |
|---|---|---|
| **1 - No agent - Reference node** | 1,52 | 59,7 |
| **2 - Target Sunspot** | 17,26 | 74,4 |
| **3 - No CA / No Target present** | 3,44 | 65,8 |
| **4 - No CA/ Target detection** | 15,73 | 71,7 |
| **5 - CA_Slave** | 28,32 | 73,9 |
| **6 - CA_Master** | 60,85 | 85,1 |

Table 2 presents the amount of memory used in each node according to its configuration, and the remaining available memory space. In terms of memory, the burden of the agent oriented approach is not as strong as it is for the energy and CPU usage. As expected, the sensor node running the CA_Master requires more memory, but the difference in relation to the other configurations is not too significant.

**Table 2.** Results for the memory usage

| Sensor Node Configuration / Memory | Conf 2 Target | Confs 3, 4 Sensor | Conf 5 Slave | Conf 6 Master |
|---|---|---|---|---|
| **Free (Bytes)** | 337164 | 334852 | 320300 | 301164 |
| **Used (Bytes)** | 106359 | 108668 | 123221 | 142358 |

By analyzing the achieved results, it is possible to evaluate the imposed overhead due to the use of the adopted agent-oriented approach to implement the WSN application. This information can be used by the system developer to consider alternatives to reduce this overhead, for example, observing the high cost in the node running the CA_Master, the designer can consider redistribute some of the computation to the other nodes that run the CA_Slave. Another possible consideration is in relation to messages exchanged among the nodes, which will impact the design of the application.

## 6 Conclusion and Future Work

This paper presented the application of mobile agents to implement WSN applications. The classical target tracking application was chosen as case study, which was implemented, deployed in real sensor nodes, the SunSPOTs, and evaluated. The achieved results indicated a significant consumption of processing resources and moderate energy consumption. The memory utilization was not a specific concern, for this particular application in the SunSPOT platform, as a significant amount of memory was not used. These results can be used as basis for new designs in which high level decisions can have their impact considered in the system performance.

There are a number of tasks in the pipeline to proceed with this work. One of them is to perform the implementation of the same design using different agent frameworks, so that the specific burden due to the supporting framework can be evaluated and compared. Moreover, other metrics can be acquired, such as static ones to compare different implementations. Another future work is to breakdown the energy consumption for different workload cases, to better understand the impact of an agent-base framework in the communication channel utilization, usually pointed as the major energy consumer.

## References

1. Arampatzis, T., Lygeros, J., Manesis, S.: A Survey of Applications of Wireless Sensors and Wireless Sensor Networks. In: Proceedings of the 13th Mediterranean Conference on Control and Automation, Limassol, Cyprus, pp. 719–724 (2005)
2. Zhao, F., Guibas, L.: Wireless Sensor Networks: An Information Processing Approach. Elsevier, Amsterdam (2004)
3. Allgayer, R.S., Götz, M., Pereira, C.E.: FemtoNode: reconfigurable and customizable architecture for wireless sensor networks. In: Rettberg, A., Zanella, M.C., Amann, M., Keckeisen, M., Rammig, F.J. (eds.) IESS 2009. IFIP Advances in Information and Communication Technology, vol. 310, pp. 302–309. Springer, Heidelberg (2009)
4. Lange, D.B., Oshima, M.: Seven Good Reasons for Mobile Agents. Communications of the ACM 42(3), 88–89 (1999)
5. Fok, C.-L., Roman, G.-C., Lu, C.: Rapid Development and Flexible Deployment of Adaptive Wireless Sensor Network Applications. In: Proc. of the 24th ICDCS, pp. 653–662 (2006)

6. Simon, D., Cifuentes, C.: The squawk virtual machine: Java on the bare metal. In: Proceedings of Conference on Object-oriented Programming, Systems, Languages, and Applications, pp. 150–151. ACM Press, New York (2005)
7. France, R., Rumpe, B.: Model-driven Development of Complex Software: A Research Roadmap. In: Proceedings of Future of Software Engineering 2007, pp. 37–54. IEEE Computer Society, Washington, DC (2007)
8. Tseng, Y.C., Kuo, S.P., Lee, H.W., Huang, C.F.: Location tracking in a wireless sensor network by mobile agents and its data fusion strategies. In: Information Processing in Sensor Networks Book, p. 554. Springer, Berlin (2003)
9. Muldoon, C., O'Hare, G.M.P., Collier, R., O'Grady, M.J.: Agent Factory Micro Edition: A Framework for Ambient Apps. Comp. Science, pp. 727–734. Springer, Berlin (2006)
10. SunSPOTWorld, `http://www.sunspotworld.com` (accessed April 2011)
11. Giarratano, J.C., Riley, G.: Expert Systems: Principles and Programming. Brooks/Cole Publishing Co., Pacific Grove (1989)
12. Object Management Group, Unified Modeling Language, UML (2011), `http://www.uml.org` (acessed March 11, 2011)

# Ubiquitous Sensor Networks Traffic Models for Telemetry Applications

Andrey Koucheryavy[1] and Andrey Prokopiev[2]

[1] JSC "Giprosvyaz", Krasnogo Kursanta 25, St. Petersburg, Russia
akouch@mail.ru
[2]JSC "UbiTel", Ruzovskaya 19, St. Petersburg, Russia
andrey.prokopiev@gmail.com

**Abstract.** Changes in network development concepts and paradigms are the key process at the current telecommunication arena. A shift from NGN concept to IoT, USN, M2M and other proposals is taking place. The major reason of a shift is a wide adoption of wireless sensor nodes and RFIDs. According to forecasts, more than 7 trillion wireless devices are expected to become networked by 2020. The traffic models for networks with great number of sensors and RFIDs should be studied well. This paper studies USN traffic models. The study results show that the traffic flows for fixed and mixed fixed/mobile sensor nodes are of the self-similar nature with middle level of self-similarity in both cases. The traffic flow for reconfiguration and signaling is of self-similar nature with high level of self-similarity. The Hurst parameter mean value estimations are determined for different scenarios.

**Keywords:** USN traffic, telemetry applications, fixed/mobile sensor nodes, Hurst parameter.

## 1 Introduction

The Next Generation Networks (NGN) concept was introduced at the beginning of this century [1]. The main goal of the NGN concept was a formation of a single network base for different telecommunication services provision such as voice, data, video and others. The NGN includes fixed network segment based on Softswitch and IP Multimedia System (IMS) technology, mobile networks UMTS and LTE standards, wireless broadband access network built upon IEEE 802.11 and IEEE 802.16 standards. The single network NGN concept for the mentioned technologies gave the possibilities to support the global networks interoperability [2], which includes the technical means, services, QoS classes and parameters. In any case the NGN concept was born around a decade ago and today a new conceptual platform is needed.

Wide adoption of the wireless sensor nodes and radio identification devices suggests for a new network concept design. It is expected [3] that around 7 trillion wireless telecommunication unites for 7 billion peoples will be connected to a network by 2020. This future network will be self-organizing and the most important traffic sources will be machines.

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 287–294, 2011.

The Internet of Things (IoT) [4] is one of the concept for a future network. There is no clear IoT definition today, however it is clear that at least IoT will include the Ubiquitous Sensor Network (USN) and Web of Things (WoT). Hence, USN traffic models should be studied well. To the authors' best knowledge the research activities in this area currently are pretty limited.

The general topic of traffic modeling in USN is studied in several key publications. Poisson arrival process was assumed for traffic model to each individual sensor node in [5]. The ON/OFF method [6] for USN traffic models is analyzed in [7]. Authors proved that ON period distribution as OFF period distribution could be described by generalized Pareto distribution. The autocorrelation functions for electrocardiogram and body temperature monitoring traffic were studied in [8]. Both of them are non-Poisson. The pseudo long range dependent (LRD) traffic model was proposed in [9] for mobile sensor networks. The Hurst parameter is greater than 0.5, mobility variability higher and spatial correlation smaller.

## 2 The USN Applications Classification

There are a lot of USN applications – everywhere, anywhere, anytime. The most important USN applications include building automation, industrial automation, logistics, transportation, body and intra body sensor nodes and RFIDs, military, agriculture, environment data [10,11]. Further, it can be tree growth, animal development and so on [12]. The USN application map is shown in the fig.1.

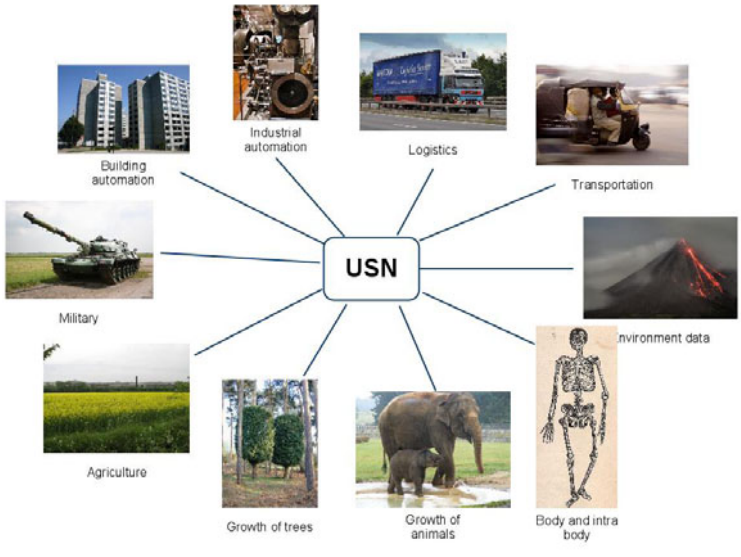



**Fig. 1.** USN application areas

The traffic classification to event-driven and periodic data generation was proposed in [13] where wireless sensor networks for intrusion detection were studied. We propose to classify USN applications in according to traffic model characteristics.

- Voice,
- Signaling,
- Telemetry,
- Pictures (photo),
- Reconfiguration,
- Local positioning.

The USN traffic models for Telemetry applications are considered in this paper both for fixed wireless sensor nodes and mixed fixed/mobile wireless sensor nodes.

## 3 The Traffic Models for Fixed Nodes

The network model includes 50 wireless sensor nodes, which randomly located on the plane of size 30m*30m. The sink is located on the plane center. This model is shown in the fig.2. The described scenario is typical for a warehouse. Each sensor node sends messages every 15, 30, 45 or 60s in according with a given initial random distribution.

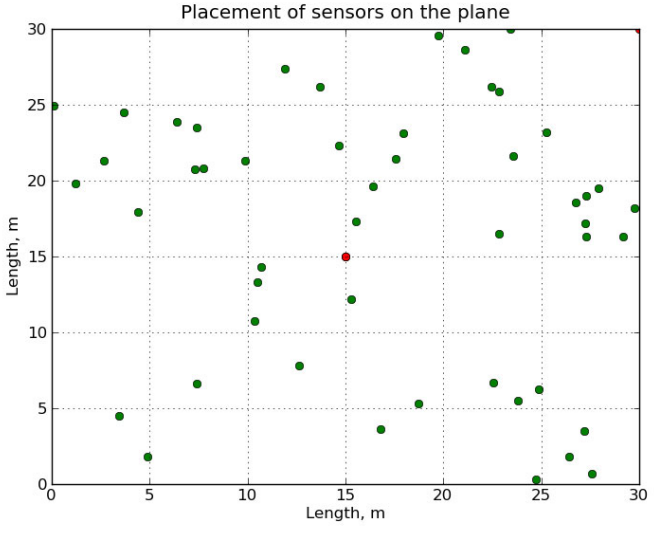



**Fig. 2.** Placement of sensors on the plane

We model the system by ns-2 [14]. Subsequent data processing was made by Python language libraries Numpy and Scipy. Data visualization was done using the Matpplotlib library. The goal of USN traffic modeling was to detect traffic self-similary and correspondent value of Hurst parameter.

The Hurst parameter can be determined as following:

$$D(X^{(m)}) = m^{2(H-1)} D(X) \tag{1}$$

The next equation is used frequently for Hurst calculation:

$$\ln\left(\frac{D(X^{(m)})}{D(X)}\right) = (2H-2)\ln(m) \tag{2}$$

The expression (2H-2) has a geometrical sense. This is the line slope coefficient. This line approximated by function

$$\ln\left(\frac{D(X^{(m)})}{D(X)}\right) = f\left(\ln(m)\right) \tag{3}$$

where $D(X^{(m)})$ is an aggregated flow variance, D(X) is an original flow variance.

The autocorrelation function for realization of a random process of the measured traffic is shown in the fig.3. The appropriate line slope coefficient traffic compared to Poisson process is shown in the fig.4.

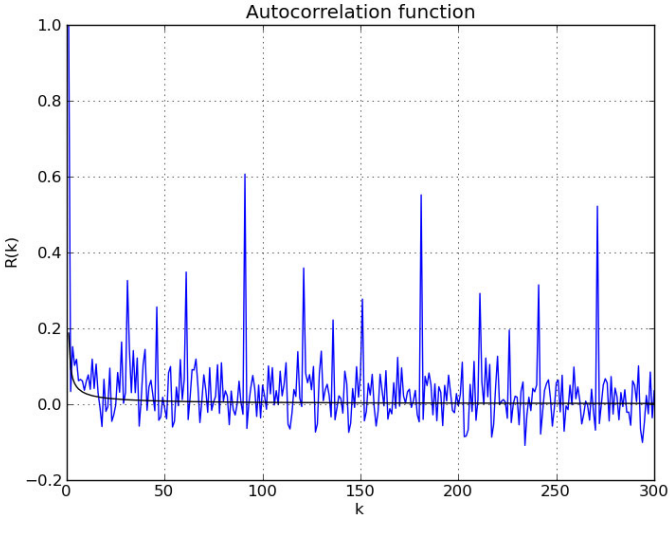



**Fig. 3.** Autocorrelation function

The Hurst parameter mean value is H = 0.675. This value was calculated over 1,000 realizations of a random process. Hence, USN traffic for telemetry applications based on fixed sensor nodes is of a self-similar nature with a middle level of self-similarity.

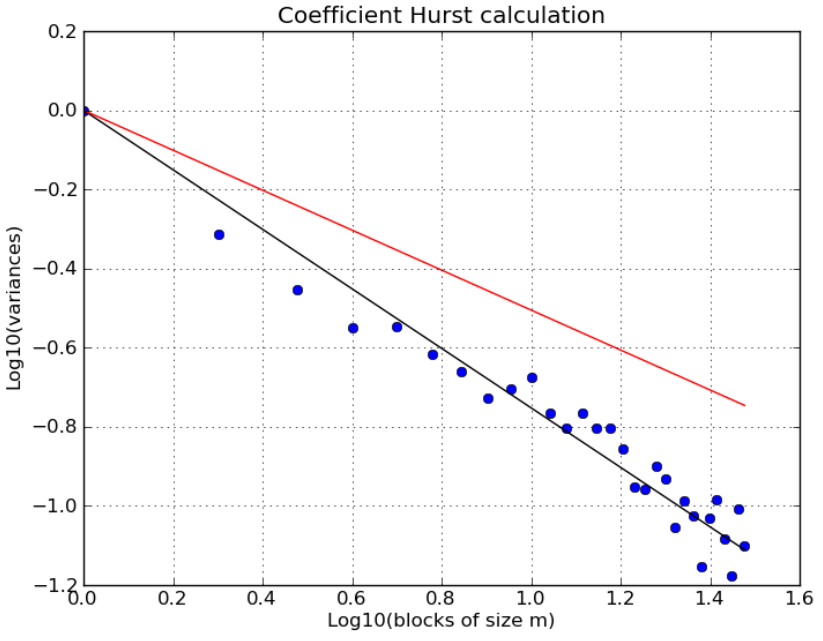



**Fig. 4.** Hurst coefficient calculation based on the line slope

## 4 The Traffic Models for the Mixed Fixed/Mobile Nodes

The network model is the similar as for fixed nodes, but for half of nodes a request for movement comes every 50s. The movement speed is 2m/s [15].

The autocorrelation function for realization of random process measured traffic is shown in the fig.5. The appropriate line slope coefficient traffic compared to Poisson process is shown in the fig.6. The Hurst parameter mean value is H = 0.687. This value was calculated over 3,000 realizations of a random process. Hence, USN traffic for telemetry applications based on mixed fixed/mobile sensor nodes is of a self-similar nature with a middle level of self-similarity.

The measured values of a Hurst parameter are shown in the fig.7. There are two clusters in the fig.7. One represents reconfiguration and signaling traffic (up to 2,200 events), another cluster includes all types of traffic – reconfiguration, signaling and information (telemetry data).

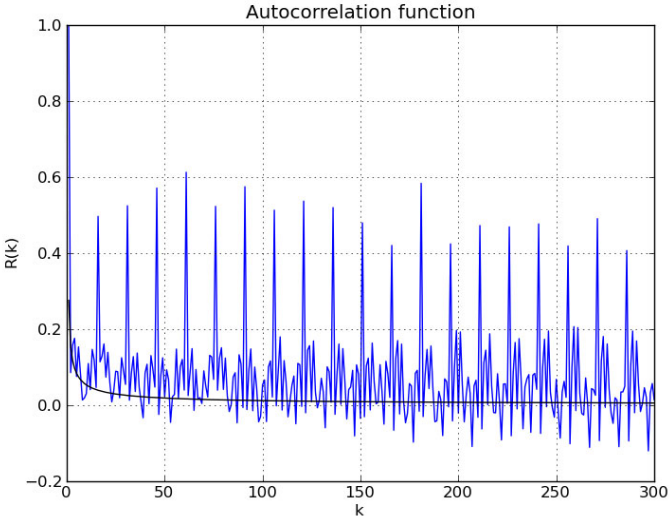



**Fig. 5.** Autocorrelation function for mixed nodes traffic

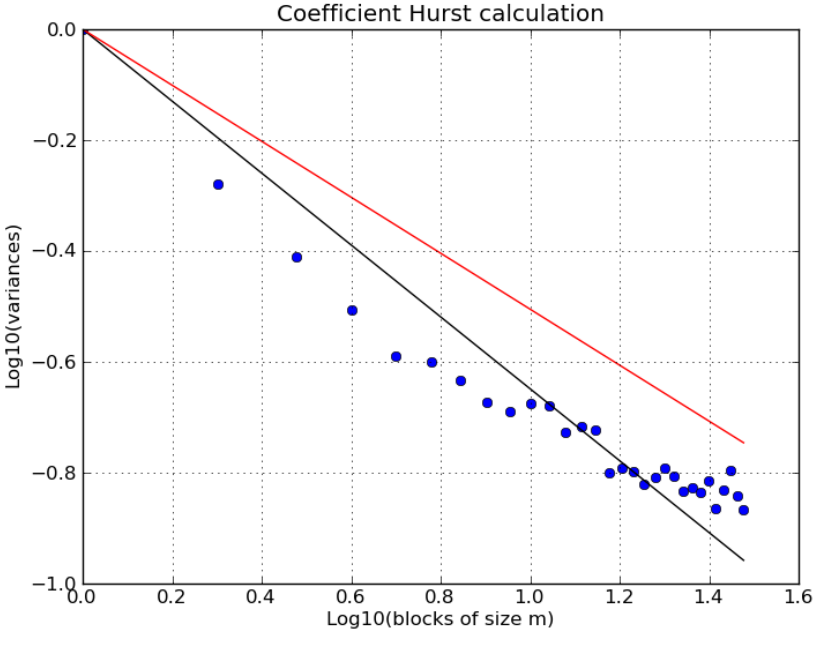



**Fig. 6.** Hurst coefficient calculation based on the line slope for mixed nodes traffic

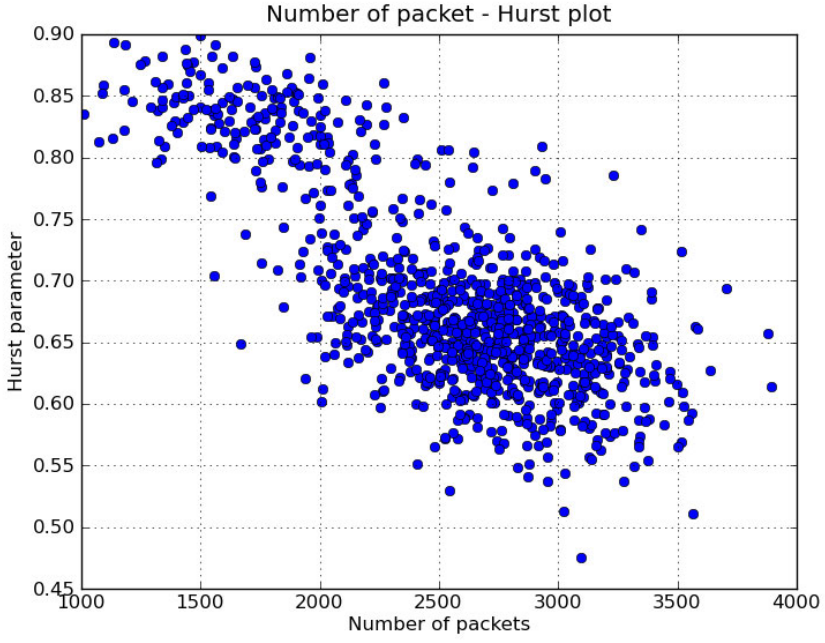



**Fig. 7.** The Hurst parameter measured values for mixed nodes traffic.

The Hurst parameter mean value for reconfiguration and signaling traffic is H=0.829. Hence, USN traffic for reconfiguration and signaling in the telemetry applications based on mixed fixed/mobile sensor nodes is of a self-similar nature with high level a self-similarity.

## 5 Conclusions

The USN traffic models for telemetry applications study using ns-2 identified that the traffic flows for fixed and mixed fixed/mobile sensor nodes are self-similar with middle level of self-similarity in both cases. The traffic flow for reconfiguration and signaling is the self-similar with high level of self-similarity. The Hurst parameter mean values are determined for all models.

## References

1. Recommendation Y.2001. General Overview of NGN. ITU-T, Geneva (December 2004)
2. Koucheryavy, A.: Networks Interoperability. In: Proceedings 11th International Conference on Advanced Telecommunication Technologies, ICACT 2009, Phoenix Park, Korea, February 15-18 (2009)

3. Cheung, N.: Technologies for the Wireless Future: Wireless World Research Forum (WWRF) (August 2009)
4. Iera, A., Floerkemeier, C., Mitsugi, J., Morabito, G.: The Internet of Things. IEEE Wireless Communications 17(6) (December 2010)
5. Tang, S.: An Analytical Traffic Flow Model for Cluster-Based Wireless Sensor Networks. In: 1st International Simposium on Wireless Pervasive Computing (2006)
6. Willinger, W., Taqqu, M., Sherman, R., Wilson, D.: Self-similarity through High-variability. IEEE/ACM Transaction on Networking 15(1) (1997)
7. Wang, Q., Zhang, T.: Source Traffic Modelling in Wireless Sensor Networks for Target Tracking. In: Proceedings 5th ACM International Simposium on Performance Evaluation of Wireless Ad Hoc, Sensor and Ubiquitous Networks (PEWASUN 2008), Vancouver, Canada, October 27-31 (2008)
8. Messier, G.G., Finvers, I.G.: Traffic Models for Medical Wireless Sensor Networks. IEEE Communications Letters 11(1) (January 2007)
9. Wang, P., Akyildiz, I.F.: Spatial Correlation and Mobility Aware Traffic Modelling for Wireless Sensor Networks. In: Proceedings IEE Global Communications Conference (GLOBECOM 2009), Honolulu, Havaii, USA, 30 November-4 December (2009)
10. Shelby, Z.: Embedded Web Services. IEEE Wireless Communications 17(6) (December 2010)
11. Kim, B.-T.: Broadband Convergence Network (BcN) for Ubiquitous Korea Vision. In: Proceedings 7th International Conference on Advanced Telecommunication Technologies, ICACT 2005, Phoenix Park, Korea, February 21-23 (2005)
12. Marrocco, G.: Pervasive Electromagnetics: Sensing Paradigms by Passive RFID Technology. IEEE Wireless Communications 17(6) (December 2010)
13. Demirkol, I., Alagoz, F., Delic, H., Ersoy, C.: Wireless Sensor Networks for Intrusion Detection: Packet Traffic Modeling. IEEE Communications Letters 10(1) (January 2006)
14. Fall, K., Varadhan, K.: The ns Manual (formerly known as ns Notes and Documentation) (May 2010), http://www.isi.edu/nsnam/ns/doc/ns_doc.pdf
15. Koucheryavy, A., Salim, A.: Prediction-based Clustering Algorithm for Mobile Wireless Sensor Networks. In: Proceedings 7th International Conference on Advanced Telecommunication Technologies, ICACT 2010, Phoenix Park, Korea, February 7-10 (2010)

# Evaluation of RTSJ-Based Distributed Control System

Ivan Müller[1], André Cavalcante[1,2], Edison Pignaton de Freitas[3,4], Rodrigo Schmidt Allgayer[1], Carlos Eduardo Pereira[1,4], and Tony Larsson[3]

[1] Electrical Engineering Department,
Federal University of Rio Grande do Sul, Brazil
[2] Electronics and Telecommunication Department,
Federal University of Amazonas, Brazil
[3] School of Information Science,
Computer and Electrical Engineering, Halmstad University, Halmstad, Sweden
[4] Institute of Informatics, Federal University of Rio Grande do Sul, Brazil
ivan.muller@ufrgs.br, andre.d.cavalcante@gmail.com,
{allgayer,cpereira}@ece.ufrgs.br,
{edison.pignaton,tony.larsson}@hh.se

**Abstract.** In this paper an analysis of a distributed control system based on Java is presented. A classical PID controlled system is implemented simulating each part of a real control system running in different computers connected to a local area network. The communication message time periods and their jitter are measured running the system in different computer environments and the results are presented and discussed at the end. Real time specification for Java is used in the implemented software and the results are compared to other implementations.

**Keywords:** Distributed Control Systems, Real-time Specification for Java, Real-time Evaluation.

## 1 Introduction

Systems with a distributed control scheme are desirable due to several advantages such as the increase of overall robustness, redundancy and task migration [1]. Also, the usage of computational resources per machine is usually reduced promoting a more balanced system. Today there are several promising networks solutions that can be used to perform distributed control systems in different environments, including personal, local and wide area networks. Distributed control systems (DCS) can make use of legacy network infrastructures such as LAN, Wi-Fi and IEEE 802.15.4.

Centralized control systems require very expensive and fixed physical structures. On the other hand spreading out the tasks of a control system leads to a decentralized architecture that strongly depends on the communications layer behavior of the system. Networks usually introduce transportation lags that lead to control system instability [2]. The delay generated by the communication layer of the network must be known and its jitter kept as low as possible in order to guarantee control system stability. Synchronous control messages are necessary and their delivery must be guaranteed, considering time and order. To do this a message priority arbitration

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 295–303, 2011.

scheme is usually implemented. Asynchronous messages such as calibration parameters will occur and their treatment must be decoupled as much as possible from the system synchronicity. Keeping these facts in mind, the communications strategy rules an important part of a distributed control system.

In this paper, a distributed control system is proposed and analyzed evaluating a classical control scheme within a simulated plant. Three communication topologies were studied and implemented using different operational systems and plug-ins. This investigation searches for in impact that the usage of different operational systems and plug-ins may cause in the overall system performance, from a perspective of its timing properties. The analysis of their impact helps the developers to choose the appropriate components to develop a DCS that accomplishes with the expected timing behavior. In Sections 2 and 3 a background concerning distributed control systems and network communications are presented. In Section 4 the evaluation of the proposed test system is described and in Section 5 the conclusions are presented along with the future directions of this work.

## 2 Distributed Control Systems

Classical control systems are usually composed by four main parts: set point variable, controller, plant to be controlled and the sensor as Figure 1 depicts. The set point is the desired input value and the output variable which is obtained by the sensor is subtracted from the desired signal thereby closing the control loop. The basic idea behind distributing a control system is to split these main blocks into different machines as Figure 2 shows.

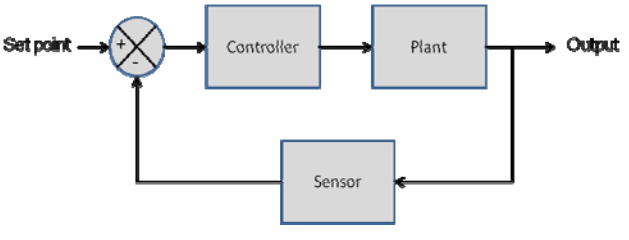



**Fig. 1.** Basic Control System

The use of DCS promotes system scalability [1]. The nodes can be easily changed and can even perform different tasks. In more complex designs, tasks could migrate from one node to another. In a fault scenario, intelligent devices can assume different behaviors to place the system in a secure situation [3]. Node devices can afford with more complex software, which provides intelligence, as they need a microcontroller unit to implement a communication protocol. Nowadays embedded systems permit easy development of intelligent nodes with these features [4].

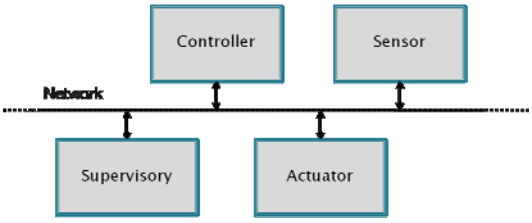



**Fig. 2.** Basic Control System

## 3 Communications on DCS

The design of communication systems used to perform a DCS is an important concern in the system design. Basically, it depends on network topology as well as physical aspects. Considering LAN and WAN, the widely used protocols are TCP and UDP, the internet protocols. The main advantage of these protocols is their large compatibility and usage. On the other hand they are not real time protocols. Some efforts are being done in order to make Ethernet a deterministic network [5]. TCP/IP protocol makes error detection and generates retransmission of lost packages that leads to extra payload generation. As it is a connection oriented protocol, there is package delivery guarantee, but they can occur in different times and even in different orders. Reliability is one of the main advantages of this protocol, but it can introduce large communication jitters. Also, the TCP/IP connection oriented scheme leads to a client-server topology which can be undesirable in DCS. On the other hand, the use of sockets simplifies software design shortening the development time. If another topology is desired, UDP can be taken into account. It permits some degree of freedom to developers that want to make their own protocols while keeping the advantage of simplicity and compatibility with large installed networks. As a disadvantage, it does not have lost package detection therefore the developer must implement this feature. Other approaches include DCS made under wireless sensor networks which represents new challenges to developers [6].

By developing a DCS, two types of messages have to be considered: synchronous, that maintains the control system stable and asynchronous, that will occur sporadically such as calibration, alarm and report procedures. When asynchronous messages occur their management must not overcome the priority of cyclic messages that keep the control system stable. One possibility is the development of a time synchronized protocol such as FTT-CAN [7]. Implementing free time slots to be filled with asynchronous messages will guarantee the synchronous message delivery without any interference. These slots are actually part of a synchronous protocol and the only disadvantage is the extra payload generated. The messages can be passed through the devices of a DCS until they reach the correct destination like a token oriented communication.

Another possibility to implement a DCS is the use of a remote procedure call (RPC). In this case the communication layer is abstracted and procedures are called remotely. There are many types of remote procedure calls such as Java remote method protocol, Java message services and .NET remoting [8,9], and they were

already used in several DCS. Java remote method invocation (RMI) is another type of RPC that have being used [10]. The main advantage of RMI when compared with other RPCs lies in the fact that it is a true polymorphic object oriented technology because it uses object serialization to marshal and unmarshal parameters, not truncating types. Beyond the choice of communications paradigm, messages must be managed in order to keep the control system stable in a DCS. By the use of RPCs, threads and multithreads come naturally as a choice to manage message synchronicity. Threads are easily implemented in Java codes as well as RMI. However RMI uses TCP as a communications protocol and the jitter will be present. One possibility is to implement RMI or other RPC that makes use of another protocol such as a time synchronized one in order to keep the delivery of periodical messages with a minimum jitter.

## 4 Case Study

In order to analyze the impact of communication jitter, a DCS is implemented and its diagram is showed on Figure 3. Communications is done by means of publish-subscribe and sockets through TCP and UDP. Also RMI is implemented in order to compare with previous topologies. The system is tested in different operational systems and using Java real time plug-in (RTSJ) in order to obtain data for comparison. The application is started by a form that contains the main method. Buttons are used to start node devices; controller, sensor and actuator so the desired device can be started in different machines.

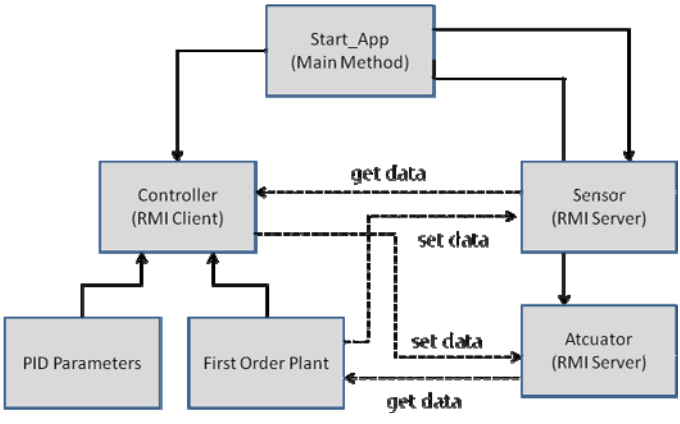



**Fig. 3.** Block diagram of the evaluated DCS

RMI and sockets leads to a client-server behavior and in the case of the studied system, the sensor and the actuator are remote server objects communicating with a client (the controller). The cyclic messages that guarantee system stability are presented by the *get data* arrows from the sensor and by the *set data* arrows to actuators, as depicted in Figure 3. Two additional classes are implemented, plant and

PID parameters in order to emulate a real control system. They are part of a Java controller class giving it control law parameters and interacting with the servers. The first order plant has a thread that generates data to the sensor and grabs data from the actuator. Its transfer function is presented as follows:

$$P(s) = K\left(\frac{s}{s+p}\right) \tag{1}$$

The classical proportional, integrative and derivative is implemented using Tustin's approximation [11] to make the discrete equations to be computed. Two extra RPCs are implemented to get calibration data from sensor and actuator thus generating asynchronous messages. For certain PID parameters the maximum transportation lag before system instability is 1200 ms. Fixing this value, it is possible to have a common term to compare the system performance in different environments, leaving the control system aspects in a second plane. Different control systems can be more adequate to perform a DCS over Ethernet due to its non-linear behavior [12] but, in this paper, the control system theory itself is beyond the scope.

The system was tested in different operational systems and under different conditions. The goal of these tests was to evaluate the timing behavior of the system in relation to its predictability. As unexpected variations are the main source of problems in real-time systems, the jitter of the specified timing requirement is measured and analyzed. Table 1 presents the performed tests and the average jitter result.

**Table 1.** Different tested setups and achieved results for the measured jitter

| O.S | Plug-in | Average Jitter |
|---|---|---|
| Windows | - | 3.5 ms |
| Linux | - | 3.1 ms |
| Linux-RT | RTAI | 3.2 ms |
| Linux-RT | RTAI + RTSJ | 2.3 ms |

At first glance, jitters are similar, but the combination of real-time operation system plus RTSJ diminished the average value as expected. Although the communication layer contributes to the greater part of jitter, using a real-time operation system together with a real-time plug-in proven to diminish problem related to jitter, as it effectively handle real-time requirements, as was deeply discussed in [13]. The need for a real-time behavior in the RMI or other communication strategy to be implemented on a DCS is clear, as previously stated [14]. Figures 4 - 6 graphically presents the results obtained from the different implementations. Only Linux O.S results are presented because of different implementations of Java in this system (there is no RTJS plug-in for Windows up to now). The DCS was tested running the devices in one (local host), two and three different computers interconnected by a switch. The experiments were done varying the sample time that actually is the thread sleep parameter of the control system. The values were fixed in 100, 500 and 1000 ms below the system stability limit.

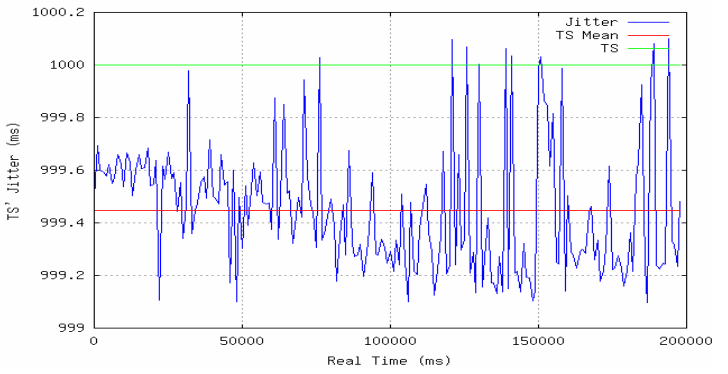



**Fig. 4.** Jitter on a plain Linux OS

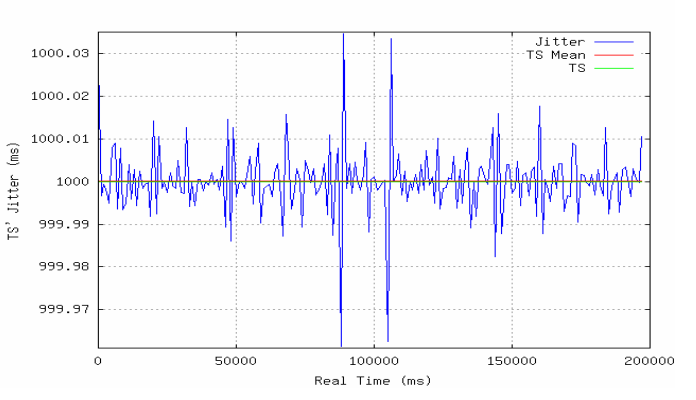



**Fig. 5.** Jitter on Linux real-time OS

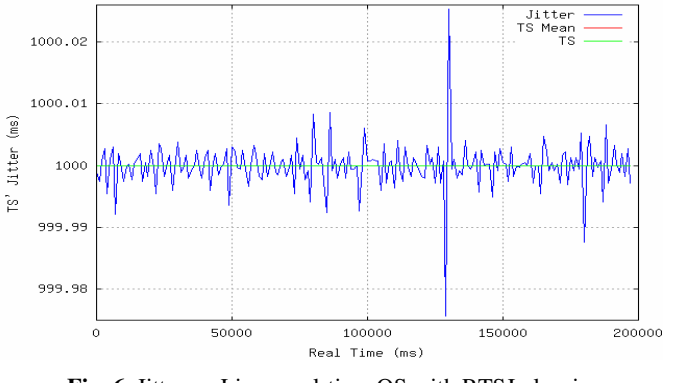



**Fig. 6.** Jitter on Linux real-time OS with RTSJ plug-in

The data presented in Figures 4 - 6 were generated with three computers DCS with a 1000 ms fixed thread sleep time. The use of a real-time operational system provided the correct timing for thread sleep, fixing the transportation lag of DCS (can be noticed comparing TS variable on graphics). Non real-time OS cannot handle time requirements as Figure 4 denotes.

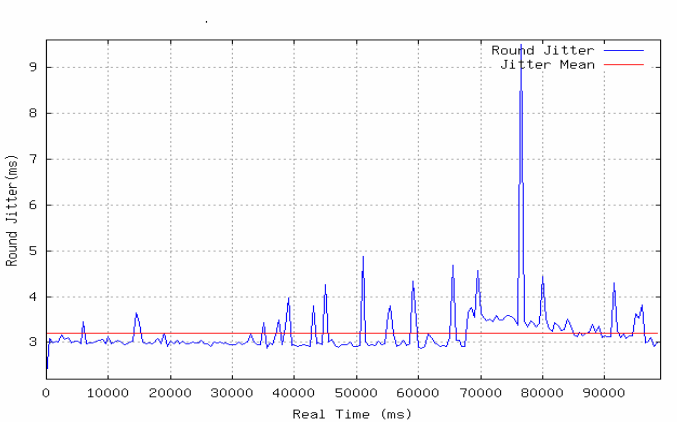



**Fig. 7.** Round jitter on a plain Linux OS

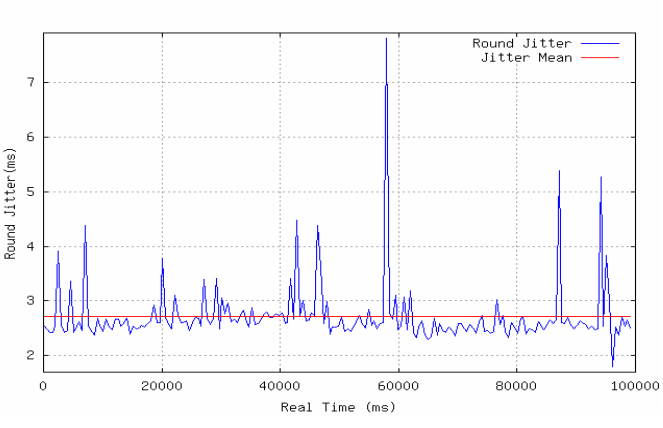



**Fig. 8.** Round jitter on a plain Linux real-time OS

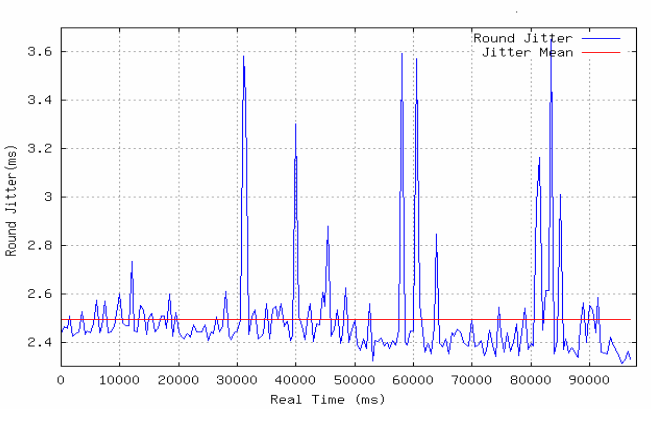



**Fig. 9.** Round jitter on a plain Linux real-time OS with RTSJ plug-in

Figures 7 - 9 depicts the round jitter for a 500 ms fixed sleep time. It can be noticed again that jitter is much better controlled in a real-time OS and RTSJ system. However, the average jitter values are closer to each other by the influence of RMI communications that are common to each experiment.

## 6 Conclusion and Future Work

This paper presented a proposal and an evaluation of distributed control system using Java remote method invocation. The simulated classical control system was tested in different environments under controlled stability situations. The communication jitter was kept under control for the experiment setups using a real-time operational system in conjunction with a real-time Java plug-in. Other communication protocols were implemented and proven to be feasible, but tended to be laborious when compared to Java RMI. The use of TCP by Java RMI simplifies the design of a DCS, but introduces jitter. Also, the results showed the differences among the systems and plug-ins. It was proven that the use of real-time operating systems in conjunction with real-time plug-ins promotes jitter reduction. The experiments clearly confirmed the need for a real-time behavior in the RMI or other communication strategy to be implemented on a DCS. Based on these findings, it is possible to state that despite the variations in the timing properties imposed by the communication channel, the real-time capabilities of the underlying system supporting the application plays an important role in improving predictability.

In a future work, the Java RMI will be overlaid to a time synchronized protocol in order to enhance the results here presented, by diminish communications jitter even more. Additionally, the use of wireless communication among the nodes in the network for future tests is in the pipeline, as this type of communication is gaining importance in the DCS domain [14].

**Acknowledgement.** The authors would like to acknowledge Capes and CNPq, the Brazilian commissions for post graduation and research, FEPEAM, the Swedish Knowledge Foundation and the Brazilian Army for supporting this work.

## References

1. Arampatzis, T., Lygeros, J., Manesis, S.: A Survey of Applications of Wireless Sensors and Wireless Sensor Networks. In: Proceedings of the 13th Mediterranean Conference on Control and Automation, Limassol, Cyprus, pp. 719–724 (2005)
2. Equiraun, M., Jugo, J.: A Java based tool for Distributed Control Systems. In: Proceeding of IEEE International Conference on Emerging Technologies and Factory Automation, ETFA 2008, pp. 1076–1079 (2008)
3. Yang, S., Zhengrong, X., Quingwei, C., Weili, H.: Dynamic output feedback control of discrete switched system with time delay. In: Proceedings of the V World Congress on Intelligent Control, Hangzhou, China, pp. 1088–1091 (2004)
4. Galdun, J., Takac, L., Ligus, J., Thiriet, J.M., Sarnovsky, J.: Distributed Control Systems Reliability: Considerations of Multi-agent Behavior. In: 6th International Symposium on Applied Machine Intelligence and Informatics, SAMI 2008, pp. 157–162 (2008)

5. Meyer, G.G., Framling, K., Holmstrom, J.: Intelligent Products: A survey. Computers in Industry 60(3), 137–148 (2009); Intelligent Products
6. Felser, M.: Real-Time Ethernet - Industry Prospective. Proceedings of the IEEE 93(6), 1118–1129 (2005)
7. Franceschinis, M., Spirito, M.A., Tomasi, R., Ossini, G., Pidalà, M.: Using WSN Technology for Industrial Monitoring: A Real Case. In: Proceedings of the Second International Conference on Sensor Technologies and Applications, pp. 282–287 (2008)
8. Almeida, L., Pedreiras, P., Fonseca, J.A.G.: The FTT-CAN protocol: why and how. IEEE Transactions on Industrial Electronics 49(6), 1189–1201 (2002)
9. Shwarzkopf, R., Mathes, M., Heinzl, S., Freisleben, B.: Dohmann. H.: Java RMI versus.NET Remoting Architectural Comparsion and Performace Evaluation. In: Proceedings of Seventh International Conference on Networking, ICN 2008 (2008)
10. Koutsogiannakis, G., Savva, M.: Performance Studies of Remote Method Invocation in Java. In: Proceedings of 21st IEEE International Performance, Computing, and Communications Conference (2002)
11. Waldo, J.: Remote Procedure Calls and Java Remote Method Invocation. IEEE Concurrency 6(3), 5–7 (1998)
12. Haykin, S., Van Veen, B.: Signals and Systems. Wiley, Chichester (1999)
13. Cheingjong, P., Wongsaisuwan, M.: Adaptive PI control application of a heat exchanger via distributed control system. In: Industrial Technology, ICIT 2008, pp. 1–4 (2008)
14. Pereira, C.E., Ataide, H.A., Kunz, G.O., Freitas, E.P., Silva, E.T., Carvalho, F.C.: Performance evaluation of Java architectures in embedded real-time system. In: Proceedings of 10th IEEE Conference on Emerging Technologies and Factory Automation, ETFA 2005, pp. 841–848 (2005)
15. Borg, A., Wellings, A.: A Real-time RMI Framework for the RTSJ. In: Proceedings of the 15th Euromicro Conference on Real-Time Systems (2003)

# Test Scenarios for EMS Operation in Hybrid PON System

Kyu Ouk Lee, Sang Soo Lee, and Jong Hyun Lee

Optical Access Tech. Research Team,
Electronics and Telecommunications Research Institute (ETRI),
218 Gajeongno, Yuseong-gu, Daejeon, Korea
{kolee,soolee,jlee}@etri.re.kr

**Abstract.** Test scenarios are proposed for EMS operation in hybrid PON which are composed of E-PON, G-PON, and WDM PON system. The EMS operates, manages, tests the hybrid PON system, and consistes of EMS manager, EMS GUI and EMS agent. The proposed test scenarios provides guidelines for implementation of hybrid PON system to operators who already operate the E-PON or G-PON systems.

**Keywords:** EMS, PON, SNMP, OAM.

## 1 Introduction

The Hybrid PON system is composed of G-PON, E-PON, and WDM-PON as shown in Figure 1.

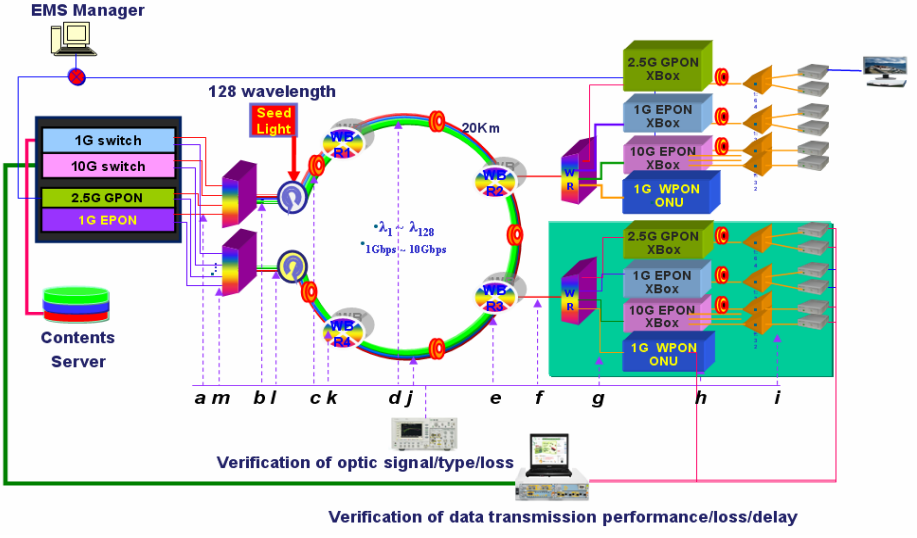



**Fig. 1.** System configuration of hybrid PON system

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 304–313, 2011.

Until now on, many operators already have operated the E-PON or G-PON system, and these operators want to install the WDM-PON system, especially for mobile backhaul implementation. Existing E-PON or G-PON operators have his own EMS for system operation and maintenance, so this paper proposes the hybrid EMS and test scenarios for operation, maintenance and testing the Hybrid PON system.

## 2 Hybrid EMS (Element Management System)

The Hybrid EMS operates, manages, tests the hybrid PON system, and consists of EMS Manager, EMS GUI and EMS Agent as shown in Figure 2.

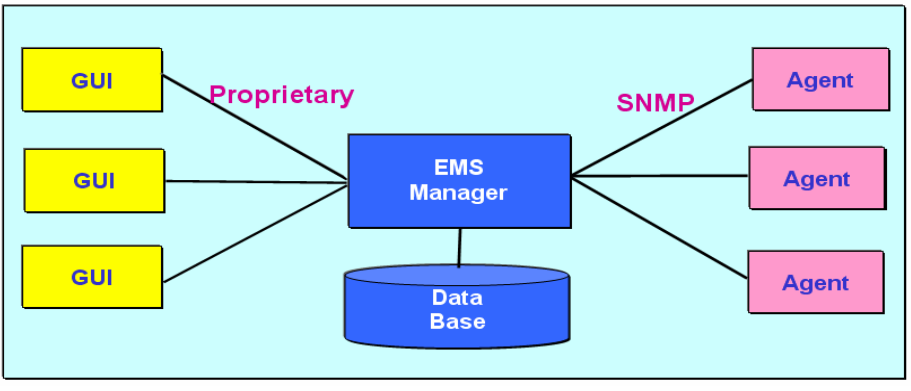



**Fig. 2.** Hybrid EMS structure

### 2.1 EMS Manager

The EMS Manager has a database and stores the collected data from agents. The SNMP protocol is used between EMS Manager and Agent, and the internal protocol based on TCP/IP is used between EMS Manager and GUI for data communications. Several EMS GUIs can be connected to the EMS Manager and EMS Manager has following requirements

- CPU: 32-bit Pentium or alike (recommended: 400MHz or greater)
- Memory: 512MB RAM system memory (recommended: 256MBytes or more)
- HDD: 100MB available disk space (plus size of user's capture files, e.g. 100MB extra)
- Monitor: More than 1280*1024 display
- O/S: windows XP/Linux
- JVM (Java Virtual Machine)

While, the EMS Manager is composed of data collecting function, data offering function, notification function. The data collecting function stores the data which is collected from Agent periodically. The data offering function is interworking function with EMS GUI, and creates or return the data set, in case requesting the specific equipment's information from EMS GUI. Otherwise, user management, certification function, agent status identification functions are executed. The icon of EMS Manager is created by executing the startEmsmanager.bat file, and the EMS Manager window is displayed by double clicking the Tray Icon.

### 2.2 EMS GUI

The EMS GUI shows the graphic view and main screens of EMS GUI are composed of tree view, node view, and detailed view. The tree view manages the each system components and the node view manages the network configuration, and the detailed view displays the collected data, along with the InfoBox of node view displays the summary information of collected data.

## 3 Test Scenarios

The test scenarios for EMS operation are composed of 4 different scenarios; optic transmission link test (scenario 1), interworking test between E-PON/G-PON and WDM-PON (scenario 2), long term environment test, real service interworking test (scenario 4), and each scenario are composed of test setup, required resource, test pre-conditions, test procedure, observation results.

### 3.1 Optic Transmission Link Test (Scenario 1)

This test is to measure the optic signal source test, optic devices power test, bypass/add/drop channel power test, and OAM link status and service channel test as shown in Figure 3.

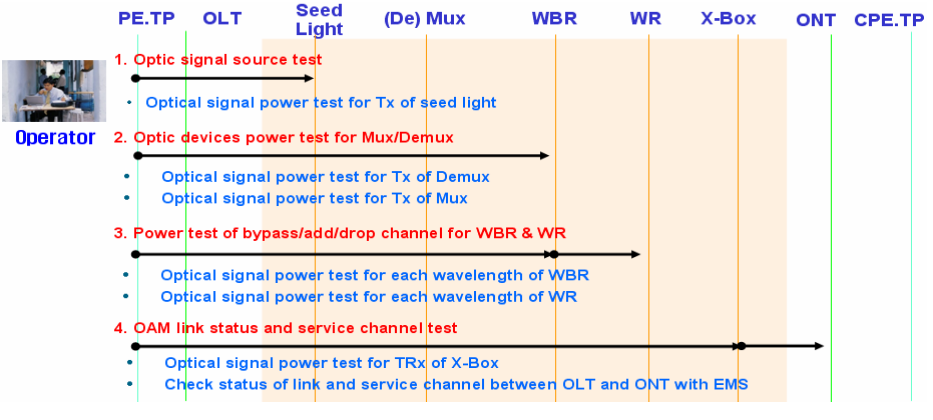



**Fig. 3.** Test procedure of optic transmission link test

### 3.1.1 Test Set-Up

The test setups for optic transmission link test are as follows

- Measure the optic power of seed light
- Measure the optic power of multiplexer and de-multiplexer
- Measure the optic power of WBR and WR
- Measure the optic power of service channel for X-box
- Measure the link status for OAM

### 3.1.2 Required Resource

The required resources for optic transmission link test are as follows

- Seed light
- Optic Spectrum Analyzer (OSA)
- OLT and ONU
- Optic power meter
- Optic link and X-box
- EMS manager and EMS server

### 3.1.3 Test Pre-conditions

The test pre-conditions for optic transmission link test are as follows

- The OLT, ONT, optic link, and seed light should be operated normally
- Power on the EMS server, EMS manager GUI, X-box, and seed light
- Start the SNMP running on EMS server and X-box

### 3.1.4 Test Procedure

The test procedures for optic transmission link test are as follows.

- Power on the seed light, OLT, and ONT
- Measure the wavelength and signal power at point **b** in Figure 1 with OSA
- Measure the optic power of multiplexer and de-multiplexer with power meter
- Measure the optic power of WBR and WR with power meter

### 3.1.5 Observation Results

The observation results for optic transmission link test are as follows.

- Optic power of seed light: 29 dBm
- Optic signal power of seed light: 128λ C/L-Band
- Optic power of multiplexer: 0dBm
- Optic power of de-multiplexer: -2dBm
- Optic power of WBR: -7/-6 dBm
- Optic power of WR: -9/-3 dBm
- The optic power of X-Box: -9/-3dBm
- Checking results of the link status information

### 3.2 Interworking Test between E-PON/G-PON and WDM-PON (Scenario 2)

This test is to check the operation status of E-PON/G-PON OLT, ONU, WBR, WR, multiplexer, de-multiplexer, and X-box during interworking between E-PON/G-PON and WDM-PON system.

#### 3.2.1 Test Set-Up

The test setups for interworking test is to verify each component operation status during interworking between E-PON/G-PON and WDM-PON system

#### 3.2.2 Required Resource

The required resources for interworking test between E-PON/G-PON and WDM-PON system are as follows

- Seed light
- Optic Spectrum Analyzer (OSA)
- OLT and ONU
- Optic link
- X-box, WBR, and WR
- Optic power meter
- EMS manager and EMS server

#### 3.2.3 Test Pre-conditions

The test pre-conditions for interworking test between E-PON/G-PON and WDM-PON system are as follows

- Seed light, OLT, ONU, WBR, and WR should be operated normally
- Configure the E-PON/G-PON OLT and ONT
- Prepare the traffic test analyzer

### 3.2.4 Test Procedure

The test procedures for interworking test between E-PON/G-PON and WDM-PON system test are as follows

- Power on the seed light, OLT, ONU, and X-box
- Measure the wavelength of the Seed light with OSA.
- Measure the power of multiplexer, de-multiplexer with optic power meter
- Measure the optic power of WBR and WR with power meter
- Log-in the EMS manager GUI
- Log-in the traffic tester
- Set-up the several types of traffics (unicast, multicast, broadcast)
- Transmit the traffics to the upstream, downstream, and both-stream (a)
- Measure and confirm the traffic results (b)
- Check the port status information (c)
- Repeat above-mentioned procedures of (a), (b), (c) by inserting and deleting the optic link and optic module

### 3.2.5 Observation Results

The observation results for interworking test between E-PON/G-PON and WDM-PON system are as follows

- Optic power of seed light: 20 dBm
- Optic signal of seed light: 128λC/L-Band
- Channel: Even and Odd
- Optic power of WBR: -7/-6 dBm at ***f*** point in Figure 1
- Optic power of WR: -9/-3 dBm at ***g*** point in Figure 1
- Traffic loss ratio between input traffic and output traffic: $1.3 \times 10^{E-11}$
- End-to-end traffic delay time: 250 us
- Throughput of link performance: 99%

## 3.3 Long Term Environment Test (Scenario 3)

This test is to check the running status of each component under long term high temperature and humidity conditions as shown procedure in Figure 4.

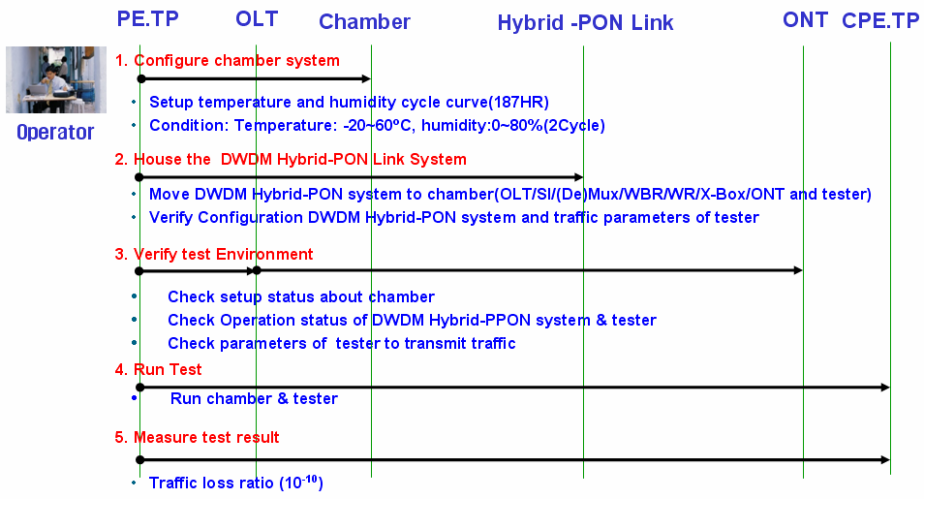



**Fig. 4.** Test procedure of environment and real service test

### 3.3.1 Test Set-Up

The test setup for long term environment test is to verify the running status of each component under high temperature and humidity conditions

### 3.3.2 Required Resource

The required resources for long term environment test are as follows

- Optic link and seed light
- Optic power meter
- OLT, ONU, X-box, WBR, and WR
- EMS Server and EMS manager
- Traffic generator of VoIP, data, IPTV stream , and broadcast stream
- Chamber

### 3.3.3 Test Pre-conditions

The test pre-conditions for long term environment and real service test are as follows

- Check the normal operation of seed light, OLT, ONU, X-box, WBR, WR, and EMS
- Configure the OLT, ONT for long term test
- Check the normal operation of the traffic tester and console
- Pre-define the transmission traffics (VoIP, data, IPTV stream, and broadcast stream)
- Check the normal operation of chamber for temperature and humidity test

### 3.3.4 Test Procedure

The test procedures for long term environment test are as follows

- Set-up the chamber for temperature and humidity test
- House the DWDM hybrid system in chamber
- Transmit the traffics (VoIP, Data, IPTV stream, and broadcast stream)
- Run the normal operations for 180 hours under -20$^{oC}$ ~ 60$^{oC}$ and 80% humidity
- Measure and confirm the traffic results
- Check the OLT, ONT, hybrid PON link status

### 3.3.5 Observation Results

The observation results for long term environment test are as follows

- Running results of chamber
- System status of OLT, ONT, X-box, seed light, and each links
- Traffic loss ratio between input traffic and output traffic: 1.3 x 10 $^{E-11}$
- End-to-end traffic delay time: 250 us
- Throughput of link performance: 99%

## 3.4 Real Service Interworking Test (Scenario 4)

This test is to verify the real service as VoIP, IPTV streaming service under normal conditions as shown procedure in Figure 5.

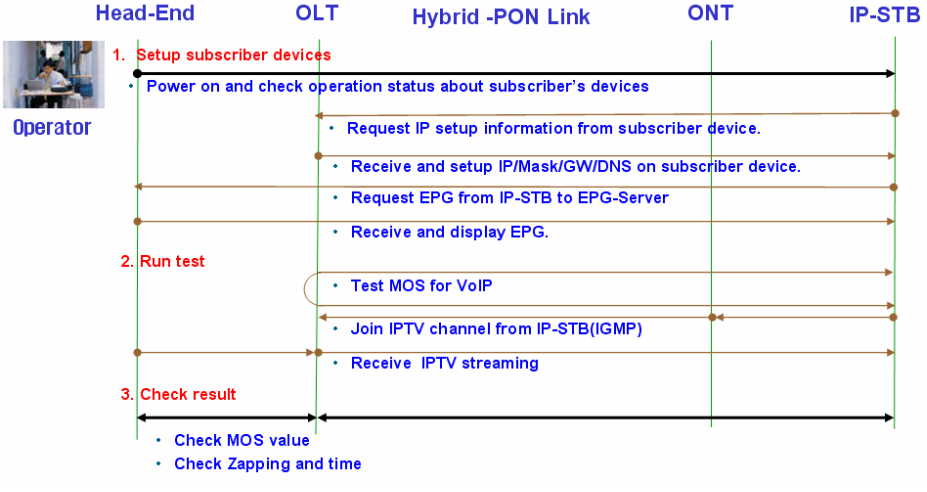



**Fig. 5.** Test procedure of real service interworking test

### 3.4.1 Test Set-Up

The test setup for real service interworking test is to verify the operation status of E-PON/G-PON OLT, ONU, and X-box in Figure 1.

### 3.4.2 Required Resource

The required resources for long term environment test are as follows

- Optic link and seed light
- Optic power meter
- OLT, ONU, X-box, WBR, and WR
- EMS Server and EMS manager
- Traffic generator of VoIP, data, IPTV stream , and broadcast stream
- HDTV terminal and set-up box
- VoIP phone and MOS tester

### 3.4.3 Test Pre-conditions

The test pre-conditions for long term environment and real service test are as follows

- Prepare the stream sever, EPG, DHCP, IPTV set-up Box, VoIP phone, MOS tester
- Normal operation of seed light, OLT, ONU, X-box, WBR, WR, HDTV, and EMS
- Configure the HD/SD streamer, server of EPG and DHCP, IPTV set-up box, VoIP phone, and MOS tester
- Normal operation of the MOS agent and console
- Broadcast channel for IPTV stream

### 3.4.4 Test Procedure

The test procedures for real service interworking test are as follows

- Power on the set-up box, VoIP phone, HDTV, MOS server, agent and console
- Check the EPG display of set-up box, VoIP phone, HDTV, MOS agent and console
- Run the VoIP traffic between MOS agents
- Confirm the HDTV video display quality
- Operate the channel zapping with remote controller
- Confirm the video quality on HDTV display
- Call set-up of the VoIP service and confirm the voice quality
- Check the MOS value results

### 3.4.5 Observation Results

The observation results for real service interworking test are as follows

- Setting time of IP address: 2 Sec
- EPG download time: 1.5Sec
- Zapping time of 8M SD Channel: <0.8Sec
- Zapping time of 20M HD Channel: <1.2Sec.
- VoIP MOS: 4.0 <

## 4 Conclusion

Until now on, many operators already have operated the E-PON or G-PON system with his own EMS, and these operators want to install the WDM-PON, especially for mobile backhaul implementation. The hybrid EMS has been developed for operation, maintenance and testing of hybrid PON system PON which are composed of E-PON, G-PON, and WDM PON system, along with the test scenarios. Especially the proposed test scenarios provide the guidelines and detail test methods for implementation of hybrid PON system to existing operators.

## References

1. Han, K.H., Son, E.S., Lim, K.W., Choi, H.Y., Jung, S.P., Chung, Y.C.: Bi-directional WDM Passive Optical Network using Spectrum-Sliced Light-Emitting Diodes. In: OFC 2005, Los Angeles, CA (2005); Paper MF98
2. Park, S.B., Jung, D.K., Shin, D.J., et al.: Demonstration of WDM-PON with 50 GHz channel spacing employing spectrum-sliced reflective semiconductor optical amplifiers. IEE Electron. Lett. 42, 1172–1173 (2006)
3. Park, S.J., Kim, G.Y., Park, T.S.: WDM-PON system based on the laser light injected reflective semiconductor optical amplifier. Optical Fiber Technology 12, 162–169 (2006)
4. Kim, T.Y., Han, S.K.: Reflective SOA-based bidirectional WDM-PON sharing optical source for up/downlink data and broadcasting transmission. IEEE Photon. Technol. Lett. 18, 2350–2352 (2006)
5. Won, Y.Y., Kwon, H.C., Han, S.K.: OBI noise reduction using gain saturated SOA in reflective SOA based WDM/SCM-PON optical links. IEE Electron. Lett. 42, 992–994 (2006)
6. Kim, T.Y., Kang, J.M., Han, S.K.: Performance analysis of bidirectional hybrid WDM/SCM-PON link based on reflective semiconductor optical amplifier. Microwave and Optical Technology Letters 48, 2306–2309 (2006)
7. Kang, J.M., Han, S.K.: A novel hybrid WDM/SCM-PON sharing wavelength for up- and down- link using reflective semiconductor optical amplifier. IEEE Photon. Technol. Lett. 18, 502–504 (2006)

# A Characterization of Mobility Management in User-Centric Networks*

Andréa Nascimento[1], Rute Sofia[1], Tiago Condeixa[2], and Susana Sargento[2]

[1] Informatics Systems and Technologies Research Unit,
Lusófona University, Portugal
{andrea.nascimento,rute.sofia}@ulusofona.pt
[2] Instituto de Telecomunicações, University of Aveiro, Portugal
{tscondeixa,susana}@ua.pt

**Abstract.** Mobility management is a key aspect to consider in future Internet architectures, as these architectures include a highly nomadic end-user which often relies on services provided by multi-access networks. In contrast, today's mobility management solutions were designed having in mind simpler scenarios and requirements from the network and where roaming could often be taken care of with previously established agreements. With a more dynamic behavior in the network, and also with a more prominent role from the end-user, mobility management has to deal with additional requirements derived from new Internet paradigms. To assist in understanding such requirements and also how to deal with them, this paper proposes a starting point to dismantle current mobility management notions. Our contribution is an initial proposal on defining mobility management in concrete functional blocks, their interaction, as well as a potential grouping which later can assist in deriving novel and more flexible mobility management architectures.

**Keywords:** Wireless networks, mobility management, user-centricity.

## 1 Introduction

Internet services and models have been going through a paradigm shift, product of three main factors: i) widespread wireless technologies; ii) increasing variety of user-friendly and multimedia-enabled terminals; iii) wider availability of open-source tools for content generation. Together, these three factors are changing the way that Internet services are delivered and consumed as there is a trend where the end-user has a particular role in controlling content as well as connectivity, based upon cooperation. These spontaneous environments, *user-centric* networks, rely on the notion that Internet users carry or own devices that may be part of the network. Hence the human roaming behavior of each user, be

* This work has been developed in the context of project UMM: User-centric Mobility Management, reference PTDC/EEA-TEL/105709/2008, sponsored by Fundação para a Ciência e Tecnologia.

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 314–325, 2011.

it individually or from an aggregate perspective, directly impacts the way the network is operated and perceived.

Human movement patterns in these environments may exhibit high variability as they are based on individual users routines and on users interests towards targets (e.g. locations, other users). Hence, mobility management is required to ensure adequate connectivity models and adequate network operation to support end-user expectations towards his/her roaming services. Considering the dynamics of user-centric networks and its self-organizing nature, it is crucial to attempt to develop end-to-end mobility management solutions more flexible than the ones existing today, as user-centric wireless networks are starting to heavily populate Internet fringes.

Currently, the most popular solutions for global mobility management have in common a model where a centralized and static *mobility anchor point* is responsible for keeping some form of association between previous and current identities of a mobile node. In user-centric environments, as explained, there is the need to better understand the roles that a mobility anchor point can have; the best location for these elements; and efficient ways to select the best anchor point for a mobile node. Moreover, considering that user-centric environments are heavily based on the users interests on being part of the network, and also assuming that the users might also control management functionalities, the period of time a mobility anchor point may or may not be available is highly variable. This poses extra stress on seamless and centralized mobility mechanisms, which have to manage handovers more often.

The aim of this paper is to provide an initial analysis of aspects that have to be considered when attempting to make end-to-end mobility management schemes more flexible. Our expectations are to contribute to an out-of-the-box notion of mobility management, by splitting mobility management as a whole into concrete functional blocks, and by explaining their impact and how to group such blocks. Our model is based on centralized solutions which, independently of the OSI Layer they tackle, are based in the same principles, roles, as well as similar operational behavior. Such splitting and categorization will give rise, in our opinion, to new mobility management architectures which are user-centric and more flexible.

The paper is organized as follows. In section 2 we describe related work, explaining the contributions that our work provides. Section 3 provides a few examples on user-centric networking scenarios, including for each a brief mobility characterization. In sections 4 and 5 we describe our study on mobility management, which is a characterization based on the current needs of this emerging user-centric networks, and in section 6 we conclude this work.

## 2 Related Work

This section provides a brief description on current work related to mobility management proposals which are based on different perspectives than the standardized solutions.

Bolla et al. consider the application of overlays to deal with mobility from a global perspective [3]. They provide a distributed mobility management scheme where mobility anchor points may be located within customer premises. The mobility anchor point itself is still a centralizing element as all the signaling goes through this device. Following the same line of thought, in order to deal with personal mobility and session migration Bolla et al. propose an application layer mobility framework [4] and the usage of a personal address, *"a network identifier dinamically assigned to a specific user for a specific communication session"*. The framework performs functions of personal mobility, terminal handover, session migration, and media adaptation for interactive multimedia applications. Although the authors are focused on addressing specific aspects of environments involving media, they do not attempt to analyze how to globally make mobility management more flexible.

Sofia et al. [14] propose an approach whose main objective is to separate control and data functionalities from the mobility anchor point into two different elements, in order to provide a more flexible mobility management framework, and to assist in developing non-centralized (e.g. distributed or hierarchical) mobility architectures. However, the authors do not present a proposal on how the communication between those separated elements can be performed, nor an analysis on why such splitting was relevant.

Chan [5] proposed the splitting of a mobility system into three logical functions: home network prefixes allocation, location management and mobility routing. The approach is based on the *Proxy Mobile IPv6* [8] extension for *Mobile IPv6* [9], and it is also proposed the usage of two mobility anchor elements, called *Home Mobility Anchor* and *Visited Mobility Anchor*. The main objective is to provide a system with mobility anchors distributed over different networks.

Having in mind the recent trend of flatter mobile network architectures, *Dynamic Mobility Anchoring* [2] [13] addresses the concept of "flattening" by confining mobility support in the access network, e.g. only confining it to access routers through a specific implementation of the application of Proxy Mobile IP. Following the same line of thought, i.e. IP mobility management in flatter mobile networks, Chan [6] describes the differences between centralized and distributed mobility management systems, as well as a list of potential problems and limitations of a centralized approach when compared with a distributed one.

Condeixa et al. [7] analyzed mobility management assumptions and requirements in user-centric scenarios, debating on challenges that need to be addressed to obtain a global mobility management solution considering user-centricity. The authors point out three major concerns for a mobility management system: binding definition, binding maintenance, and forwarding data problem.

Our work has in common with these approaches the motivation that by splitting, de-centralizing, or decoupling mobility management functionality into different blocks may assist in better understanding how and where to manage mobility. As described, most of today's attempts of flattening mobility management are being applied in the evolved packet core being the sole reason the

urgent need to simplify mobility management. We believe that understanding on how such mechanism may work is key to give rise to new research and business opportunities.

## 3 User-Centric Networking Notions

User-centric networks are environments where an Internet end-user owns and often carries devices that can share Internet access. These environments and the amount of end-user devices sharing Internet access are expected to grow, despite the limitations imposed by traditional operator-driven Internet communication models.

In our study, mobility management aspects are addressed from an end-to-end perspective but the analysis is applied in user-centric spontaneous wireless environments, which today correspond to the majority of technical scenarios on the last hop towards the end-user. Our user-centric environments are located within the customer premises region (where residential households, and enterprise environments reside). While in contrast, today's mobility management relies on functional blocks that are on the access or service regions.

Out of the several possible user-centric scenarios, we consider here three: a regular hotspot, a user-provided network (UPN) and a delay tolerant network (DTN). Each scenario is described both from an architectural perspective, as well as from a mobility characterization perspective. The line of thought driving this analysis is that these representative scenarios hold different requirements and are based on specific mobility assumptions. Hence, after providing a mobility characterization for each of the scenarios, the section concludes with a discussion which shall result in the identification of mobility functionality blocks, based on common requirements that each of these scenarios attain. A more complete description of these and of additional user-centric scenarios can be found in [16]:

- **Hotspot:** a hotspot scenario corresponds to the regular infrastructure mode in Wireless Fidelity (Wi-Fi) environments. This is currently the most common wireless architecture being deployed around us: each Internet enabled household corresponds to one hotspot. In this scenario mobility of users is local and confined to small regions, e.g. a room, an apartment, a small office. Moreover, if the user moves across different *Access Points (APs)*, then connectivity is expected to be intermittent. In a generic hotspot scenario users' mobility speed is low (pedestrian). Mobility inside each hotspot scenario is mostly managed at OSI layer 2; however, the IP address of the active user equipment's interface can change after a break. A key aspect to consider is that if current mobility management solutions are applied to this scenario, despite the fact that most of the movement is local, the mobility anchor point is located on the access or service regions.
- **User-provided networks:** UPNs [15] have been applied as complement to existing access networks: they allow expansion of infrastructures across one wireless hop. There is usually one individual or entity (the *Micro-Provider, MP*) which is responsible for sharing his/her connection with N-1 other users (out of a universe of N users, who today belong to a single community).

Moreover, a user is, in a specific community, simply identified by a virtual identifier (usually, a set of credentials username and password) which is stored by a *Virtual Operator (VO)* and relied upon whenever the user decides to access the Internet by means of a specific community hotspot. In these emerging architectures, the nodes that integrate the network are in fact end-user devices which may have additional storage capability and sustain networking services. Such nodes, being carried by end-users, exhibit a highly dynamic behavior. Nodes move frequently following social patterns and based on their carriers interests. The network is also expected to frequently change (and even to experience frequent partitions) due to the fact that such nodes, being portable, are limited in terms of energy resources.

– **Delay tolerant networks:** The DTN scenario relates to the need to establish on-the-fly an autonomous network within a disaster region (e.g. after an earthquake) based upon the devices that users in the region control and carry. Hence, such DTN consists of a network composed by users with a common objective (a community), grouped in regions. Some nodes move from region to region, establishing the communication between them (since gateways are mobile). Considering the main purpose of this kind of network, and the specific type of scenario where it is deployed, it is possible to establish behavior patterns on the mobility of the nodes, making possible to predict their location in a given instant and to schedule the delivery of information. In this case, the mobility pattern may also impact the routing process. Users moving may be good candidates to act as gateways, because they present a higher possibility of reaching other regions. It is important to notice that a region may be composed by only one user.

Table 1 summarizes the main characteristics related to the scenarios described, concerning inherent characteristics, and mobility behavior of the users on each of the scenarios presented. Based on a detailed analysis of the scenarios described [16] we consider a set of parameters that should be taken into account when characterizing any mobility management scheme: *i) identification*, which stands for the device identification both from a user and an access perspective; ii) *network scope*, which relates to the reach of the network; iii) *access control*, which relates to the location of the access control mechanism that is normally applied in each scenario; iv) *movement patterns*, related to the pattern that nodes are expected to exhibit in each scenario when roaming; v) *pause time behavior*, related to the time that a node exhibits a speed that is zero or close to zero; vi) *handover frequency*, related to the node having to switch between different networks or attachment points; vii) *connectivity sharing*, related to the sharing of Internet access.

In table 1 we provide a brief analysis on how each of the mentioned parameters relate to the three scenarios described. UPNs stand for a relevant case to address in terms of mobility management, as this scenario exhibits features that are not available on the hotspot scenario. The same conclusion can be drawn by looking at the DTN characterization. Both UPNs and DTNs exhibit aspects that were not considered when devising the current (centralized) mobility solutions.

**Table 1.** Summary of mobility characterization across user-centric scenarios

| Scenario/ Parameters | Hotspot | UPN | DTN |
|---|---|---|---|
| Identification | MAC address, credentials managed by WISP | Trust management scheme community credentials | Tokens or certificates; public/private key pair |
| Network scope | Small environment, e.g. household shops, universities | Small-large, e.g. household to village/city; varies dinamically | Small-large but static does not exhibit a quick growth |
| Access control | Centralized, on the provider | Decentralized and spontaneous | Decentralized |
| Node speed | Low | High | Varying |
| Expected movement frequency | Low | High and global | Low and routine based |
| Mobility pattern | Local mobility; preferred locations | Human/social patterns; short distance traveling preferred | Local mobility social patterns |
| Pause time | Long pause times | Mix, depends on location and user routine | Long |
| Handover frequency | Low | High | High |
| Connectivity | None | Yes | Yes |

## 4 Defining Mobility Management: A Characterization

This section is dedicated to a proposal on a global architectural definition of mobility management functional blocks, as well as roles based on the scenarios previously described.

### 4.1 Elements and Roles

In a mobility management system, three elements are considered in related literature: the *Mobile Node (MN)*, an end-user device for which a mobility service is provided; a *;Mobility Anchor Point (MAP)*, the element responsible for providing the mobility management service, it may reside in the network (e.g. router or access element) or in a server; and the *Correspondent Node (CN)*, that is any element engaged in active communication with the MN. These are generic roles that are today present in different management solutions, independently of the OSI Layer where the solution resides. For instance, in MIP [9] the MAP is the Home Agent (HA). In the *Session Initiation Protocol (SIP)* [12] the MAP is the SIP server. In a 3GPP architecture, the mobility anchor is centralized and located in the core network, having all traffic flowing through it, even if services to be used are locally placed closer to the MN.

Towards the idea of making mobility management more flexible (being the aim a reduced operational cost) Seite et al. and Chan et al. suggest to position the mobility anchors closer to the mobile nodes [13], ideally in the first element visible on the path from a MN perspective [6]. Sofia et al. proposed the separation of management functionalities into two elements, attempting to decouple data plane and control plane [14]. In the proposed architecture, the HAC (control plane element) is located in a server, and HADs (data plane elements) are positioned in the access nodes, close to mobile nodes. Chan relies on the Proxy Mobile IP [8], and also splits the mobility anchor functionalities into three logical blocks [6]. Although the author states that those functionalities are placed in the home network, they do not need to be placed in the same physical entity. Those works can be considered as a first step towards an architecture where the management functionalities are splitted and distributed in different places in the network.

Such approaches, the positioning of the MAP as well as the definition of interactions between the different roles of mobility management have been object of heavy analysis. Still, today there is not truly consensus in where MAP and additional functionality should reside. Such positioning depends on the network architecture and requirements; on the OSI Layer being tackled, as well as on the overall complexity from a technical and policing perspective. Considering that user-centric networks present particular characteristics (e.g. there is no clear splitting between network elements and end-devices), the current centralized standards may not be suitable. Thus, a novel mobility management approach should be designed for such networks, considering all its particularities and following this trend of rethinking the mobility anchor point element.

Therefore, thinking about mobility management functioning in a fine-grained way, we have identified a group of functionality blocks. Based on the dynamics of user-centric networks, the first step towards a more suitable mobility management approach is by understanding and further analyzing the basic tasks a mobility management should provide.

### 4.2 Functional Blocks

In order to perform a mobility management characterization, as result of an initial analysis on current available mobility management approaches and standards, we have identified the following mobility management functional blocks:

- **Device identification:** corresponds to the network identification for the MN. Usually the main mechanism for a location management is the association between the device's *known-address* and the device's *real-address*. In MIP, known-address and real-address are IP addresses; in SIP, the known-address is a URI, and the real-address is an IP address. In MIP the device identification control is the Home Agent (HA)/Correspondent Node (CN) cache binding. In SIP, it is the user database used by the Proxy server.
- **Identification database control:** corresponds to the mechanism that is applied to control the database identification. This is normally a block relevant from an access perspective, which today follows a centralized approach.

- **Binding mechanism:** it is the signaling related to the device's register to the mobility system. It creates/updates a record in the identification database control, associating the known-address to the real-address. In MIP it is the Binding Update message sent to a HA/CN. In SIP it is the REGISTER message sent to the Registrar server.
- **Routing or forwarding:** it is the process of intercepting the packets destined to the known-address, encapsulating them with the real-address, and forwarding them. In MIP this is performed by the HA; in SIP this process is performed by an element named RTP translator (when it is used).
- **Handover negotiation:** the process taken when the device has its real-address changed. It involves negotiation and signaling. The main objective is to guarantee that the user will keep active all its sessions during the handover process. In MIP, the handover negotiation may be anticipated with the Fast Handover extension [10], and the SIP does not implement any anticipation, performing a re-negotiation after the connection between the peers is lost.
- **Resource management:**the resource management is a necessary procedure for the mobility management to guarantee the quality of the connection when the MN changes its point of attachment to the network. However, it is not provided by most of the mobility management approaches. The 802.21 Media Independent Handover (MIH) [1] standard is focused on the handover process based on a resource management aware negotiation for vertical handovers.
- **Mobility estimation:** it is the procedure of changing the MN point of attachment to the network before its current connection breaks. The extension Fast Handovers for MIP, and the 802.21 MIH provide this functionality.
- **Security/privacy:**it refers to any security or privacy mechanism used to assure the integrity of the elements and signaling in the mobility management system.

### 4.3 Discussion on Mobility Characterization

Based on the block characterization there are a few aspects worth to highlight. Firstly, today's mobility management solutions completely ignore the need for adequate resource management. However, this is a crucial aspect for cellular or wireless networks, in particular for session continuity. Database control is normally centralized, an aspect which may not be compatible with the notion of communities that user-centric networks embody. Routing and forwarding is also based on mechanisms (e.g. proxy mechanisms) which may not be completely compatible with the fact that users in our scenarios are expected to roam frequently. This is an aspect that can be improved by integrating mobility estimation mechanisms. Security and privacy aspects are also often disregarded.

Moreover, analyzing the identified blocks, one can notice that there are a few categories onto which they seem to be naturally grouped. Firstly, they can be grouped into *data plane* and/or *control plane*. It is also possible to group the functionality blocks into *location management* and/or *handover management* procedures.

These are aspects that we debate on the next section in an attempt to raise awareness to new and more flexible mobility management schemes.

# 5 Deconstructing Mobility Management Centralized Approaches

This section delves into the potential development of a mobility management architecture that is more adequate to the emerging wireless scenarios described in section 3.

As of today the functional blocks described reside both on the MN and mobility anchor point, being the functionality fully controlled in the later one, which is physically located in the access or service regions. Our aim in analyzing initial forms of deconstructing the need for a centralized mobility management scheme is motivated by the need to find simple and operational ways to split such functionality, as well as ways to "push" such functionality closer to the end-user, having in mind an optimization of mobility management in the context of the scenarios described.

## 5.1 Location and Handover Management Categorization

Mobility management usually is mentioned as consisting of two main blocks: location management and handover management. Location management is the block responsible for locating the devices, i.e. for guaranteeing that they are always reachable, independent of their point of attachment to the network. The handover management block is responsible for maintaining active sessions while MNs roam. Therefore, from a high level perspective, mobility management functionality can be split into these two main blocks. Today, these blocks both reside on the mobility anchor point and are based on information provided by the MN. Solutions such as the *Host Identity Protocol (HIP)* [11] attempt to provide a decoupling by isolating location management and handover management. Other solutions (e.g. Hierarchical Mobile IP [17]) optimize handover management by scoping the extent of the impact of such negotiation.

## 5.2 Control and Data Plane Categorization

Another way to categorize mobility management functionality is to consider a splitting between control and data planes. As part of the control plane we can cite all the procedures related to the signaling, and the data plane is related to the data traffic, routing, forwarding and address translation. Figure 1 shows the relationship between the blocks, in order to identify the communication between them. It shows also the classification concerning data and control planes, and location and handover management.

Between the functional blocks, it is possible to identify two types of communication, in regards to its periodicity. *Periodic communication* is related to procedures that need to be performed in a regular basis, in order to maintain the system updated. The *occasional communication* is related to the procedures performed only as result of a change in the system, for instance, when a MN performs a handover from one point of attachment to another.

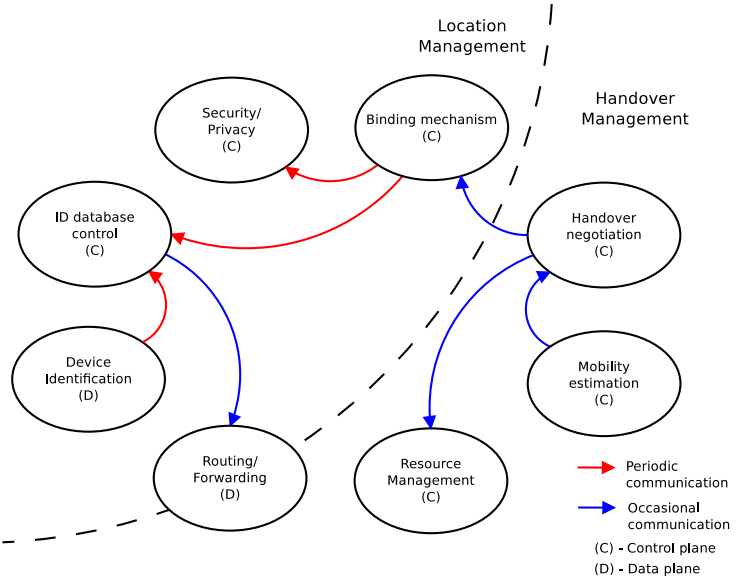



**Fig. 1.** Mobility management functional blocks

Usually, all the communication between the blocks of the handover management side of the picture is triggered when a node movement is detected, or predicted. When a handover is detected, the mobility estimation block triggers the handover negotiation, which will take part in the process. The handover negotiation needs to consult the resource management in order to guarantee that the user will be "always best connected". For the handover process to complete, the binding mechanism is triggered, so it can update the location information in the identification database control. The identification database control then updates the information in the element responsible for routing/forwarding.

The binding mechanism has a periodic communication with the ID database control, because it is the procedure performed to maintain the ID database control updated. It needs to use the security/privacy procedures to guarantee that no third part could take place in the communication.

### 5.3 User Perspective and Access Perspective Categorization

Currently, the available mobility management approaches offer most of the functionalities described here, but none of those approaches offer all of the functionalities. Those functionalities are placed in different locations in the network and customer premises, and most of them are centralized in one unique element (usually the mobility anchor point). By taking this perspective, we can categorize the blocks into two groups, blocks located in the user perspective and in the access perspective as provided in table 2.

**Table 2.** Location of mobility management functional blocks

| Parameter | Access and user perspective categorization |
|---|---|
| Device Identification | User |
| ID database control | Access |
| Binding mechanism | User and access |
| Routing / Forwarding | Access |
| Handover negotiation | User |
| Resource management | Access and user |
| Security/privacy | User |
| Mobility estimation | Access and user |

Table 2 shows the current location of each block. It is important to notice that this location is based on current mobility management approaches functioning.

## 6 Conclusion

This paper provides a study and a new perspective on ways to make end-to-end mobility management schemes more flexible, being the motivation the fact that user-centricity and in particular user-centric environments are a crucial part of the future of the Internet. We went over three different cases of spontaneous wireless deployments abounding around us, and characterized each from a mobility perspective. Based on such characterization we have derived a set of parameters and functional blocks, and discussed ways to attempt to de-construct the need for centralized architectures, starting by proposing concrete categories to tackle.

As follow-up of this work we intend to take advantage on the blocks identification and data/control planes and location/handover management categorizations to evaluate what is the best location for each of the identified functional blocks. Focusing on the user-centricity, the objective is to perform a deeper study on each of those functionality blocks, in order to identify which of them could be placed into customer premises equipment. Placing mobility management functionalities in the customer premises could provide a mobility system user-centric and independent of the access network. A deeper study should clarify if that is possible, and what is the cost to maintain such approach. Hence, as next steps we intend to address ways to bring mobility management closer to the customer premises in a way that is adequate for the network, while keeping the end-user agnostic in regards to the complexity. A second step to be considered is to analyze such splitting based on the potential impact that it may have both from an end-user and from an access perspective.

## References


1. IEEE 802.21 Working Group, http://ieee802.org/21/
2. Bertin, P., Bonjour, S., Bonnin, J.M.: An evaluation of dynamic mobility anchoring. In: 2009 IEEE 70th Vehicular Technology Conference Fall (VTC 2009-Fall), pp. 1–5 (September 2009)
3. Bolla, R., Ranieri, A., Rapuzzi, R., Repetto, M.: Moving towards user-centric paradigms for internet mobility. In: International InterMedia Summer School (June 2009)
4. Bolla, R., Rapuzzi, R., Repetto, M.: A user-centric mobility framework for multimedia interactive applications. In: 6th International Symposium on Wireless Communication Systems, ISWCS 2009, pp. 293–297 (September 2009)
5. Chan, H.: Proxy Mobile IP with Distributed Mobility Anchors. In: IEEE Globecom Workshop on Seamless Wireless Mobility (2010)
6. Chan, H.: Problem statement for distributed and dynamic mobility management. draft-chan-distributed-mobility-ps-02 (March 2011) (work in progress)
7. Condeixa, T., Matos, R., Matos, A., Sargento, S., Sofia, R.: A new perspective on mobility management: Scenarios and approaches. In: Second International ICST Conference on Mobile Networks and Management (September 2010)
8. Gundavelli, S., Leung, K., Devarapalli, V., Chowdhury, K., Patil, B.: Proxy Mobile IPv6 - RFC5213 (August 2008), http://www.ietf.org/rfc/rfc5213.txt
9. Johnson, D., Perkins, C., Arkko, J.: Mobility Support in IPv6 - RFC3775 (June 2004), http://www.ietf.org/rfc/rfc3775.txt
10. Koodli, R.: Mobile IPv6 Fast Handovers - RFC5568 (July 2009), http://www.ietf.org/rfc/rfc5568.txt
11. Moskowitz, R., Nikander, P.: Host Identity Protocol (HIP) Architecture - RFC4423 (May 2006), http://www.ietf.org/rfc/rfc4423.txt
12. Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., Schooler, E.: Session Initiation Protocol - RFC3261 (June 2002), http://www.ietf.org/rfc/rfc3261.txt
13. Seite, P., Bertin, P.: Dynamic mobility anchoring. draft-seite-netext-dma-00.txt (May 2010)
14. Sofia, R., Hof, A., Wevering, S.: Method for packet-based data transmission in a networkhaving mobility functionality (January 2008)
15. Sofia, R., Mendes, P.: User-provided networks: Consumer as provider. IEEE Communications Magazine 46, 86–91 (2008)
16. Sofia, R., Nascimento, A., Sargento, S., Condeixa, T., Matos, R.: User-centric mobility management - D1: Use-cases. Tech. rep., Lusófona University (March 2011)
17. Soliman, H., Castelluccia, C., ElMalki, K., Bellier, L.: Hierarchical Mobile IPv6 (HMIPv6) Mobility Management - RFC5380 (October 2008), http://www.ietf.org/rfc/rfc5380.txt

# Network Attack Detection at Flow Level*

Aleksey A. Galtsev** and Andrei M. Sukhov

Samara State Aerospace University,
Moskovskoe sh., 34, Samara, 443086, Russia
galaleksey@gmail.com, amskh@yandex.ru

**Abstract.** In this paper, we propose a new method for detecting unauthorized network intrusions, based on a traffic flow model and Cisco NetFlow protocol application. The method developed allows us not only to detect the most common types of network attack (DDoS and port scanning), but also to make a list of trespassers' IP-addresses. Therefore, this method can be applied in intrusion detection systems, and in those systems which lock these IP-addresses.

**Keywords:** DDoS attack, flow traffic model, Cisco NetFlow.

## 1 Introduction

Currently, Internet information resources are actively growing, penetrating many spheres of social life. Information technologies are being introduced not only into private enterprises, but also in the provision of public services. With each passing day, more and more confidential transactions are carried out via the Internet. In relation to these trends, the question of computer networks security is starkly raised. Attackers have developed and actively use many types of network intrusion [1,2,3,4], most of which can be prevented by standard methods of protection.

This article focuses on detecting and preventing network attacks of two types that are impossible to prevent by the standard settings of information resource software. This are "Distributed Denial of Service" attacks (DDoS attacks) [4,5,6] and port scanning, that are used to find bottlenecks in network information systems. In recent years the rate of end-user connections to the Internet has increased sharply, which has given rise to an increase in the number and intensity of attacks such as DDoS. These attacks are highly damaging to the information service, and at the same time simple in their execution. Port scanning is used by hackers to conduct "network intelligence". In this article we would like to propose a new method of detecting DDoS attacks and port scanning, based on the Cisco NetFlow protocol [7,8].

NetFlow is a network protocol developed by Cisco Systems for collecting IP traffic information. NetFlow has become an industry standard for traffic monitoring and is supported by platforms other than Cisco IOS (Internetwork Operating System) and NXOS (Nexus Operating System) [9].

* This work submitted by part of NPRIR 01200964488.
** Corresponding author.

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 326–334, 2011.

A network flow has been defined in many ways. The traditional Cisco definition is to use a 7-tuple key, where a flow is defined as a unidirectional sequence of packets sharing all of the following 7 values:

- Source IP address
- Destination IP address
- Source port for UDP or TCP, 0 for other protocols
- Destination port for UDP or TCP, type and code for ICMP, or 0 for other protocols
- IP protocol
- Ingress interface (SNMP ifIndex)
- IP Type of Service

The proposed method for detecting network attacks based on the traffic flow model is described in [10]. Traffic models shows that two parameters, the load of the channel and the number of active flows in it, must be used for a full representation of the network state. In this paper, the criteria of abnormal network conditions, which can determine the start of the attack, were formulated. A more detailed model is described in Section 2. The values of these parameters can be measured using the NetFlow protocol, implemented on Cisco routers. In [11] and [12], the authors suggested that the traffic flow model can be used for network security problems, in particular to detect network attacks such as DDoS, port scanning and network worms. Also in [13] and [14], an attempt was made to use the NetFlow protocol to detect DoS attacks such as Smurf and worms W32.Blaster Worm and Red Worm.

The aim of this work is to show that the NetFlow protocol can be used for the detection of DDoS attacks and port scanning, and to formulate an algorithm to identify IP-addresses from which the attack is carried out. This algorithm enables to the creation of "black lists" of addresses that should be blocked to prevent the attack. This article is organised as follows:

- Section 2 - describes the flow traffic model, on which a method for detecting network attacks has been built
- Section 3 - experiment to study the various attacks
- Section 4 - the definition of the detection algorithm under consideration
- Section 5 - describes the Research Center of DDoS attacks at the Samara State Aerospace University

## 2 Traffic Model

In this paper, we would like to propose a method of diagnosing the backbone links and testing it on existing networks. This method is based on a traffic model [10], according to which the number of active flows can be considered as an important characteristic of the real network state. Two variables, the number of active flows and the utilisation of the channel, best describe the current network state. Analysis of all data on the network, represented by individual points on the plane

with axes, which are plotted the number of active flows and utilization of the network, allows to the definition of three areas that correspond to qualitatively different states of the network.

It has been previously shown [10] that the first part of the curve formed from average values of the data produces a straight line, which ends with an inflection point. The straight line corresponds to the part of the network that is characterised by a minimal loss of IP-packets, less than a half of one percent. The bent part of the curve corresponds to an overloaded network, and is characterised by a significant packet loss of up to 5%, which reduces the effective size of the transferred segment of the TCP/IP. The third, nearly horizontal, portion of the curve corresponds to a completely unusable network with significant packet loss of over 5%.

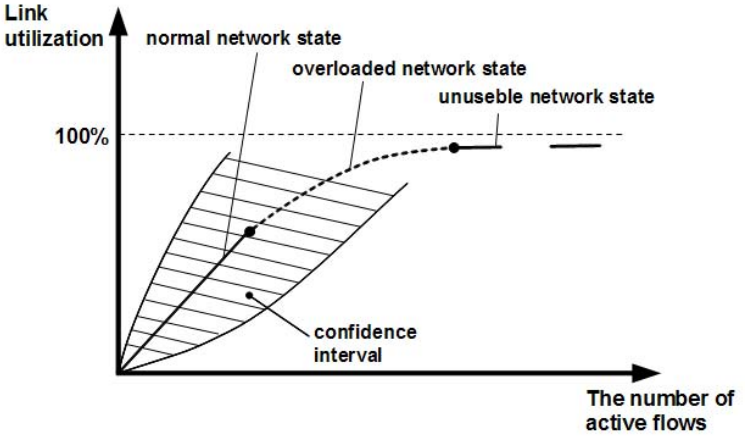



**Fig. 1.** Traffic model

The distribution of total load tends to a normal (Gaussian) distribution, since the total load of the studied channel is the result of multiplexing a large number of flows that are independent of each other. The theoretical model allows us to estimate the confidence interval for the working area of the curve:

$$B(t) = b(N + kA(\epsilon)\sqrt{N}) \tag{1}$$

Here $b$ is average loading on router, $A(\epsilon)$ is the normal quantile function and coefficient $k$ should be found experimentally. Equation 1 indicates that the real state of the network, described by the number of active flows $N$ and the flow data rate $B(t)$, will be outside these limits in only $100\% \times \epsilon$ of the total observation time.

The traffic model presented here allows the formulation of a simple criterion for finding anomalous network states: if several consecutive measurements go beyond the confidence interval of $\epsilon$=0.05, we can confidently consider problems on the network. If we collect the data every 5 minutes, then the statistics of a

few hours will make it possible to determine all the parameters of equation 1 with reasonable accuracy.

Presumably, the network state will be out of the confidence interval during the progress of a network attack. During port scanning, the number of active flows will increase with a nearly constant load, as the data transmitted is only limited to establish the connection and to close it. The channel load as well as the flow number should sharply increase during the progress of DDoS attacks. In order to prove these hypotheses it was decided to conduct two experiments with network scanning and with a DDoS attack.

## 3 Experiment

In order to clarify the details of unauthorized intrusion, it was decided to perform experiments that emulated attempted attacks. Experiments were carried out on the network of the Samara State Aerospace University (SSAU).

Remote machines were used as the source of the attack which were located in an external network. The utility Nmap was applied for port scanning, which was ordered to carry out a full scan of all hosts on the network.

A Web server was selected as the target during the progress of the DDoS attack. A few computers located in the external network were the sources of the attack. In the first part of the experiment the attacking computers sent ping requests simultaneously within half an hour. In the second part of the experiment the target computers were attacked (DDoS attack) with the help of a specialised program, LOIC (Low Orbit Ion Cannon). The Web server was attacked with the use of different types of traffic (HTTP, UDP, TCP) over an hour. Data were collected from all experiments, which are then analysed to identify patterns of different types of attacks.

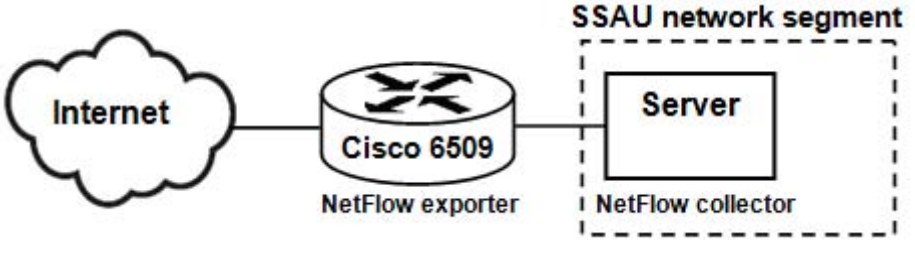



**Fig. 2.** The experimental scheme

Flow data, that are the basis for the analysis, were collected from the boundary router Cisco 6509 of the SSAU network. NetFlow collector nfdump [15] was used to gather data from the router. NetFlow export data is taken for analysis at regular intervals of five minutes. A file with the parameters of all the flows recorded on the router is formed every five minutes. The parameters are listed

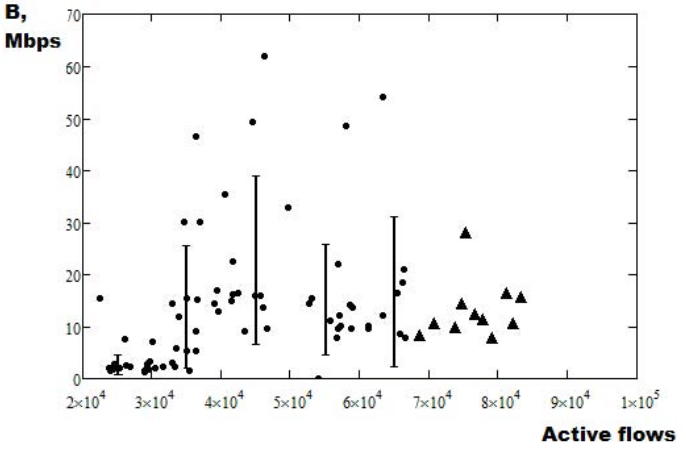



**Fig. 3.** Port scanning

in the introduction, and include the beginning of the stream, the duration of the stream, the data transfer protocol, source address and port, destination address and port, the number of transmitted packets, and the amount of transferred data in bytes.

The analysis of data collected during network scanning has revealed a sharp increase in the number of active flows for almost the same amount of traffic transferred (see Figure 3). Each scanning computer generated of the order of 10-20 thousand of very short flows (up to 50 bytes) within 5 minutes. In the testing period the total number of active flows on a router that is generated by all sources is about 50-60 thousand.

Figure 3 shows a graph of the network states, the X-axis displays the number of completed flows N, the Y-axis displays the total load in Megabits per second (Mbps). Each point on the graph reflects the network state of the preceding five-minute interval, showing the dependence of the average channel load on the number of active flows. The dots correspond to the normal network state and the triangles describe the state of the network, registered during a port scanning. Segments are depicted on the graph's parallel vertical axis and show the confidence intervals for the average load calculated for five flow intervals (20000-30000, 30000-40000, 40000-50000, 50000-60000, 60000-70000).

As a result of the experiment with the ping requests, it was found that every attacking computer accounts for a very long flow of ICMP traffic, if we send requests through a single port. The data has been subsequently written into a nfdump file after the attack is finished, making it difficult to detect. It should be noted that one active ICMP flow to identify the occurrence of a failure in the information system is clearly inadequate; the number must extend to the tens of thousands of requests.

The analysis of modelling the DDoS attack by the LOIC utility also showed a sharp increase in the number of active flows, along with an increase in the traffic. The utility sends data in parallel to different ports of the target, thereby

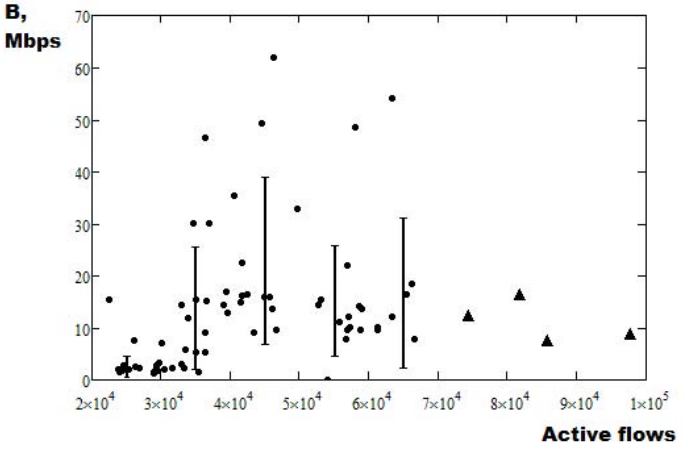



**Fig. 4.** DDoS attack

creating a large number of short flows for up to a minute (see Figure 4). The triangles show the network states recorded during the attack.

Thus, it becomes apparent that the NetFlow protocol may not only reveal the start of the attack, but also determine its type. A detailed description of attack detecting algorithms and work to create secure hosting services may be found in the following sections.

## 4 Algotithm for Intrusion Detection

Our studies have revealed patterns, based on the NetFlow data, that allow the IP addresses of the computers with which conducted DDoS attacks and port scans to be determined. Based on these patterns we developed an algorithm for attack detection. Before formulating the algorithm we will specify the format for recording flow data:

- Date and time of flow
- Duration of flow (in seconds, up to thousandths)
- Transfer protocol
- IP address and source port
- IP address and destination port
- The number of transmitted packets
- The number of bytes transferred

The algorithm developed for the detection of attacks such as DDoS and port scanning is:

1. Find IP-addresses of sources that generate a large number of flows.
   (a) If the size of these flows is very short, up to 50 bytes, it is most likely port scanning.

   (b) If the duration of the flows is greater then this IP-address might be carrying out DoS attacks.
2. Find IP-addresses of sources that generate very long streams (lasting more than 5 minutes). The IP-address assignment can be carried out DoS attack in this case.

If many IP-addresses from which a potential DoS attack are found, we may classify this attack as DDoS.

In order to prevent any network attack, early detection is important to enable steps to be taken to neutralise it. NetFlow data comes from the router from time to time, depending on the settings. At the same time, a balance between the frequency of collection of flow statistics and the time needed for processing is also needed. Therefore it was decided to establish the frequency of querying the NetFlow data to once a minute.

It should be noted that the NetFlow statistics provide information on flows that are already completed. Since the flow is considered as active for a certain time after its completion, completed flows also need to be considered active.

## 5 DDoS Attacks Study Centre (Secure Hosting Creation)

We have developed practical algorithms that are implemented as a script in Perl. The script has been installed on the protected server. NetFlow data comes to the server running the NetFlow nfdump collector from the boundary router (BGP) on the SSAU network every minute. The script receives a file with data on entering flows. These data are processed by a script in accordance with the attack detecting algorithm described in the previous section. A list of suspicious IP addresses, from which an attack may be carried out, are produced as the output of the script.

The processing time of the NetFlow data is very small (tens of a millisecond), whereas the intrusion detection addresses will be equal to the period of the export data from the router, i.e. one minute.

Suspicious IP addresses are entered into the database and all traffic from those addresses are blocked by an iptables firewall [16] for 5 minutes. Iptables is installed on the protected server, i.e. only the server is protected, not the whole network. If necessary, the protection can be extended to a whole SSAU network, blocking suspicious IP address on the boundary router. In the coming year we plan to explore the possibilities of using the NetFlow protocol for the detection of DDoS attacks, for a combination of several basic types of attacks.

The problem of preventing DDoS attacks, as well as unauthorized network intrusion; do not lose their sharpness, so SSAU created the Centre for the Study of Network Attacks. The main purpose of the new centre is to develop new techniques to detect and prevent various types of unauthorized network intrusion. Hosting that is protected from DDoS attack has been created inside one segment of the university network. The method of protection is based on the method presented in this article. The server that is running the NetFlow collector receives

NetFlow data from the boundary router of university network. This data is then processed to produce a "black list" of addresses that are blocked by an iptables firewall.

## 6 Conclusion

In this paper, the detection of attacks such as DDoS and port scans using a flow traffic model was proposed, based on receiving data according to the Cisco NetFlow protocol from the border routers. An experiment to test this model and create prevention algorithms has been described. The experimental results have confirmed that the proposed flow traffic model can be used effectively to detect these attacks.

An algorithm for detecting suspicious IP addresses that can go attack was suggested. These addresses can be used in intrusion prevention systems in order to block them. Also, the algorithm for the detection of suspicious addresses was implemented as a script that works in conjunction with a firewall iptables. This system of detecting and preventing attacks such as DDoS and port scanning was installed on the SSAU host network. In the future we plan to continue studying the possibility of using the NetFlow protocol to detect various types of unauthorized network intrusion. It is also planned to create a network protection system directly on the SSAU network boundary router using Cisco IOS features.

## References

1. Paulauskas, N., Garsva, E.: Computer System Attack Classification. Electronics and Electrical Engineering 2(66) (2006)
2. Mirkovic, J., Reiher, P.: A taxonomy of DDoS attack and DDoS defense mechanisms. ACM SIGCOMM Comput. Commun. 34(2), 39–53 (2004)
3. Hussain, A., Heidemann, J., Paradopoulos, C.: A Framework for Classifying Denial-of-Service Attacks, Karlsruhe, Germany, pp. 99–110 (2003)
4. Douligeris, C., Mitrokotsa, A.: DDoS Attacks and Defense Mechanisms: Classification and State-of-the-art. Comp. Networks 44, 643–666 (2004)
5. Paxson, V.: An Analysis of Using Reflectors for Distributed Denial-of-Service Attacks. CCR 31(3) (July 2001)
6. Chang, R.K.C.: Defending against Flooding-based Distributed Denial of Service Attacks: A tutorial. IEEE Communications Magazine 40(10), 42–51 (2002)
7. Cisco IOS NetFlow site, Cisco Systems, www.cisco.com/go/netflow
8. Claise, B.: NetFlow Services Export Version 9. RFC 3954 (2004)
9. White Paper: Cisco IOS and NX-OS Software Reference Guide, Cisco Systems, http://www.cisco.com/web/about/security/intelligence/ios-ref.html
10. Afanasiev, F., Petrov, A., Grachev, V., Sukhov, A.: A Flow-based analysis of Internet traffic. Russian Edition of Network Computing 5(98), 92–95 (2003)
11. McGlone, J., Marshall, A., Woods, R.: An Attack-Resilient Sampling Mechanism for Integrated IP Flow Monitors. In: 29th IEEE International Conference on Distributed Computing Systems Workshops (2009) ISBN: 978-0-7695-3660-6
12. Yang, W., Gong, J., Ding, W., Wu, X.: Network Traffic Emulation for IDS Evaluation. In: IFIP International Conference on Network and Parallel Computing, pp. 608–612 (2007) ISBN: 978-0-7695-2943-1

13. Deal, R.A.: Cisco Router Firewall Security: DoS Protection (October 2004), www.informit.com/articles/printerfriendly.aspx?p=345618
14. Fullmer, M., Roming, S.: The OSU Flow-tools Package and Cisco Netflow logs. In: Proceedings of the 2000 USENIX LISA Conference, New Orleans, LA (2000)
15. Haag, P.: Watch your Flows with NfSen and NfDump. In: 50th RIPE Meeting (2005)
16. Marmorstein, R., Kearns, P.: A tool for automated iptables firewall analysis. In: 2005 USENIX Annual Technical Conference, FREENIX Track, pp. 71–82 (April 2005)

# An Intelligent Routing Protocol for Delay Tolerant Networks Using Genetic Algorithm

Saeid Akhavan Bitaghsir[1] and Faramarz Hendessi[2]

[1] Department of Science & Engineering,
Sharif University of Tech., International Campus, Iran
saeid_akhavan@kish.sharif.edu
[2] Department of Electrical & Engineering,
Isfahan University of Tech., Isfahan, Iran
hendessi@cc.iut.ac.ir

**Abstract.** Due to the dynamic topology of vehicular ad hoc networks, routing of packets in these networks faces a lot of difficulties. The situation will become more challenging when we have to deal with Delay Tolerant Networks (DTN) which are also sparse and partitioned and we need to use some vehicles to store the packets and carry them from one partition to another. Despite all these difficulties, by looking through movements of vehicles in an urban environment, we can find out that the topology of the network does not change in a pure random way and we can bring the traffic models of streets into account for having better routing performance. In this paper we proposed an intelligent routing protocol for delay tolerant networks using genetic algorithm as the learning method for choosing the best vehicle to carry the packets from one partition to another.

**Keywords:** Delay Tolerant Networks, Routing Protocol, VANET, Learning, Genetic Algorithm.

## 1 Introduction

Vehicular Ad hoc Networks are a subclass of Mobile Ad hoc Networks in which most of the nodes are mobile and the topology of the network changes dynamically. The aim of Vehicular Ad hoc Network routing protocols is to establish efficient routes between network nodes adaptable to rapidly changing topology of vehicles in motion. These routes enable us to form a self-organized network between vehicles without the need for a permanent infrastructure.

Although nodes are mobile in VANET, they have a distinct controlled mobility pattern that is subject to vehicular traffic regulations. It is because vehicles are generally constrained to roadways and streets. Also by looking through a real urban environment and by analysing the mobility model of vehicles in the streets, we can find out that despite the rapid changes in the topology of VANET, we can bring the traffic model of roads into account for our routing protocol. For example in our simulation environment we supposed to have some major roads with high density of vehicles, some streets with mediocre traffic and some bystreets with low density of

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 335–347, 2011.

vehicles. Then by using Genetic Algorithm as the learning method, we tried to detect the proper vehicle for storing the packet and carrying it from one partition of our network to another.

There are two categories of routing protocols in VANET: topology-based and geographic routing. Topology-based routing protocols use the information about links that exist in the network to perform packet forwarding. Geographic routing protocols use neighboring location information to perform packet forwarding. Since link information changes in a regular basis, topology-based routing protocols suffer from routing route breaks [5]. The idea of geographic routing protocols is that each intermediate node chooses one of its neighbors which is geographically closer to the destination as the next hop for forwarding the packets. This procedure may fails when a node couldn't find any neighbour closer to the destination. This situation is called "Local Maximum". Several routing protocols have been proposed (GPSR [7], GPCR [4], VCLCR [2]) in order to recover from local maximum.

Sometimes in Vehicular Ad hoc Networks, there exists no end-to-end route between the source node and the destination node. It occurs when the density of vehicles in streets decreases or when law enforcement, military, and financial armored vehicles may each wish to exchange data privately within their own vehicular network, due to the sensitivity of the information exchanged. Even in densely-populated urban scenarios, sparse sub-networks can be prevalent. In these situations, Delay Tolerant Network (DTN) routing algorithms are needed [6]. To make the communication possible in this kind of networks, intermediate nodes take custody of the data being transferred and forward it as the opportunity arises. Both links and nodes may be inherently unreliable and disconnections may be long-lived. Some routing protocols (e.g. GeOpps [3], MoVe [11]) have been proposed to overcome the Delay Tolerant Networks problems.

In this paper we proposed a routing protocol for Delay Tolerant Networks which is a combination of efficient position-based routing for connected partitions of our simulated network and delay tolerant forwarding using genetic algorithm for routing between partitions of our network.

The rest of the paper is organized as follows: In section 2 we discuss about "GeoDTN+Nav", a related work to our proposed method. Section 3 presents our proposed method and contains two subsections. The first subsection is about Vehicle evaluation system of our algorithm and the second subsection is about genetic algorithm. Section 4 provides a discussion about how we simulate and test the performance of our algorithm in a synthetic environment. Finally section 5 concludes the paper.

## 2 Related Work

In this section we briefly describe GeoDTN+Nav [1], a hybrid geographic and DTN routing with navigation assistance in urban vehicular networks, which has some relations with our proposed method. This routing protocol contains three different modes (Greedy mode, Perimeter mode and DTN mode) for forwarding the packets. A network partition detection method has been used to switch between these modes

with the aim of increasing the packet delivery ratio even in sparse or partitioned networks.

The main idea of this routing protocol is using a VNI (Virtual Navigation Interface) framework as a parameter to determine proper delay tolerant forwarder. The goal of VNI is to standardize the content and transmission format of navigation information provided by GPS devices. For example, road identification can differ from one navigation system to another. Also the map encoding of a road on one navigation system may define a road as one separated by junctions; whereas, the map encoding of a road on another navigation system may define a road naturally from the name of the road. Based on vehicles movement pattern, vehicles are categorized into four different categories as shown in table below:

**Table 1.** Categories of vehicular route pattern

| Categories | Examples |
|---|---|
| Deterministic (fixed) route | Metro Bus, Metro Train, Campus Shuttle |
| Deterministic (Fixed) Destination | Taxi, Van Pool |
| Probabilistic (Expected) Route/Destination | Navigation system guided vehicles |
| Unknown | Random movement |

By considering these categories, VNI provides two kinds of primitive information:

1- Route_info: It may consist of detailed path, destination, or the direction of the vehicle.

2- Confidence: Confidence indicates the probability that the vehicles movement would abide by the given route information.

For example VNI on taxis would broadcast *(Dest/100%)* because taxis move deterministically toward its destination. In addition to VNI, GeoDTN+Nav uses other parameters (e.g. disconnection probability, direction of the node) to indicate whether a node is proper enough to be a DTN node.

## 3 Proposed Method

Similar to other types of position-based routing protocols, we used geographical position information of the nodes for forwarding the packets. Traditionally there exist two modes for forwarding the packets toward the destination node. First one is Greedy forwarding mode in which a node receiving a packet will send it to an immediate neighbor that is geographically closer to the destination. This process will

continue until the packet traps in a "local maximum" node where there exists no neighbor closer to the destination than itself.

Perimeter forwarding as the second mode is then considered to recover from local maximum by using right hand rule. GPSR and GPCR are two routing protocols which are using these two modes for forwarding the packets toward destination. The problem arises in DTN networks where the network is sparse. GPSR and GPCR are only applicable in networks with high density of nodes where there always exists a path from source to destination. Therefore for partitioned networks we need to add another mode to make connection between partitions. This mode is called DTN forwarding and the idea behind this mode is to choose a proper node for storing the packet and carrying it from one partition to another.

In GeoDTN+NAV routing protocol the proper DTN node and the switching time from perimeter mode to DTN mode are calculated using some parameters discussed in previous section. Although compared with GPSR and GPCR this routing protocol improves the packet delivery ratio of the DTN networks, it still has some limitations. The first limitation is all the vehicles have to be equipped with a VNI system. As it is mentioned in the previous section this VNI system broadcasts information about the path or destination of vehicles which may causes some privacy problems. For example it is not wise that law enforcement, military, or financial armored vehicles broadcast such these information to all their neighbors. Another problem is that some other effective parameters such as the speed of the candidate vehicles or the traffic model of the streets are not considered for the DTN node score formula.

Our routing strategy is somehow the expletive of GeoDTN+Nav routing protocol. The differences are first, in our proposed method we tried to consider other effective parameters to choose better DTN node and second, we used genetic algorithm to train our DTN node evaluation system and to determine how important each parameter is in the simulation environment.

### 3.1 DTN Node Evaluation System

As it is mentioned above, both greedy mode and perimeter mode will fail to forward packets to destination in the sparse and partitioned networks. So we need to choose a node as DTN node to store the packet and to carry it from one partition to another. Another thing that we should determine is when to switch from perimeter mode to DTN mode. As it is discussed in [1], we will switch from perimeter mode to DTN mode when the network is disconnected and a proper DTN node is detected. Several parameters exist that affect the switching decision. The speed of the vehicle, the direction of the vehicle, the distance between that vehicle to destination and the probability of network disconnectivity are examples of these parameters. Similar to "GeoDTN+Nav", we used hop count as a factor to recognize network disconnectivity. Combined these parameters with each other, we derive the bellow formula. The result of this formula is then compared with a threshold.

$$\alpha S(N_i) + \beta D(N_i) + \gamma F(N_i) + \delta P(h) > \text{Thresh} \quad (1)$$

where:

$S(N_i)$ : Speed of the $i^{th}$ neighbor

$D(N_i)$ : Direction of $i^{th}$ neighbor

$F(N_i)$ : Distance between $N_i$ and destination

$P(h)$ : Probability of network disconnectivity

$h$ : Hop counts that the packet has traversed in perimeter mode

α, β, γ, δ : System parameters

$N_i$ : $i^{th}$ neighbor of current node

Function $S(N_i)$ represents how suitable a neighbor is to be a DTN node according to its speed. The faster the node the better for choosing it as DTN node. So this function assigns a greater number to the vehicle with higher speed and lower number to the slower vehicle. The next parameter is the direction of the node. It is obvious that a node which is moving toward destination is a better choice to be DTN node than the node which moving away from destination. Therefore according to the angle formed with movement direction line of the node and the line between that node and destination, $D(N_i)$ will assign a number to each node to show how proper that node is to be DTN node according to the moving direction point of view. The next parameter is $F(N_i)$ that assign a number to a node based on the distance between that node to the destination. The last one is $P(h)$ that calculates the probability of network disconnection based on the hop count the packet traverse in perimeter mode. It is recommended to review [1] for better understanding of how these functions return a value for each node.

Till now the calculation steps were very similar to GeoDTN+Nav routing protocol. We just changed some parameters of the evaluation function to choose the better node for storing and carrying the packet. In the next section we start discussing our main idea that improves the node evaluation system and simultaneously improves the DTN node selection method.

### 3.2 Genetic Algorithm

Genetic algorithm (GA) is a subclass of evolutionary algorithms (EA) which generate solutions to optimization problems using techniques inspired by natural evolution such as selection, crossover, mutation and inheritance. Four steps have to be considered to solve a problem with GA. The first step is initialization in which the evolution usually starts from a population of randomly generated individuals. Each individual or chromosome is a set of genomes. In each generation, we need to evaluate the chromosomes by a fitness function and determine how suitable each of them is to be chosen for the next generation. The population of subsequent generations is formed by selecting and modifying the proper prior chromosomes.

These modifications are based on crossover and mutation. Crossover is a method to combine two chromosomes to produce new offspring. The idea behind crossover is that the new chromosome may be better than both of the parents if it takes the best characteristics from each of the parents. In mutation we alter one or more genome values in a particular chromosome. Mutation is done in order to prevent the population from stagnating at any local optima. The procedure of breeding offspring from previous generation will continue until a termination condition has been reached. This condition may be either finding a solution that satisfies minimum criteria or reaching a fixed number of generations.

In the following subsections we explain how we used the genetic algorithm for our purpose to improve the simulated network parameters.

### 3.2.1 Initialization

In this step we need to construct the first generation of the chromosomes. In our case of study each chromosome contains five genomes. As shown in figure 1, we put the coefficients of the proposed parameters in the chromosome. Also we considered the threshold of the evaluation formula as the fifth genome.

| α | β | γ | δ | Thresh |
|---|---|---|---|---|

**Fig. 1.** Chromosome Structure

According to this fact that we have no idea about the proper values of these genomes at the beginning of the Simulation, many individual solutions are randomly generated to form an initial population. As you will see in the simulation & result section, the value of each coefficient varies from 0 to 1 that causes the value of evaluation formula to vary in the range of 0 to 4.

### 3.2.2 Selection

After forming the first generation, we need to choose some chromosomes from current generation's population for inclusion in the next generation's population. This step of genetic algorithm is called selection. There exist different methods to select chromosomes from a particular population. Roulette-wheel selection [9], Tournament selection [10] and Top percent selection [8] are examples of these methods. In all these selection algorithms a fitness function is used to determine the optimality of a chromosome, so the particular chromosome can be ranked against all the other chromosomes. The fitness function that is considered in our algorithm is as follows:

$$\text{Fitness } (Ci) = \frac{1}{N}\sum_{j=1}^{N} \text{DeliveryDelay}\big(P(i,j)\big) \tag{2}$$

In this formula we have:

*Ci* : $i^{th}$ chromosome in the population

*N* : Number of packets generated in each generation

*P(i,j)* : $j^{th}$ packet generated in the simulation run of chromosome *Ci*

In the Roulette-wheel selection method which we also used for our simulation, the chance of a chromosome getting selected for the next generation is proportional to its fitness function value. This is why this method is also called Fitness Proportionate Selection. By using this type of selection the probability of individual ***i*** to be chosen for the next generation in the population is:

$$p(Ci) = \frac{Fitness(Ci)}{\sum_{k=1}^{M} Fitness(Ck)} \tag{3}$$

where:

*Ci* : the $i^{th}$ chromosome in the population

*Fitness(Ci)* : Fitness function for *Ci*

*M* : Number of chromosomes in each generation

Combining equation 2 with equation 3 we have:

$$p(Ci) = \frac{\sum_{j=1}^{N} DeliveryDelay(P(i,j))}{\sum_{k=1}^{M}\sum_{j=1}^{N} DeliveryDelay(P(k,j))} \tag{4}$$

As it is discussed in [9], with fitness proportionate selection there is a chance some weaker solutions may survive the selection process; this is an advantage, as though a solution may be weak, it may include some component which could prove useful following the reproduction process.

### 3.2.3 Reproduction

After selecting the proper chromosomes as the parents we need to breed new solutions from them for the next generation. The method that we used for breeding child solutions from a pair of parent solutions is One-point crossover. As it is shown in figure 2, a single crossover point on both parents' chromosome is selected. Then we interchange the two parent chromosomes at this point and produce two new offspring. After reproduction of new chromosome we need to apply mutation on some of them. Mutation should be applied to avoid the solutions from trapping in local optima. Mutation occurs during evolution according to a user-definable mutation probability. This probability should usually be set fairly low (about 0.01%) to prevent the algorithm to turn into a primitive random algorithm.

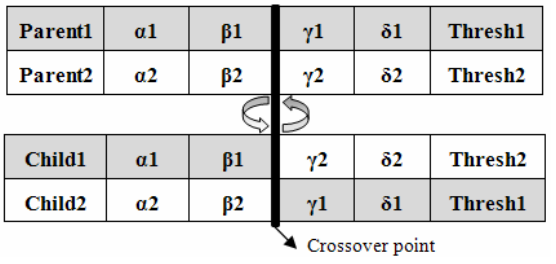



**Fig. 2.** An example of breeding two children from two parents

### 3.2.4 Termination

The genetic processes for breeding new generations of solutions discussed in the previous subsections will continue until a termination condition has been reached. It is because these processes are time consuming and they should be stoped after breeding particular generations. For example in our simulation which we will discuss in the next section the algorithm will be terminated after breeding four generation of solutions. Figure 3 illustrates the complete flowchart of discussed algorithm.

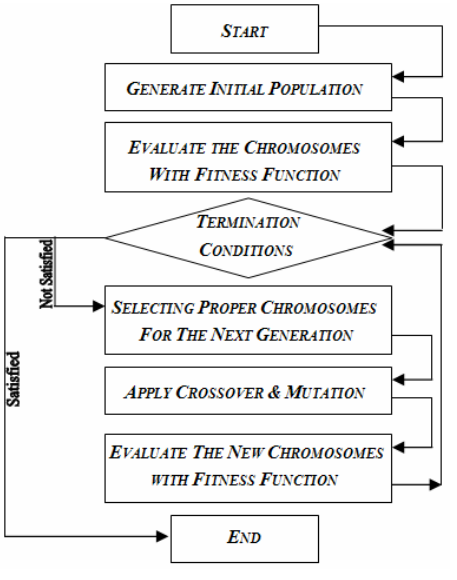



**Fig. 3.** Flowchart of genetic algorithm

In the following section we will discuss how we evaluate our proposed method by testing it in a synthetic network environment. Also by comparing the network parameters of each generation we show that this method will improve the average delivery delay and average delivery ratio of the network.

## 4 Simulation and Result

For evaluating our proposed method we implemented and tested it on a synthetic scenario. For our simulation we completed three steps including simulation of mobility, simulation of network and at last trace file analysing. Following we discuss each of these steps.

### 4.1 Simulation of Mobility

For the first step we needed to simulate our network environment including the streets, the junctions, the traffic lights and flows of mobile vehicles. To do so we used SUMO (**S**imulator of **U**rban **MO**bility) simulator which is an open source software. As it is shown in figure 4, our environment has two separate partitions of streets connected to each other with a highway. We considered this type of environment because we wanted to have a DTN network with disconnected partitions. Then some flows of vehicle are generated in a way that we had some streets with high density of vehicles and some with low. In Figure 4, the chromatic lines are the streets with high density of vehicles. After that we specified a node in the left partition to be the source or generator of the packets and a node in the right partition to be the destination of the packets.


**Fig. 4.** The topology of the simulated environment

Obstacles are placed between different road segments if they do not share the same horizontal or vertical parameters. All the streets are bidirectional and the length of each edge is 300m, and the transmission range is 300m too. The road between two partitions is 10km and contains some flows of vehicles with different speeds. All the vehicle flows in this road departure uniformly with speeds vary from 70km/h to 120km/h. The average vehicle speed in the two partitions is 40km/h.

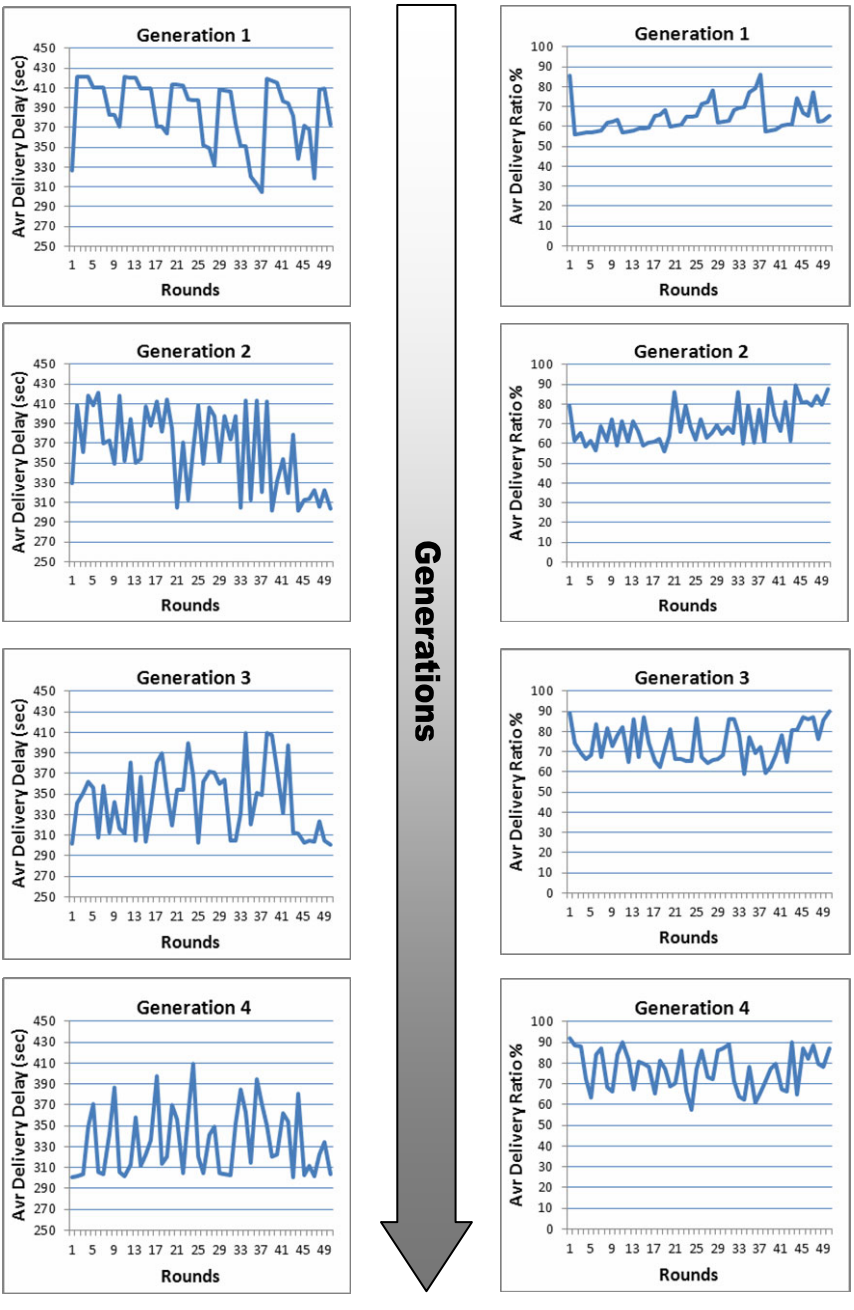



**Fig. 5.** Average Delivery Delays at left side and Average Delivery Ratios at right side for four generation

### 4.2 Simulation of Network

After simulating our streets and mobile vehicles we selected a source node in the left partition to generate packets and send them toward the destination and also selected a destination node in the right partition as a sink to receive the packets. Table 2 indicates the network parameters we used for our simulation. We attached a UDP agent to the source node to generate packets with constant bit rate and with packet size of 1460 bytes. All the related works for simulating the network were done with NS2.

**Table 2.** Simulation parameters

| Parameter | Value |
|---|---|
| Network Simulator | NS2 (Network Simulator 2) |
| Mobility Simulator | SUMO 0.12.0 |
| Packet Size | 1460 bytes |
| Transmission Range | 300 m |
| Simulation Runs | 200 |
| Average vehicle speed in the partitions | 40 km/h |
| Number of Generations | 4 generations |
| Number of Rounds in Each Generation | 50 rounds |

### 4.3 Trace File Analyzing

The third step is analyzing the trace files which are provided by NS2. In our case of study we needed to run the simulation for four generation and each generation contained fifty rounds; so the total number of runs was 200. For each run, NS2 provided us a trace file that contained all the information about the packets in the network. The total size of these files was about 1GB and we derived all the required information such as packet delivery delay and packet delivery ratio from them.

As it is illustrated in figure 5, we computed the average delivery delay of each round in each generation. Note that each round runs with particular quintuple of genomes that determined with genetic algorithm. To compare these four generations with each other and see how the genetic algorithm improves the network parameters in them we calculate the average of average delivery delays and the average of average delivery ratios for each generation. Regarding to figure 6, by breeding each new generation the delivery delay parameter of the network in our routing protocol decreased. Also it is concluded that the more we continue breeding the generations the more the quintuple parameters converge to the optimum values and the much delivery delay decreases. Meanwhile the average of average delivery delays for "GeoDTN+Nav" routing protocol remains in the range of [366,372] seconds.

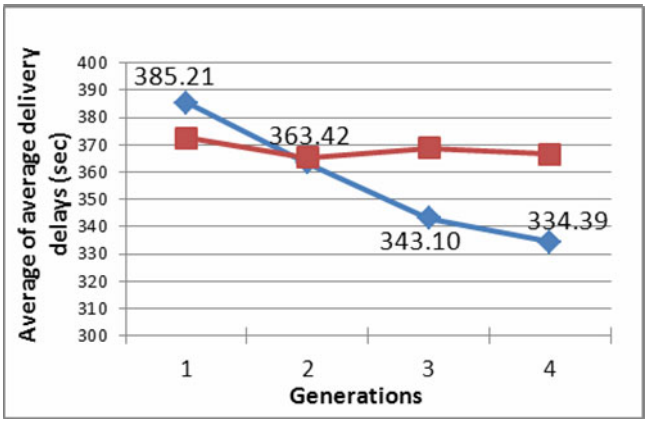



**Fig. 6.** Average of average delivery delays in four generation

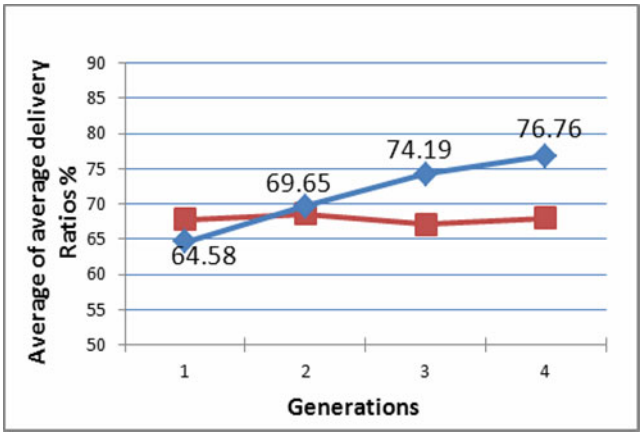



**Fig. 7.** Average of average delivery ratios in four generation

Also as illustrated in Figure 7, by breeding new generations the delivery ration parameter of the network in our routing protocol improved and increased. Again this parameter remains in a constant range for the "GeoDTN+Nav" routing protocol. As it is stated before we have to terminate the genetic algorithm and process of breeding new generations because it is time consuming. But still we can get a good perspective about the optimum value range of each genome.

For example in our simulated network the range of the coefficient of speed parameter ($\alpha$) after generation four bounded into [0.6,0.8] which means that the speed parameter in our evaluation formula is very effective. Also we can conclude that for having better delivery delay or delivery ratio in our network we should choose the value of "$\alpha$" parameter in the range [0.6,0.8]. We can find the optimum boundary of other coefficients of the parameters in the same way. Also the proposed algorithm is flexible to test the effectiveness of other parameters. For example if anyone thinks

that there exist other parameter that is effective in the evaluation formula he can add it to the formula with new coefficient and also add a new genome to each chromosome and run the algorithm again and see how that parameter affect the results. Then he can conclude weather that parameter is proper and necessary for the evaluation formula or not.

## 5 Conclusion

In this paper we proposed an intelligent routing protocol for delay tolerant networks. The routing starts with two traditional steps of other geographic routing protocols which are greedy forwarding and then perimeter forwarding. These two steps would fail to forward the packets toward destination in the case that the network is sparse and disconnect. This is where we brought the third mode of routing into account. The third mode was DTN forwarding in which we had to select a node as the DTN node to store the packet and carry it from one partition to another.

The idea presented in this paper was using genetic algorithm to determine how effective each parameter is in choosing the DTN node. This issue was done by setting the efficient of each parameter as the genomes of chromosomes and then performing the genetic algorithm. For our simulated network we considered five parameters which we thought are effective in DTN node evaluation formula. By breeding each generation the values of these coefficients converged to the optimum values.

## References

1. Cheng, P., Lee, K., Gerla, M., Harri, L.: GeoDTN+Nav, A Hybrid Geographic and DTN Routing with Navigation Assistant in Urban Vehicular Networks. In: ICVCS (2008)
2. Lee, K., Cheng, P., Weng, J., Tung, L., Gerla, M.: VCLCR, A Practical Geographic Routing Protocol in Urban Scenarios. In: Technical Report 080009, UCLA (2009)
3. Leontiadis, I., Mascolo, C.: GeOpps, Geographical Opportunistic Routing For Vehicular Networks. In: World of Wireless, Mobile and Multimedia Networks. IEEE, Los Alamitos (2007)
4. Lochert, C., Mauve, M., Fussler, H., Hartenstein, H.: Geographic Routing in City Scenarios. SIGMOBILE Mob. Comput. Commun. Rev. (2005)
5. Lee, K., Lee, U., Gerla, M.: Survey of Routing Protocols in Vehicular Ad hoc Networks. IGI Global (2009)
6. Bernsen, J., Manivannan, D.: Unicast Routing Protocols for Vehicular Ad hoc Networks, A Critical Comparison and Classification. Pervasive and Mobile Computing (2008)
7. Karp, B., Kung, H.: GPSR, Greedy Perimeter Stateless Routing for Wireless Networks. Mobile Computing and Networking, 243–254 (2000)
8. Haupt, R., Haupt, S.: Practical Genetic Algorithms, 2nd edn. Wiley, Chichester (2004)
9. Thomas, B.: Evolutionary Algorithms in Theory and Practice, Evolution Strategies, Evolutionary Programming, Genetic Algorithms. Oxford University, USA (1996)
10. Matousec, R.: Genetic Algorithm and Advanced Tournament Selection Concept. In: NICSO (2008)
11. LeBurn, J., Chuah, C., Ghosal, D., Zhang, M.: Knowledge Based Opportunistic Forwarding in Vehicular Wireless Ad hoc Networks. In: Vehicular Technology Conference. IEEE, Los Alamitos (2005)

# Analysis of the Distribution of the Statistic of a Test for Discriminating Correlated Processes

M.E. Sousa-Vieira

Department of Telematics Engineering, University of Vigo, Spain
estela@det.uvigo.es

**Abstract.** In this paper, we analyze the distribution of the statistic of a test for identifying the type of correlated time series. The rule for selecting a model suitable to the data is based on the comparison between the normalized prediction errors of the Whittle estimator applied to the candidate models. We consider one application of the test: assessing the significance of increasing the number of parameters within a given class of models. The results obtained demonstrate that the Weibull distribution is a good approximation for the distribution of the test statistic.

**Keywords:** Correlated processes, Whittle estimator, Model selection, Traffic modeling.

## 1 Introduction

Identifying the type of serial dependence of a sequence has long been of primal interest in time series analysis. That dependence may be characterized by the autocorrelation function, or equivalently, by the spectral density function in the frequency domain. If the autocorrelation function is summable, the sequence has Short-Range Dependence (SRD) and conversely, if the autocorrelation function is not summable, the sequence has Long-Range Dependence (LRD). Mathematically, LRD is closely related to the concept of self-similarity [4].

We are interested on the subject of network traffic modeling and simulation. The existence of persistent correlations at large time scales was established more than a decade ago after analyzing measurements of aggregated traffic [14,13,2,19,5]. Either at the fundamental transport or at the applications levels, persistent correlations may have a drastic impact on network performance [9,17,15,8,11] and models accurate as well as realistic are basic for network design and performance evaluation. Aside from examples in communication networks traffic, persistent correlations are also of interest in other domains, like network topologies for ad hoc wireless networks, meteorological and hydrological data, stock markets data or biometric signals [18]. In all these contexts, determining the form of the autocorrelation function as closely as possible is fundamental in order to build accurate and concise statistical models.

There are several families of Gaussian stochastic processes able to exhibit persistent correlations in a parsimonious way that could fit the empirical data equally well, such as fractional Gaussian noise (fGn) or fractional autoregressive

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 348–359, 2011.

integrated moving average (F-ARIMA) [1,16], and other approximate Gaussian long-range dependent (LRD) processes for that purpose, like M/G/$\infty$ [3]. The inference problem addressed in this article is the experimental analysis of the distribution of the statistic of a test proposed in a previous work [23] to check whether a sequence is statistically consistent with a given LRD process. The statistic used is a (normalized) comparison between the estimated prediction error when the standard Whittle estimator is applied over the sequence. As is well-known, the Whittle estimator is an unbiased, asymptotically Gaussian estimator of the long-memory parameter of a Gaussian process [26].

The remainder of the paper is organized as follows. We begin rewieving the main concepts related to the Whittle estimator in Section 2. In Section 3 we review the definition and properties of some LRD stochastic processes. In Section 4 we remember the test and analyze the distribution of the statistic. Finally, concluding remarks are given in Section 5.

## 2 Whittle Estimator

Let $f_\theta(\lambda)$ be the spectral density function of a zero-mean stationary Gaussian stochastic process, $X \triangleq \{X_n; n = 0, 1, \dots\}$, where $\theta = (\theta_1, \dots, \theta_M)$ is a vector of unknown parameters that is to be estimated from observations. Let

$$I_{X^N} = \frac{1}{2\pi N} \| \sum_{i=0}^{N-1} X_i \mathrm{e}^{-\jmath\lambda i} \|^2$$

be the periodogram of a sample of size $N$ of the process $X$. The approximate Whittle estimator [26] is the vector $\widehat{\theta} = (\hat{\theta}_1, \dots, \hat{\theta}_M)$ minimizing, for a given sample $X^N$ of size $N$ of $X$, the statistic

$$Q_{X^N}(\theta) \triangleq \frac{1}{2\pi} \left( \int_{-\pi}^{\pi} \frac{I_{X^N}(\lambda)}{f_\theta(\lambda)} \mathrm{d}\lambda + \int_{-\pi}^{\pi} \log f_\theta(\lambda) \mathrm{d}\lambda \right). \tag{1}$$

If $\theta^{\mathrm{o}}$ is the true value of $\theta$, then $\lim_{N\to\infty} \Pr\big(\|\hat{\theta} - \theta^{\mathrm{o}}\| < \epsilon\big) = 1$, for any $\epsilon > 0$, namely, $\hat{\theta}$ converges in probability to $\theta^{\mathrm{o}}$ (is a weakly consistent estimator). It is also asymptotically normal, since $\sqrt{N}(\hat{\theta} - \theta^{\mathrm{o}})$ converges in distribution to $\zeta$, as $N \to \infty$, where $\zeta$ is a zero-mean Gaussian vector with matrix of covariances known. Thus, from this asymptotic normality, confidence intervals of the estimated values can be computed.

A simplification of (1) arises by choosing a special scale parameter $\theta_1$, such that $f_\theta(\lambda) = \theta_1 f_{\theta^*}(\lambda) = \theta_1 f^*_\eta(\lambda)$ and

$$\int_{-\pi}^{\pi} \log f^*_\theta(\lambda) \mathrm{d}\lambda = \int_{-\pi}^{\pi} \log f^*_\eta(\lambda) \mathrm{d}\lambda = 0,$$

where $\eta = (\theta_1, \dots, \theta_M)$ and $\theta^* = (1, \eta)$. Therefore

$$\theta_1 = \exp\left( \frac{1}{2\pi} \int_{-\pi}^{\pi} \log f_\theta(\lambda) \mathrm{d}\lambda \right) = \frac{\sigma_\epsilon^2}{2\pi},$$

where $\sigma_\epsilon^2$ is the optimal one-step-ahead prediction error, that is equal to the variance of the innovations in the AR($\infty$) representation of the process [1], $X_i = \sum_{j=1}^{\infty} \beta_j X_{i-j} + \epsilon_i$.

Using this normalization, equation (1) simplifies to

$$Q_{X^N}(\theta^*) = Q^*_{X^N}(\eta) = \int_{-\pi}^{\pi} \frac{I_{X^N}(\lambda)}{f_{\theta^*}(\lambda)} \mathrm{d}\lambda = \int_{-\pi}^{\pi} \frac{I_{X^N}(\lambda)}{f^*_{\eta}(\lambda)} \mathrm{d}\lambda$$

which is usually evaluated numerically via integral quadrature. In addition, it can be shown [1] that the estimated prediction error is given by $\widehat{\sigma_\epsilon^2} = Q^*_{X^N}(\hat{\eta})$.

## 3 LRD Processes

Following, we review briefly the definition and properties of the long-memory processes that we consider in this work.

### 3.1 The fGn Process

The fGn process is the sequence of increments of the fractional Brownian motion (fBm) process. Its autocorrelation function and its spectral density are given, respectively, by

$$r_{\mathsf{H}}[k] = \frac{1}{2}\Big(|k+1|^{2\mathsf{H}} - 2|k|^{2\mathsf{H}} + |k-1|^{2\mathsf{H}}\Big) \quad \forall k,$$

$$f_{\mathsf{H}}(\lambda) = (1-\cos\lambda)\sin(\pi\mathsf{H})\Gamma(2\mathsf{H}+1)\sum_{i=-\infty}^{+\infty} \|2\pi i + \lambda\|^{-2\mathsf{H}-1}$$

for $\lambda \in [-\pi, \pi]$. In these formulas, $\mathsf{H}$ is the Hurst parameter [10], a measure of the persistence of correlations, and $\Gamma(\cdot)$ denotes the Gamma function. For any value of $\mathsf{H}$,

$$f_{\mathsf{H}}(\lambda) = c\big(|\lambda|^{-2\mathsf{H}+1} + B(\lambda, \mathsf{H})\big)$$

with

$$B(\lambda, \mathsf{H}) = \sum_{i=1}^{\infty} \|\lambda + 2\pi i\|^{-2\mathsf{H}-1} + \|\lambda - 2\pi i\|^{-2\mathsf{H}-1},$$

so $f_{\mathsf{H}}(\lambda) \sim c|\lambda|^{1-2\mathsf{H}}$ when $|\lambda| \to 0$, and the fGn process exhibits statistical long-range dependence when $\mathsf{H} \in (0.5, 1)$.

### 3.2 The F-ARIMA Processes

The class of F-ARIMA processes is a generalization of ARIMA processes. A F-ARIMA$(p, d, q)$ process $X = \{X_n; n = 0, 1, \dots\}$ satisfies the formal equation

$$\phi_p(B)(1-B)^d X_n = \phi_q(B)\epsilon_n, \quad n \geq 0$$

where $B$ is the backshift or delay operator ($B^j X_n = X_{n-j}$), $\phi_p(B)$ is a polynomial of degree $p$ in $B$, $\phi_q(B)$ is a polynomial of degree $q$ in $B$, $d$ is a real value and $\{\epsilon_n\}_{n\geq 0}$ is a renewal process with zero mean and finite variance $\sigma_\epsilon^2$.

Although it is not generally feasible to obtain the autocorrelation function for a general F-ARIMA$(p,d,q)$ process, for the subclass of F-ARIMA$(0,d,0)$ processes it takes the form

$$r_d[k] = \prod_{i=1}^{k} \frac{i+d-1}{i-d} \quad \forall k.$$

It is important to highlight that, for large lags, the correlation structure of F-ARIMA$(0,d,0)$ is very similar to that of any F-ARIMA$(p,d,q)$ process. In other words, the long-term correlation structure is mainly driven by the fractional integration exponent $(1-B)^d$, and $d$ is the long-memory parameter of the process.

The spectral density of F-ARIMA$(0,d,0)$ is known to be

$$f_d(\lambda) = \frac{\sigma_\epsilon^2}{2\pi} \|1 - \mathrm{e}^{\jmath\lambda}\|^{-2d} = \frac{\sigma_\epsilon^2}{2\pi} \left(2\sin^2 \frac{\lambda}{2}\right)^{-d} \quad \forall \lambda \in [-\pi, \pi].$$

And for the particular case of a F-ARIMA$(1,d,0)$ process

$$f_{\alpha_1,d}(\lambda) = \frac{f_d(\lambda)}{\|1 - \alpha_1 \mathrm{e}^{\jmath\lambda}\|^2} \quad \forall \lambda \in [-\pi, \pi],$$

with $\alpha_1$ the coefficient of the AR(1) filter.

It is easy to see that $f_d(\lambda) \sim c|\lambda|^{-2d}$ if $|\lambda| \to 0$. Therefore, when $d \in (0, 1/2)$ the F-ARIMA processes have long-range dependence with Hurst parameter $\mathsf{H} = d + 1/2$.

### 3.3 The M/G/$\infty$ Process

The M/G/$\infty$ process is a stationary version of the occupancy process of an M/G/$\infty$ queueing system. In this queueing system, customers arrive according to a Poisson process, occupy a server for a random time with a generic distribution $Y$ with finite mean, and leave the system.

Though the system operates in continuous time, it is easier to adopt a discrete-time description, so this will be the convention henceforth [24]. The number of busy servers at instant $n \in \mathbb{Z}^+$ is $X_n = \sum_{i=1}^{\infty} A_{n,i}$, where $A_{n,i}$ is the number of arrivals at instant $n-i$ which remain active at instant $n$, i.e., the number of active customers with age $i$. For any fixed $n$, $\{A_{n,i}, i = 1, \dots\}$ are a sequence of independent and identically distributed (iid) Poisson variables with parameter $\lambda \mathbb{P}(Y \geq i)$, where $\lambda$ is the rate of the arrival process. The expectation and variance of the number of servers occupied at instant $n$ is $\mathbb{E}(X_n) = \mathrm{Var}(X_n) = \lambda \sum_{i=1}^{\infty} \mathbb{P}(Y \geq i) = \lambda \mathbb{E}(Y)$.

The discrete-time process $X_n, n = 0, 1, \dots$ is time-reversible and wide-sense stationary, with autocovariance function

$$\gamma[k] = \mathrm{Cov}(X_{n+k}, X_n) = \lambda \sum_{i=k+1}^{\infty} \mathbb{P}(Y \geq i), \quad k = 0, 1, \dots$$

The function $\gamma[k]$ determines completely the expected service time $\mathbb{E}(Y) = \frac{\gamma[0]}{\lambda}$, and the distribution of $Y$, the service time

$$\mathbb{P}(Y = i) = \frac{\gamma[i-1] - 2\gamma[i] + \gamma[i+1]}{\lambda}, \quad i = 1, 2, \ldots. \tag{2}$$

By (2), the autocovariance is a non-negative convex function. Alternatively, any real-valued sequence $\gamma[k]$ can be the autocovariance function of a discrete-time M/G/$\infty$ occupancy process if and only if it is decreasing, non-negative and integer-convex [12]. In such a case, $\lim_{k\to\infty} \gamma[k] = 0$ and the probability mass function of $Y$ is given by (2).

If $A_{0,0}$, i.e. the initial number of customers in the system, follows a Poisson distribution with mean $\lambda\mathbb{E}(Y)$, and their service times have the same distribution as the residual life $\widehat{Y}$ of the random variable $Y$

$$\mathbb{P}\big(\widehat{Y} = i\big) = \frac{\mathbb{P}(Y \geq i)}{\mathbb{E}(Y)},$$

then $\{X_n, n = 0, 1, \ldots\}$ is strict-sense stationary, ergodic, and enjoys the following properties:

1. The marginal distribution of $X_n$ is Poissonian for all $n$, with mean value $\mu = \mathbb{E}(X_n) = \lambda\mathbb{E}(Y)$.
2. The autocovariance function is $\gamma[h] = \gamma[0]\mathbb{P}\big(\widehat{Y} > h\big)$ for any $h \in \mathbb{Z}^+$.

In particular, the M/G/$\infty$ process exhibits LRD when $Y$ has infinite variance, as in the case of some heavy-tailed distributions. These are the discrete probability distribution functions satisfying $\mathbb{P}(Y > i) \sim i^{-\alpha}$ asymptotically as $i \to \infty$.

We are interested on the M/G/$\infty$ process due to its theoretical simplicity (there exists a substantial body of research results about the system's behavior [6,25,20,7]), its flexibility to exhibit LRD in a parsimonious way and its advantages for simulation studies [12,21], such as the possibility of on-line generation (the main drawback of fGn and F-ARIMA processes is that only off-line methods for trace generation are efficient enough to be of practical use) and the lower computational cost.

## 4 Test

We propose to use $\widehat{\sigma_\epsilon^2}$ as a measure of the suitability of a candidate model for a time series. Smaller values of $\widehat{\sigma_\epsilon^2}$ bear the meaning of more accurate numerical adjustment to the actual autocorrelation function of the series, and suggest a close match between the sample and the model.

In this work we consider the application of this similarity criterion to answer this question: within a fixed class of stochastic models, increasing the number of parameters (e.g., the memory order) gives more degrees of freedom and may result in a better fit of the empirical autocorrelation function. Consequently, the estimated prediction error would be lower. But is this improvement statistically

significant so as to compensate for the sophistication of the model? If the fitness only improves marginally, adding more parameters just complicates the model without giving better insight into the relevant features of the process.

To solve this question, we proposed [23] a comparison over the spectral density to aid in the selection of the best model. Following, we remember briefly the methodology.

Consider two possible processes, labeled 0 and 1. Set out the base model $\mathcal{M}_0 : f(\lambda) = f_0(\lambda)$, where $f_0(\lambda)$ is the spectral density of process 0, as opposed to the alternative model $\mathcal{M}_1 : f(\lambda) = f_1(\lambda)$, that the spectral density is best described by other type of process, with spectral density $f_1(\lambda)$ for instance. We define the test statistic $t$ as the difference between the estimated prediction error when the base model $\mathcal{M}_0$ is assumed and the estimated prediction error when the alternative model $\mathcal{M}_1$ is regarded true, normalized by the sample variance

$$t \triangleq \frac{\widehat{\sigma_\epsilon^2}(\mathcal{M}_0) - \widehat{\sigma_\epsilon^2}(\mathcal{M}_1)}{\widehat{\sigma^2}}.$$

We use the comparison test in the classical way, choosing the simpler of the two as process 0. The degree of significance $\epsilon$ is fixed beforehand, and its critical region $\mathcal{R}$ evaluated. Once $\mathcal{R}$ is known, if the null hypothesis could not be rejected we should conclude that the apparently better fit of the model with more parameters is not significant, and the simpler one must be preferred.

In this paper, we study the distribution of the statistic $t$ when applied to this inference problem.

### 4.1 F-ARIMA

Consider a setting consisting in the comparison of two processes in the F-ARIMA class in the first place. The simplest case consists of adding a single AR(1) term to a pure fractionally integrated process. This amounts to selecting between F-ARIMA$(0, d, 0)$ and F-ARIMA$(1, d, 0)$ for different choices of the one-lag autocorrelation coefficient $\alpha_1$.

Fix this time as base model $\mathcal{M}_0 : f_0(\lambda) = f_{\widehat{d_0}}(\lambda)$, and as second model $\mathcal{M}_1 : f_1(\lambda) = f_{\widehat{\alpha_1},\widehat{d_1}}(\lambda)$. In other words, we seek to assess the goodness of fit between two spectral densities differing in their long-term correlation ($\widehat{d_0} \neq \widehat{d_1}$) and also in their short-term correlation structure ($\widehat{\alpha_1} \neq 0$). Each one of the parameters $\widehat{d_0}$, $\widehat{d_1}$ and $\widehat{\alpha_1}$ are inferred via the Whittle estimator, after postulating the supposed spectral density function.

We generated $10^3$ synthetic traces of varying lengths $n = \{2^{13}, 2^{14}, 2^{15}, 2^{16}\}$ of a F-ARIMA$(0, d, 0)$ process for $d = 0.1$, $d = 0.25$ and $d = 0.4$. With these data, $t$ was calculated, along with the estimated critical region $\mathcal{R}$ for a degree of significance $\epsilon = 5\%$. Table 1 contains the results. The lower boundary $r_0$ of $\mathcal{R}$ is the real number such that the 5%-quantile of the simulation runs attain $t$-values greater than $r_0$. Thus, note that $\mathcal{R}$ in that table is really a function of the empirical distribution function of $t$. Moreover, note that all the lower ends of $\mathcal{R}$ are positive and close to 0. Figs. 1 and 2 plot the distribution of the statistic.

**Table 1.** Estimated critical region: F-ARIMA$(0,d,0)$ vs. F-ARIMA$(1,d,0)$

| $n$ | $\mathsf{H}=0.6$ | $\mathsf{H}=0.75$ | $\mathsf{H}=0.9$ |
|---|---|---|---|
| $2^{13}$ | $(4.59\cdot10^{-4},\infty)$ | $(4.04\cdot10^{-4},\infty)$ | $(2.77\cdot10^{-4},\infty)$ |
| $2^{14}$ | $(2.32\cdot10^{-4},\infty)$ | $(2.02\cdot10^{-4},\infty)$ | $(1.32\cdot10^{-4},\infty)$ |
| $2^{15}$ | $(1.15\cdot10^{-4},\infty)$ | $(1.00\cdot10^{-4},\infty)$ | $(0.64\cdot10^{-4},\infty)$ |
| $2^{16}$ | $(0.58\cdot10^{-4},\infty)$ | $(0.50\cdot10^{-4},\infty)$ | $(0.32\cdot10^{-4},\infty)$ |

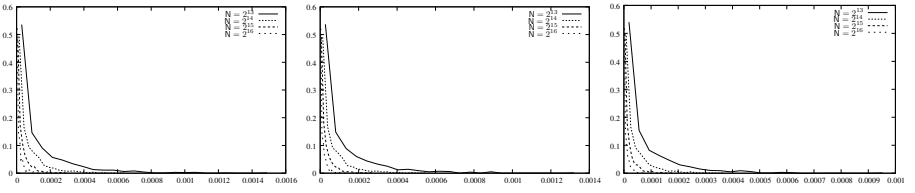


**Fig. 1.** Distribution of the statistic $t$: F-ARIMA$(0,d,0)$ vs. F-ARIMA$(1,d,0)$. $\mathsf{H}=0.6$ (top left), $\mathsf{H}=0.75$ (top right) and $\mathsf{H}=0.9$ (bottom)

In order to choose the distribution that gives rise to a better adjustement of the empirical samples of the statistic, we obtain the Q-Q graphs for different distributions (Exponential, Lognormal, Gamma and Weibull). In Fig. 3 we show the results obtained with the Weibull distribution, that in all cases is the one that gives rise to lower error.

Table 2 shows the MLE (Maximum Likelihood Estimation) parameters of the Weibull distribution ($\mu$,$\sigma$,$\alpha$,$\beta$). The results strongly suggest that $r_0$, $\mu$, $\sigma$ and $\beta$ are uniformly decreasing in $\mathsf{H}$, and the variation is approximately inverse to that of $n$. $\alpha$ lightly increases with $\mathsf{H}$ and $n$.

Finally, Fig. 4 shows the adjustment of the distribution of the statistic with the Weibull distribution.

### 4.2 M/G/$\infty$

The robustness of the Whittle estimator is satisfactory even for non-Gaussian processes. For that reason, it makes sense to study the behavior of the $t$ statistic in this case. So, in this Section we consider the M/G/$\infty$ process tuned so that its autocorrelation function belongs to the class of exactly second order self-similar processes and equals that of a fGn process

$$r_{\mathsf{H}}[k]=\frac{1}{2}\Big(|k+1|^{2\mathsf{H}}-2|k|^{2\mathsf{H}}+|k-1|^{2\mathsf{H}}\Big)\quad\forall k.$$

From (2) the probability mass function of the service times in an M/G/$\infty$ system whose occupancy process shows such correlation function is

$$\mathbb{P}(Y=i)=\frac{\sum_{j=i-2}^{i+2}a_j j^{2\mathsf{H}}}{2-2^{2\mathsf{H}-1}}\quad i=1,2,\ldots.$$

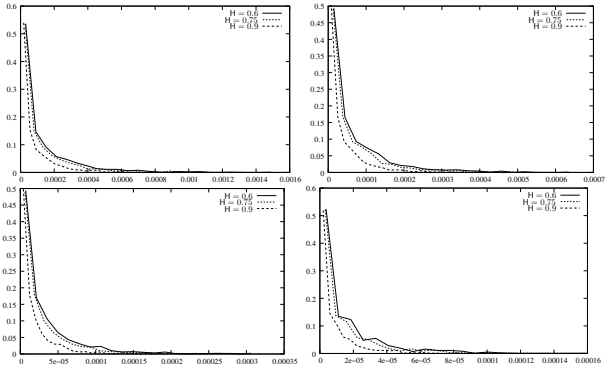



**Fig. 2.** Distribution of the statistic $t$: F-ARIMA$(0, d, 0)$ vs. F-ARIMA$(1, d, 0)$ $n = 2^{13}$ (top left), $n = 2^{14}$ (top right), $n = 2^{15}$ (bottom left) and $n = 2^{16}$ (bottom right)

**Table 2.** Adjustment of the distribution of the statistic $t$ with the Weibull distribution. MLE parameters ($\mu$,$\sigma$,$\alpha$,$\beta$): F-ARIMA$(0, d, 0)$ vs. F-ARIMA$(1, d, 0)$.

| $n$ | $\mu$ | $\sigma$ | $\alpha$ | $\beta$ |
|---|---|---|---|---|
| | | H = 0.6 | | |
| $2^{13}$ | $1.26 \cdot 10^{-4}$ | $1.81 \cdot 10^{-4}$ | $6.61 \cdot 10^{-1}$ | $9.46 \cdot 10^{-5}$ |
| $2^{14}$ | $6.35 \cdot 10^{-5}$ | $8.56 \cdot 10^{-5}$ | $6.91 \cdot 10^{-1}$ | $4.98 \cdot 10^{-5}$ |
| $2^{15}$ | $3.04 \cdot 10^{-5}$ | $4.24 \cdot 10^{-5}$ | $7.34 \cdot 10^{-1}$ | $2.48 \cdot 10^{-5}$ |
| $2^{16}$ | $1.55 \cdot 10^{-5}$ | $2.16 \cdot 10^{-5}$ | $7.61 \cdot 10^{-1}$ | $1.29 \cdot 10^{-5}$ |
| | | H = 0.75 | | |
| $2^{13}$ | $1.11 \cdot 10^{-4}$ | $1.58 \cdot 10^{-4}$ | $6.64 \cdot 10^{-1}$ | $8.28 \cdot 10^{-5}$ |
| $2^{14}$ | $5.53 \cdot 10^{-5}$ | $7.48 \cdot 10^{-5}$ | $6.97 \cdot 10^{-1}$ | $4.36 \cdot 10^{-5}$ |
| $2^{15}$ | $2.64 \cdot 10^{-5}$ | $3.68 \cdot 10^{-5}$ | $7.43 \cdot 10^{-1}$ | $2.18 \cdot 10^{-5}$ |
| $2^{16}$ | $1.34 \cdot 10^{-5}$ | $1.87 \cdot 10^{-5}$ | $7.71 \cdot 10^{-1}$ | $1.13 \cdot 10^{-5}$ |
| | | H = 0.9 | | |
| $2^{13}$ | $7.48 \cdot 10^{-5}$ | $1.07 \cdot 10^{-4}$ | $6.74 \cdot 10^{-1}$ | $5.68 \cdot 10^{-5}$ |
| $2^{14}$ | $3.68 \cdot 10^{-5}$ | $5.01 \cdot 10^{-5}$ | $7.12 \cdot 10^{-1}$ | $2.94 \cdot 10^{-5}$ |
| $2^{15}$ | $1.71 \cdot 10^{-5}$ | $2.39 \cdot 10^{-5}$ | $7.71 \cdot 10^{-1}$ | $1.45 \cdot 10^{-5}$ |
| $2^{16}$ | $8.68 \cdot 10^{-6}$ | $1.18 \cdot 10^{-5}$ | $8.21 \cdot 10^{-1}$ | $7.68 \cdot 10^{-6}$ |

with $a_2 = a_{-2} = 1/2$, $a_1 = a_{-1} = -2$ and $a_0 = 3$. In the following, denote this random variable by F, and the resulting process by M/F/$\infty$.

In order to improve the adjustment of the short-term correlation in a previous work we incorporated an autorregressive filter [22]. Therefore, if $X$ is the M/F/$\infty$ process, the new process is simply $Z_n = \alpha_1 Z_{n-1} + \ldots + \alpha_{\mathsf{p}} Z_{n-\mathsf{p}} + X_n$, for a given set of coefficients $\alpha_1, \ldots, \alpha_{\mathsf{p}}$.

For simplicity we will focus on the particular case of an AR(1) filter. In this case, the mean values and covariances are related by $\mathbb{E}(Z) = \frac{\mathbb{E}(X)}{1-\alpha_1}$, and

$$\gamma_Z[k] = \frac{1}{1-\alpha_1^2}\Big(\gamma_X[k] + \sum_{i=-\infty, i\neq 0}^{\infty} \gamma_X[k+i]\alpha_1^{|i|}\Big).$$

Denote the class of the resulting process as M/F/$\infty$-AR.

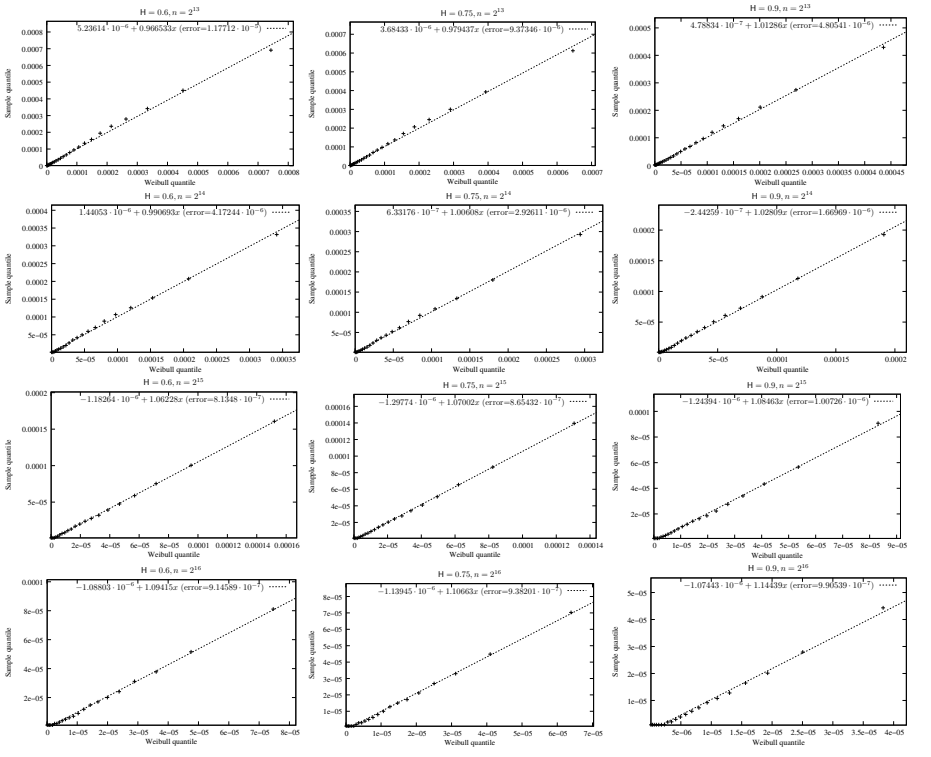



**Fig. 3.** Adjustment of the distribution of the statistic $t$ with the Weibull distribution. Q-Q graphs: F-ARIMA$(0, d, 0)$ vs. F-ARIMA$(1, d, 0)$

If $f_X(\lambda)$ is the spectral density of the process $X$, the spectral density of $Z$ is given by

$$f_Z(\lambda) = f_{\alpha_1,\mathsf{H}}(\lambda) = \frac{f_X(\lambda)}{||1 - \alpha_1 \mathrm{e}^{j\lambda}||^2} \quad \forall \lambda \in [-\pi, \pi].$$

Now, we consider as base model for the data $\mathcal{M}_0 : f(\lambda) = f_{\widehat{\mathsf{H}_0}}(\lambda)$, and as alternative possibility $\mathcal{M}_1 : f(\lambda) = f_{\widehat{\alpha_1},\widehat{\mathsf{H}_1}}(\lambda)$, where $\widehat{\mathsf{H}_0}$ is the value obtained for the Hurst parameter $\mathsf{H}$ using the Whittle estimator when the M/F/$\infty$ process is the base process, and $\widehat{\alpha_1}$ and $\widehat{\mathsf{H}_1}$ are the values obtained for the parameters $\alpha_1$ and $\mathsf{H}$ using the Whittle estimator and considering the M/F/$\infty$-AR process as base process.

We generated $10^3$ synthetic traces of varying lengths $n = \{2^{13}, 2^{14}, 2^{15}, 2^{16}\}$ of the M/F/$\infty$ process for $\mathsf{H} = 0.6$, $\mathsf{H} = 0.75$ and $\mathsf{H} = 0.9$. This time, the estimated critical region $\mathcal{R}$ ($\epsilon = 5\%$ degree of significance) is shown in Table 3. Again, $r_0$ is uniformly decreasing in $\mathsf{H}$ and in $n$. Figs. 5 and 6 plot the distribution of the statistic.

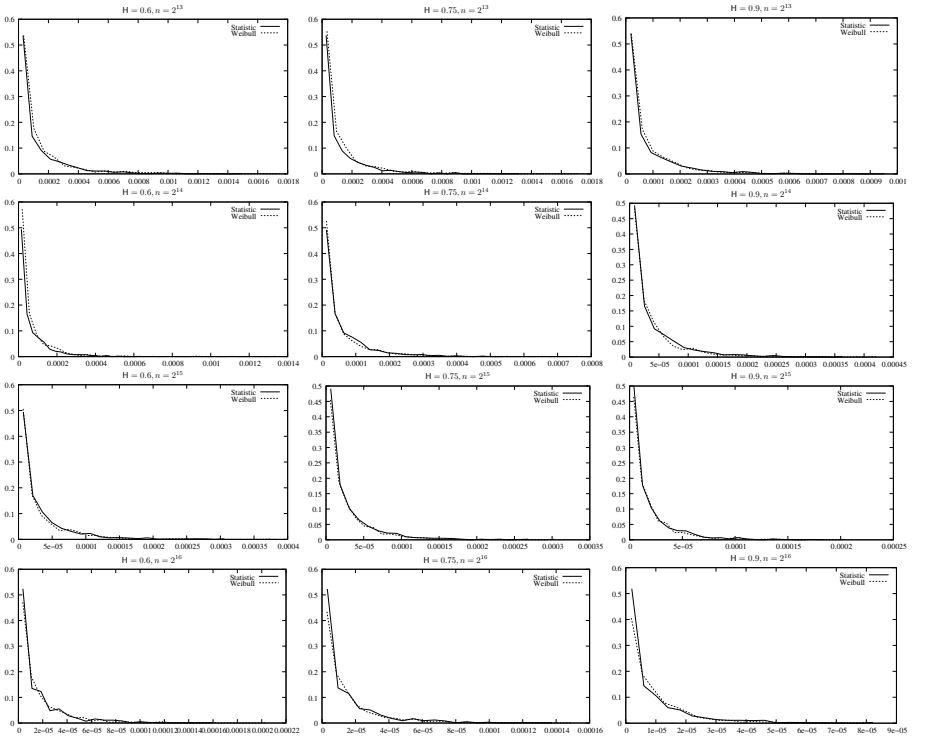



**Fig. 4.** Adjustment of the distribution of the statistic $t$ with the Weibull distribution: F-ARIMA$(0, d, 0)$ vs. F-ARIMA$(1, d, 0)$.

**Table 3.** Estimated critical region: M/F/$\infty$ vs. M/F/$\infty$-AR

| $n$ | $\mathsf{H} = 0.6$ | $\mathsf{H} = 0.75$ | $\mathsf{H} = 0.9$ |
|---|---|---|---|
| $2^{13}$ | $4.531 \cdot 10^{-4}$ | $4.216 \cdot 10^{-4}$ | $2.771 \cdot 10^{-4}$ |
| $2^{14}$ | $2.446 \cdot 10^{-4}$ | $1.956 \cdot 10^{-4}$ | $1.312 \cdot 10^{-4}$ |
| $2^{15}$ | $1.146 \cdot 10^{-4}$ | $0.944 \cdot 10^{-4}$ | $0.781 \cdot 10^{-4}$ |
| $2^{16}$ | $0.546 \cdot 10^{-4}$ | $0.477 \cdot 10^{-4}$ | $0.361 \cdot 10^{-4}$ |

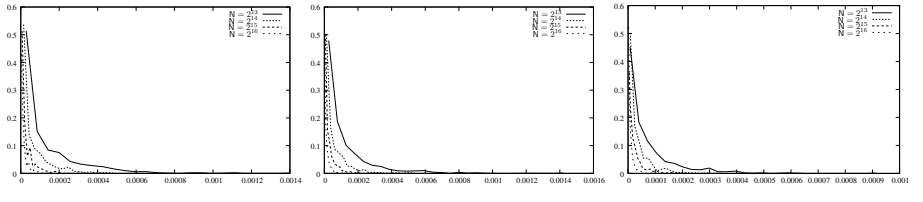


**Fig. 5.** Distribution of the statistic $t$: M/F/$\infty$ vs. M/F/$\infty$-AR. $\mathsf{H} = 0.6$ (top left), $\mathsf{H} = 0.75$ (top right) and $\mathsf{H} = 0.9$ (bottom)

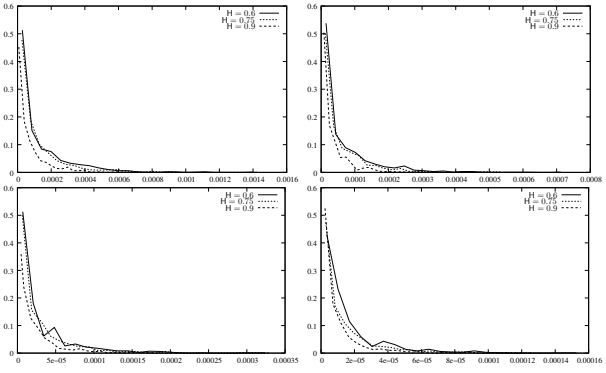



**Fig. 6.** Distribution of the statistic $t$: M/F/$\infty$ vs. M/F/$\infty$-AR. $n = 2^{13}$ (top left), $n = 2^{14}$ (top right), $n = 2^{15}$ (bottom left) and $n = 2^{16}$ (bottom right)

The discussion of the previous section about the adjustment of the distibution of the statistic is valid in this case. Again, the Weibull distribution is the one that gives rise to better results and the conclusions about the variation of its parameters are similar. We omit them for space restrictions.

## 5 Conclusions

We have analyzed a test about the spectral density of long-memory processes, where the statistic is the normalized difference between the prediction errors obtained via the Whittle estimator. We consider one application of the test: assessing the significance of increasing the number of parameters within a given class of models. The results obtained strongly suggest that the Weibull distribution is a good approximation of the distribution of the statistic. As further work, we will try to find a relationship between $\mathsf{H}$ and $n$ and the parameters of the Weibull distribution.

## References

1. Beran, J.: Statistics for Long-Memory Processes. Chapman and Hall, Boca Raton (1994)
2. Beran, J., Shreman, R., Taqqu, M.S., Willinger, W.: Long-Range Dependence in Variable-Bit-Rate video traffic. IEEE Transactions on Communications 43(2/4), 1566–1579 (1995)
3. Cox, D.R., Isham, V.: Point Processes. Chapman and Hall, Boca Raton (1980)
4. Cox, D.R.: Long-Range Dependence: A review. In: Statistics: An Appraisal, pp. 55–74. Iowa State University Press, Iowa (1984)
5. Crovella, M.E., Bestavros, A.: Self-similarity in World Wide Web traffic: Evidence and possible causes. IEEE/ACM Transactions on Networking 5(6), 835–846 (1997)
6. Duffield, N.: Queueing at large resources driven by long-tailed M/G/$\infty$ processes. Queueing Systems 28(1/3), 245–266 (1987)
7. Eliazar, I.: The M/G/$\infty$ system revisited: Finiteness, summability, long-range dependence and reverse engineering. Queueing Systems 55(1), 71–82 (2007)

8. Erramilli, A., Narayan, O., Willinger, W.: Experimental queueing analysis with Long-Range Dependent packet traffic. IEEE/ACM Transactions on Networking 4(2), 209–223 (1996)
9. Garrett, M.W., Willinger, W.: Analysis, modeling and generation of self-similar VBR video traffic. In: Proc. ACM SIGCOMM 1994, London, UK, pp. 269–280 (1994)
10. Hurst, H.E.: Long-term storage capacity of reservoirs. Transactions of the American Society of Civil Engineers 116, 770–799 (1951)
11. Jiang, M., Nikolic, M., Hardy, S., Trajkovic, L.: Impact of self-similarity on wireless data network performance. In: Proc. IEEE ICC 2001, Helsinki, Finland, pp. 477–481 (2001)
12. Krunz, M., Makowski, A.: Modeling video traffic using M/G/$\infty$ input processes: A compromise between Markovian and LRD models. IEEE Journal on Selected Areas in Communications 16(5), 733–748 (1998)
13. Leland, W.E., Taqqu, M.S., Willinger, W., Wilson, D.V.: On the self-similar nature of Ethernet traffic (extended version). IEEE/ACM Transactions on Networking 2(1), 1–15 (1994)
14. Li, S.Q., Hwang, C.L.: Queue response to input correlation functions: Discrete spectral analysis. IEEE/ACM Transactions on Networking 1(5), 317–329 (1993)
15. Likhanov, N., Tsybakov, B., Georganas, N.D.: Analysis of an ATM buffer with self-similar ("fractal") input traffic. In: Proc. IEEE INFOCOM 1995, Boston, MA, USA, pp. 985–992 (1995)
16. López, J.C., López, C., Suárez, A., Fernández, M., Rodríguez, R.F.: On the use of self-similar processes in network simulation. ACM Transactions on Modeling and Computer Simulation 10(2), 125–151 (2000)
17. Norros, I.: A storage model with self-similar input. Queueing Systems 16, 387–396 (1994)
18. Novak, M.: Thinking in patterns: Fractals and related phenomena in nature. World Scientific, Singapore
19. Paxson, V., Floyd, S.: Wide-area traffic: The failure of Poisson modeling. IEEE/ACM Transactions on Networking 3(3), 226–244 (1995)
20. Resnick, S., Rootzen, H.: Self-similar communication models and very heavy tails. Annals of Applied Probability 10(3), 753–778 (2000)
21. Sousa, M.E., Suárez, A., López, C., Fernández, M., López, J.C., Rodríguez, R.F.: Fast simulation of self-similar and correlated processes. Mathematics and Computers in Simulation 80(10), 2040–2061 (2010)
22. Sousa, M.E., Suárez, A., Rodríguez, R.F., López, C.: Flexible adjustment of the short-term correlation of LRD M/G/$\infty$-based processes. Lecture Notes in Theoretical Computer Science 261, 131–145 (2010)
23. Sousa, M.E., Suárez, A., Fernández, M., López, J.C., López, C., Rodríguez, R.F.: Application of a hypothesis test for discriminating long-memory processes to the M/G/$\infty$ process. In: Proc. Statistical Methods of Signal and Data Processing, Kiev, Ukraine (2010)
24. Suárez, A., López, J.C., López, C., Fernández, M., Rodríguez, R.F., Sousa, M.E.: A new heavy-tailed discrete distribution for LRD M/G/$\infty$ sample generation. Performance Evaluation 47(2/3), 197–219 (2002)
25. Tsoukatos, K.P., Makowski, A.M.: Heavy traffic analysis for a multiplexer driven by M/G/$\infty$ input processes. In: Proc. 15th International Teletraffic Congress, Washington, DC, USA, pp. 497–506 (1997)
26. Whittle, P.: Estimation and information in stationary time series. Arkiv Matematick 2(23), 423–434 (1953)

# Approximating Performance Measures of a Triple Play Loss Network Model*

Irina A. Gudkova and Konstantin E. Samouylov

Telecommunication Systems Department,
Peoples' Friendship University of Russia,
Ordzhonikidze str. 3, 115419 Moscow, Russia
igudkova@sci.pfu.edu.ru,ksam@sci.pfu.edu.ru
http://www.telesys.pfu.edu.ru

**Abstract.** Modern multi-service networks are inseparably linked with the commercial concept "triple play" that implies simultaneous provisioning of telephony, television (broadcast television and video on demand) and data transmission (mostly TCP-based best effort traffic) over a single broadband connection. These services generate traffics of three types – unicast streaming, multicast streaming and elastic traffics. In this paper, we propose and analyze a multi-service model of a triple play single-link network. Many research teams, including authors of the present paper, were not able to find any exact analytical solution or recurrent algorithm for models with a mixture of streaming and elastic traffics. We succeeded in developing and evaluating an approximation valid for the calculation of the elastic traffic mean transfer time for the proposed realistic traffic model.

**Keywords:** triple play network, single-link, streaming traffic, elastic traffic, unicast, multicast, blocking probability, mean transfer time, approximation.

## 1 Introduction

Some time ago, companies such as mobile operators, cable operators, internet service providers, were in different markets. The first ones offered wireless services, mainly telephony; the second ones were focused on TV and video; the last ones delivered IP data services. Nowadays, the telecom industry sees network convergence and telecom operators are employing new strategies to deliver new services via next generation networks (NGN). The network convergence is inseparably linked with the commercial concept "triple play" [1]. This term involves multiple services, multiple devices, but one service provider, one network, one bill. Usually, the triple play concept comprises three joint service categories "voice", "video" and "data". Each category consists of many services [1], [2], for

* This work was supported in part by the Russian Foundation for Basic Research (grant 10-07-00487-a).

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 360–369, 2011.

example, VoIP, Skype, SIP-telephony; IPTV, VoD, streaming video via P2P; file transfer, e-mail, instant messaging.

Evidently, such various services generate traffics discriminating one from others not only in users' popularity and traffic volume but also in sensitivity to packet losses, bit-rate, duration, and etc. Three major traffic types are generally defined: unicast streaming, multicast streaming and elastic traffics. Nevertheless, pair matching "voice" – "unicast traffic", "video" – "multicast traffic" and "data" – "elastic traffic" is rather rough and needs to be specified. Streaming traffic is considered to be real-time and is characterized by a fixed duration, whereas elastic traffic is not real time and is assumed to have a variable duration and fixed volume. Unlike unicast traffic, multicast traffic has a network resources' saving nature achieved through employing multicast technology.

In terms of analysis of mathematical models with three traffic types, the teletraffic theory is developing step-by-step. First of all, models with one traffic type were developed [3]–[10], at that models with multicast traffic are well-known and were developed in [6] and [7]. After that, researchers were aimed at pairwise combinations of traffic types, i. e. unicast and multicast [4], [11]–[14], unicast and elastic [15], [16]. The analytical solutions and recursive algorithms were derived for the models without elastic traffic, while the combination of unicast and elastic traffics requires developing approximate methods. The "triple play" model has not all the more been developed and any approximation has not been proposed.

In this paper, our research relies on the yearly renewable statistics performed by the Cisco Systems. In Sect. 2, we apply the so-called Cisco Visual Networking Index (VNI) Usage research [2] to develop a triple play traffic model. In Sect. 3, we propose a single-link mathematical model with three incoming streams, i. e. one stream per one traffic type, and then we derive an approximating solution valid for the traffic model. Finally, accounting for the realistic traffic shares, in Sect. 4 we evaluate the approximation and give a case study. A conclusion of this paper is given in Sect. 5.

## 2 Triple Play Traffic Model

Numerical analysis of performance measures requires developing a triple play service's traffic model. This task was not a subject of the paper; nevertheless we decided to make the analysis based on known datasets choosing the data from the Cisco web-site as a reference.

The Cisco VNI is Cisco's ongoing effort to forecast and analyze the growth and use of IP-networks worldwide. The Cisco VNI Usage research [2] provides quantitative insights into current activity on service provider networks and qualitative samples of consumers' online behavior. In this cooperative program, more than 20 global service providers share anonymous, aggregate data with Cisco to analyze current network usage trends and gauge future infrastructure requirements. Participating service providers serve millions of subscribers from around the world. They represent the mobile, wireline, and cable segments throughout North America, Latin America, Europe, Asia Pacific and various emerging markets.

Cisco report contains also statistics of shares of traffic generated by diverse applications that is summarized in Table 1. The "data" and "video" service categories are the largest ones corresponding to six applications groups, whereas "voice" composes only about 2% of the overall traffic. Application subgroups pointed out in the table make us possible to associate every application with unicast, multicast or elastic traffic.

**Table 1.** Triple play traffic shares

| Application group | Application subgroup | Traffic type | Traffic share |
|---|---|---|---|
| File sharing (43.54%) | P2P | Elastic | 24.85% |
| | Web-based and other file sharing | Elastic | 18.69% |
| Data (27.51%) | HTTP | Elastic | 26.05% |
| | HTTPS | Elastic | 0.63% |
| | VPN and tunneling | Unicast | 0.57% |
| | Admin | Elastic | 0.16% |
| | Default service | Elastic | 0.09% |
| | Maps | Elastic | 0.01% |
| Online video (26.15%) | Streaming video | Multicast | 10.52% |
| | Flash video | Unicast | 6.99% |
| | Streaming video via P2P | Unicast | 4.80% |
| | Audio and video over HTTP | Unicast | 3.56% |
| | Video downloads | Elastic | 0.28% |
| Voice and video communications (1.71%) | Other VoIP | Unicast | 0.64% |
| | Skype | Unicast | 0.57% |
| | MGCP | Unicast | 0.40% |
| | SIP | Unicast | 0.05% |
| | Voice and video over instant messaging | Unicast | 0.04% |
| | Phone VoIP | Unicast | 0.01% |
| Gaming (0.81%) | PC gaming | Unicast | 0.65% |
| | Xbox | Unicast | 0.12% |
| | PlayStation | Unicast | 0.04% |
| Data communications (0.28%) | E-mail | Elastic | 0.19% |
| | Instant messaging | Elastic | 0.07% |
| | Instant messaging file transfer | Elastic | 0.02% |

Summarizing shares of homogeneous traffics we get Table 2 and, without losing generality, consider 20% share of unicast traffic, 10% share of multicast traffic and 70% share of elastic traffic. It is important to note that multicast traffic is only represented by streaming video, i. e. IPTV application. Applications generating unicast traffic are subdivided into two groups – more and less demanding network resources. As long as the share of the last group makes only about 2% we neglect it and deal with unicast traffic from online video, i. e. VoD application. Finally, elastic traffic unions all non-real time applications' subgroups.

**Table 2.** Aggregate triple play traffic shares

| Traffic type | Aggregate traffic share | Application group |
|---|---|---|
| Unicast | 18.44% | Online video (VoD) & gaming (16.16%)<br>Others (2.28%) |
| Multicast | 10.52% | Streaming video (IPTV) |
| Elastic | 71.04% | All subgroups |

The streaming traffic (VoD, IPTV) bit-rates depend on compressing and coding video streams. For example, the bit-rate per one HDTV video channel with the modern MPEG-4 part 10 (ITU-T AVC/H.264) compression is in the range of 7.5–13 Mbps, comparing to 15 Mbps for the earlier MPEG-2 (ITU-T H.262) coding. The similar bit-rate ratio is observed for SDTV channels, i. e. 2.0–3.2 Mbps (MPEG-4) vs. 3.5 Mbps (MPEG-2) [1, Table 3.5]. As for elastic traffic, it is reasonable to guarantee some minimum bit-rate to it for complying with ITU-T G.1010 requirements: the preferred one way delay of 10 MB data must be less than 15 seconds, the acceptable delay must be less than 60 seconds. Relying on [1, Fig. 1.7] we assume this bit-rate to be 1 Kbps. Concluding the traffic model characterization, we specify average durations of network resource occupancy. They differ for VoD and IPTV applications and make 3 and 1 hour respectively [1, Fig. 1.7].

Thereby, we developed the traffic model containing all necessary data for numerical analysis of performance measures of a single-link triple play network, namely blocking probabilities and average elastic traffic delay. In the following section, we propose a single-link multi-service loss-like model and an approximate method to compute the corresponding probability distribution.

## 3 Single-Link Mathematical Model

We consider a single-link network of $C$ capacity units (c. u.) shared by unicast (u), multicast (m) and elastic (e) traffics. Below, the model functioning is described in terms of unicast connections, multicast service/session and elastic flows. Table 3 gives all necessary notations. We assume all arrival rates $\lambda_i$ to be Poisson and the resource occupancy durations to be exponential distributed with means $\mu_i^{-1}$. Each type of traffic has a rate guarantee of $b_i$ c. u. From the teletraffic theory point of view, the main distinctions between three incoming streams are in service disciplines and different interpretations of traffic load $\rho_i$, $i \in \{\text{u, m, e}\}$.

Requests for a unicast connection setup are served according to the first come – first served (FCFS) discipline. Requests for the multicast service providing are handled in compliance with the so-called "transparent" discipline [2, discipline $T_2$], i. e. the multicast session initiator occupies just once free $b_{\text{m}}$ c. u. for the overall session duration, which is successively increased by new arriving

requests for the service. Elastic flows share the link capacity free from streaming traffic with the egalitarian processor sharing (EPS) discipline [17].

Note that every traffic type has its own interpretation of the traffic load, i. e. number of unicast connections, number of multicast users and average volume of elastic traffic offered to the link per 1 time unit (t. u.), respectively.

**Table 3.** Model parameters

| Parameter | Traffic type ($i$) | Description |
|---|---|---|
| $\lambda_i$ | u | Arrival rate of requests for a unicast connection setup |
| | m | Arrival rate of requests for the multicast service providing |
| | e | Elastic flows' arrival rate |
| $\mu_i^{-1}$ | u | Mean holding period of a unicast connection (t. u.) |
| | m | Mean residency time of each multicast user (t. u.) |
| | e | Elastic flow mean size (c. u. $\times$ t. u.) |
| $b_i$ | u | Fixed rate guarantee for a unicast connection (c. u.) |
| | m | Fixed rate guarantee for a multicast session (c. u.) |
| | e | Minimum rate guarantee for an elastic flow (c. u.) |
| $\rho_i := \lambda_i \mu_i^{-1}$ | u | Offered load due to unicast connections (Erlang) |
| | m | Offered load due to multicast users (Erlang) |
| | e | Elastic traffic intensity (c. u.) |
| $n_i$ | u | Number of unicast connections |
| | m | State of a multicast session:<br>$n_{\mathrm{m}} = 1$ – session is active, $n_{\mathrm{m}} = 0$ – otherwise |
| | e | Number of elastic flows |

Let $\boldsymbol{n} := (n_i)_{i \in \{\mathrm{u,\ m,\ e}\}} = (n_{\mathrm{u}}, n_{\mathrm{m}}, n_{\mathrm{e}})$ be the state of the system and $b(\boldsymbol{n}) := \sum_{i \in \{\mathrm{u,\ m,\ e}\}} b_i n_i$ be the rate guarantee for all traffics types when the system is in state $\boldsymbol{n}$. Then, the system state space is given by

$$\mathcal{X} := \{\boldsymbol{n} \geq \boldsymbol{0} : \quad b(\boldsymbol{n}) \leq C\} \quad . \tag{1}$$

Below we introduce the following notation that will be used up to the end of the paper:

- $\boldsymbol{e}_{\mathrm{u}} := (1,0,0)$, $\boldsymbol{e}_{\mathrm{m}} := (0,1,0)$, $\boldsymbol{e}_{\mathrm{e}} := (0,0,1)$ – the three-dimensional unit vectors;
- $\mathcal{A}_i := \{\boldsymbol{n} \in \mathcal{X} : \ b(\boldsymbol{n} + \boldsymbol{e}_i) \leq C\}$, $i \in \{\mathrm{u,\ e}\}$ – the sets of system states where a new arrived request for a unicast connection setup or a new arrived elastic flow is accepted;
- $\mathcal{A}_{\mathrm{m}} := \{\boldsymbol{n} \in \mathcal{X} : \ [b(\boldsymbol{n} + \boldsymbol{e}_{\mathrm{m}}) \leq C] \ \wedge \ [n_{\mathrm{m}} = 0]\}$ – the set of system states where a new arrived request for a multicast session opening is accepted;
- $c_i\left(n_j\right)_{j \in \{\mathrm{u,\ m,\ e}\} \setminus \{i\}} := C - \sum_{j \in \{\mathrm{u,\ m,\ e}\} \setminus \{i\}} b_j n_j$, $i \in \{\mathrm{u,\ e}\}$ – the link capacity available for new arriving unicast connections or elastic flows in state $(\bullet, n_{\mathrm{m}}, n_{\mathrm{e}})$ or $(n_{\mathrm{u}}, n_{\mathrm{m}}, \bullet)$ respectively;

– $N_i\left(n_j\right)_{j\in\{\mathrm{u},\ \mathrm{m},\ \mathrm{e}\}\setminus\{i\}} := \left\lfloor \frac{c(n_j)_{j\in\{\mathrm{u},\ \mathrm{m},\ \mathrm{e}\}\setminus\{i\}}}{b_i} \right\rfloor$, $i \in \{\mathrm{u},\ \mathrm{e}\}$ – the maximum number of unicast connections or elastic flows that might be in the link in state $(\bullet, n_\mathrm{m}, n_\mathrm{e})$ or $(n_\mathrm{u}, n_\mathrm{m}, \bullet)$ respectively, where $\lfloor\bullet\rfloor$ is the integer part;
– $1\{\bullet\}$ – the indicator function.

It could be simply proofed that the process representing the system states is not a reversible Markov process and the solution of equilibrium equations

$$
\begin{aligned}
p(\boldsymbol{n}) \Bigg( & \sum_{i\in\{\mathrm{u},\ \mathrm{m},\ \mathrm{e}\}} \lambda_i \cdot 1\{\boldsymbol{n}\in\mathcal{A}_i\} + n_\mathrm{u}\mu_\mathrm{u} + \\
& + n_\mathrm{m}\lambda_\mathrm{m}\left(e^{\rho_\mathrm{m}}-1\right)^{-1} + c_\mathrm{e}\left(n_\mathrm{u}, n_\mathrm{m}\right)\mu_\mathrm{e}\cdot 1\{n_\mathrm{e}>0\}\Bigg) = \\
= & \sum_{i\in\{\mathrm{u},\ \mathrm{m},\ \mathrm{e}\}} p(\boldsymbol{n}-\boldsymbol{e}_i)\cdot\lambda_i\cdot 1\{n_i>0\} + \\
& + p(\boldsymbol{n}+\boldsymbol{e}_\mathrm{u})\cdot(n_\mathrm{u}+1)\,\mu_\mathrm{u}\cdot 1\{\boldsymbol{n}\in\mathcal{A}_\mathrm{u}\} + \\
& + p(\boldsymbol{n}+\boldsymbol{e}_\mathrm{m})\cdot\lambda_\mathrm{m}\left(e^{\rho_\mathrm{m}}-1\right)^{-1}\cdot 1\{\boldsymbol{n}\in\mathcal{A}_\mathrm{m}\} + \\
& + p(\boldsymbol{n}+\boldsymbol{e}_\mathrm{e})\cdot c_\mathrm{e}\left(n_\mathrm{u}, n_\mathrm{m}\right)\mu_\mathrm{e}\cdot 1\{\boldsymbol{n}\in\mathcal{A}_\mathrm{e}\}, \quad \boldsymbol{n}=(n_\mathrm{u}, n_\mathrm{m}, n_\mathrm{e})\in\mathcal{X}
\end{aligned}
\tag{2}
$$

is not of product form, i. e. $p(\boldsymbol{n}) = p(n_\mathrm{u}, n_\mathrm{m}, n_\mathrm{e}) \neq p(n_\mathrm{u})\,p(n_\mathrm{m})\,p(n_\mathrm{e})$. This fact arises from the dependence of current elastic flows' rate $c_\mathrm{e}(n_\mathrm{u}, n_\mathrm{m})\,\mu_\mathrm{e}$ on streaming traffic state $(n_\mathrm{u}, n_\mathrm{m}, \bullet)$.

The authors of this paper tested a deal of approximating solutions, but only one of them was compatible with the triple play traffic model discussed in the previous section. So, let $p_1(n_\mathrm{e} \mid n_\mathrm{u}, n_\mathrm{m})$ be the conditional probability that there are $n_\mathrm{e}$ elastic flows in the link given system state $(n_\mathrm{u}, n_\mathrm{m}, \bullet)$; and let $p_2(n_\mathrm{u}, n_\mathrm{m})$ be the marginal probability that the system is in state $(n_\mathrm{u}, n_\mathrm{m}, \bullet)$. Then, the following lemmas 1 and 2 might be used for calculating corresponding probability distributions.

**Lemma 1.** *The probability distribution* $p_1(\bullet \mid \bullet, \bullet)$ *can be computed as*

$$
\begin{aligned}
& p_1(n_\mathrm{e} \mid n_\mathrm{u}, n_\mathrm{m}) = \\
& \quad = \left(\frac{\rho_\mathrm{e}}{c_\mathrm{e}(n_\mathrm{u}, n_\mathrm{m})}\right)^{n_\mathrm{e}} \frac{\left[c_\mathrm{e}(n_\mathrm{u}, n_\mathrm{m})\right]^{N_\mathrm{e}(n_\mathrm{u}, n_\mathrm{m})}\left[c_\mathrm{e}(n_\mathrm{u}, n_\mathrm{m}) - \rho_\mathrm{e}\right]}{\left[c_\mathrm{e}(n_\mathrm{u}, n_\mathrm{m})\right]^{N_\mathrm{e}(n_\mathrm{u}, n_\mathrm{m})+1} - (\rho_\mathrm{e})^{N_\mathrm{e}(n_\mathrm{u}, n_\mathrm{m})+1}}\ , \\
& n_\mathrm{m} = 0,\ 1,\ \ n_\mathrm{u} = 0, \ldots, N_\mathrm{u}(1, 0),\ \ n_\mathrm{e} = 0, \ldots, N_\mathrm{e}(n_\mathrm{u}, n_\mathrm{m})\ .
\end{aligned}
\tag{3}
$$

Note that probability distribution (3) is derived from the well-known results for the processor-sharing theory [17]. Then, we also denote by $A_\mathrm{m}(n_\mathrm{u}) :=$ $:= \sum_{n_\mathrm{e}=0}^{N_\mathrm{e}(n_\mathrm{u}, 1)} p_1(n_\mathrm{e} \mid n_\mathrm{u}, 0)$ the probability that a new arrived request for a multicast session opening is accepted having found $n_\mathrm{u}$ unicast connections in the link; and we denote by $A_\mathrm{u}(n_\mathrm{u}, n_\mathrm{m}) := \sum_{n_\mathrm{e}=0}^{N_\mathrm{e}(n_\mathrm{u}+1, n_\mathrm{m})} p_1(n_\mathrm{e} \mid n_\mathrm{u}, n_\mathrm{m})$ the probability that a new arrived request for a unicast connection setup is accepted having found $n_\mathrm{u}$ unicast connections and the multicast service in state $n_\mathrm{m}$ in the link.

**Lemma 2.** *The probability distribution* $p_2(\bullet,\bullet)$ *can be approximated by distribution* $\tilde{p}_2(\bullet,\bullet)$ *as follows*

$$p_2(n_\mathrm{u},n_\mathrm{m}) \approx \tilde{p}_2(n_\mathrm{u},n_\mathrm{m}) = \\ = G^{-1}\cdot\left[(e^{\rho_\mathrm{m}}-1)^{n_\mathrm{m}} A_\mathrm{m}^{n_\mathrm{m}}(n_\mathrm{u})\right]\cdot\left[\frac{\rho_\mathrm{u}^{n_\mathrm{u}}}{n_\mathrm{u}!}\prod_{i=0}^{n_\mathrm{u}-1} A_\mathrm{u}(i,n_\mathrm{m})\right] , \quad (4)$$
$$n_\mathrm{m}=0,\ 1,\ \ n_\mathrm{u}=0,\ldots,N_\mathrm{u}(n_\mathrm{m},0) \ ,$$

*where* $G$ *is the normalization constant for probability distribution* $\tilde{p}_2(\bullet,\bullet)$.

The distribution $\tilde{p}_2(\bullet,\bullet)$ is just an approximation of $p_2(\bullet,\bullet)$ due to averaging intensities of transactions between macro-states. This fact also leads to appearance of coefficients $A_\mathrm{m}(\bullet)$ and $A_\mathrm{u}(\bullet,\bullet)$ in formula (4). Setting them equal to 1, i. e. blocking all elastic traffic, we get the result from [14]. Thereby, joint probability distribution $p(\bullet,\bullet,\bullet)$ can be calculated as the product of conditional distribution (3) and approximating marginal distribution (4):

$$p(\boldsymbol{n}) = p(n_\mathrm{u},n_\mathrm{m},n_\mathrm{e}) = p_1(n_\mathrm{e}\mid n_\mathrm{u},n_\mathrm{m})\cdot p_2(n_\mathrm{u},n_\mathrm{m}) \approx \\ \approx p_1(n_\mathrm{e}\mid n_\mathrm{u},n_\mathrm{m})\cdot\tilde{p}_2(n_\mathrm{u},n_\mathrm{m}), \quad (n_\mathrm{u},n_\mathrm{m},n_\mathrm{e})=\boldsymbol{n}\in\mathcal{X} \ . \quad (5)$$

## 4 Case Study

Having found probability distribution $p(\bullet,\bullet,\bullet)$ of the single-link triple play network, one could compute its performance measures, notable blocking probabilities $B_i$, $i\in\{\mathrm{u},\ \mathrm{m},\ \mathrm{e}\}$ and mean elastic flow duration $T_\mathrm{e}$, i. e. the mean time needed to transmit elastic traffic:

$$B_\mathrm{u} = \sum_{n_\mathrm{e}=0}^{N_\mathrm{e}(0,0)} p(N_\mathrm{u}(0,n_\mathrm{e}),0,n_\mathrm{e}) + \sum_{n_\mathrm{e}=0}^{N_\mathrm{e}(0,1)} p(N_\mathrm{u}(1,n_\mathrm{e}),1,n_\mathrm{e}) \ , \quad (6)$$

$$B_\mathrm{m} = \sum_{n_\mathrm{u}=0}^{N_\mathrm{u}(0,0)}\sum_{n_\mathrm{e}=0}^{N_\mathrm{e}(n_\mathrm{u},0)} 1\{(n_\mathrm{u},1,n_\mathrm{e})\notin\mathcal{X}\}\cdot p(n_\mathrm{u},0,n_\mathrm{e}) \ , \quad (7)$$

$$B_\mathrm{e} = \sum_{n_\mathrm{u}=0}^{N_\mathrm{u}(0,0)} p(n_\mathrm{u},0,N_\mathrm{e}(n_\mathrm{u},\ 0)) + \sum_{n_\mathrm{u}=0}^{N_\mathrm{u}(1,0)} p(n_\mathrm{u},1,N_\mathrm{e}(n_\mathrm{u},1)) \ , \quad (8)$$

$$T_\mathrm{e} = \frac{\sum_{n_\mathrm{e}=1}^{N_\mathrm{e}(0,0)} n_\mathrm{e}\sum_{n_\mathrm{u}=0}^{N_\mathrm{u}(0,n_\mathrm{e})} p(n_\mathrm{u},0,n_\mathrm{e}) + \sum_{n_\mathrm{e}=1}^{N_\mathrm{e}(0,1)} n_\mathrm{e}\sum_{n_\mathrm{u}=0}^{N_\mathrm{u}(1,n_\mathrm{e})} p(n_\mathrm{u},1,n_\mathrm{e})}{\lambda_\mathrm{e}(1-B_\mathrm{e})} \ . \quad (9)$$

We are also interested in utilization factor $UTIL$ of the link, as the triple play traffic model from Sect. 2 is primarily characterized by the fixed proportions of traffic shares, i. e. $\gamma_\mathrm{u}:\gamma_\mathrm{m}:\gamma_\mathrm{e}=2:1:7$:

$$UTIL := \sum_{i \in \{\mathrm{u},\ \mathrm{m},\ \mathrm{e}\}} UTIL_i \ , \tag{10}$$

where

$$UTIL_i := \frac{1}{C} b_i \sum_{(n_\mathrm{u}, n_\mathrm{m}, n_\mathrm{e}) \in \mathcal{X}} n_i \cdot p\,(n_\mathrm{u}, n_\mathrm{m}, n_\mathrm{e}) \ , \quad i \in \{\mathrm{u},\ \mathrm{m}\} \ , \tag{11}$$

$$UTIL_\mathrm{e} := \frac{1}{C} \sum_{(n_\mathrm{u}, n_\mathrm{m}, n_\mathrm{e}) \in \mathcal{X}} c_\mathrm{e}\,(n_\mathrm{u}, n_\mathrm{m}) \cdot 1\,\{n_\mathrm{e} > 0\} \cdot p\,(n_\mathrm{u}, n_\mathrm{m}, n_\mathrm{e}) \ . \tag{12}$$

Considering the proposed traffic model we evaluate the developed approximation. Numerical analysis shows that the method gives relatives errors less than 1% for calculating the mean elastic traffic transfer time. Increasing the link capacity (Table 4) does not strongly affect relative errors as they fluctuate around some fixed value not exhibiting a tendency to growth. On the contrary, increasing the link utilization factor (Table 5) is associated with the error augmentation. As for blocking probabilities, relative errors could reach 50%.

**Table 4.** Relative errors for the mean elastic traffic duration depending on the link capacity ($C = 50, \ldots, 100$, $UTIL = 0.7$, $\gamma_\mathrm{u} = 0.2$, $\gamma_\mathrm{m} = 0.1$, $\gamma_\mathrm{e} = 0.7$, $b_\mathrm{u} = 2$, $b_\mathrm{u} = C \cdot \gamma_\mathrm{m}$, $b_\mathrm{e} = 1$, $\mu_\mathrm{u}^{-1} = 10800$, $\mu_\mathrm{m}^{-1} = 3600$, $\mu_\mathrm{e}^{-1} = C$)

| | $C$ | 50 | 60 | 70 | 80 | 90 | 100 |
|---|---|---|---|---|---|---|---|
| $T_\mathrm{e}$ | Exact (sec) | 3.47 | 3.45 | 3.44 | 3.43 | 3.42 | 3.42 |
| | Error (%) | 0.048 | 0.058 | 0.055 | 0.049 | 0.042 | 0.036 |

**Table 5.** Relative errors for the mean elastic traffic duration depending on the utilization factor ($C = 100$, $UTIL = 0.6, \ldots, 0.8$, $\gamma_\mathrm{u} = 0.2$, $\gamma_\mathrm{m} = 0.1$, $\gamma_\mathrm{e} = 0.7$, $b_\mathrm{u} = 2$, $b_\mathrm{u} = 10$, $b_\mathrm{e} = 1$, $\mu_\mathrm{u}^{-1} = 10800$, $\mu_\mathrm{m}^{-1} = 3600$, $\mu_\mathrm{e}^{-1} = 80$)

| | $UTIL$ | 0.60 | 0.64 | 0.68 | 0.72 | 0.76 | 0.80 |
|---|---|---|---|---|---|---|---|
| $T_\mathrm{e}$ | Exact (sec) | 2.07 | 2.32 | 2.64 | 3.08 | 3.73 | 4.78 |
| | Error (%) | 0.0062 | 0.0110 | 0.0231 | 0.0591 | 0.1622 | 0.3835 |

Finally, we apply the approximation to compute performance measures of a 100 Mbps Fast Ethernet assess link under the traffic model from Sect. 2. 50 SDTV (MPEG-4) IPTV channels and SDTV (MPEG-4) VoD applications with 2 Mbps bandwidth requirements are transmitted over the link. The rest of the link is shared by data traffic with a minimum bandwidth requirement of 1 Kbps and mean size of 10 MB. We let 1 c. u. equal to 1 Kbps. Figure 1 shows behavior of the approximated measures in dependence on the utilization factor. Similarly to numerical experiment [13], blocking probabilities fluctuate periodically. After a qualitative analysis of the graphs in Fig. 1 we could recommend not overloading

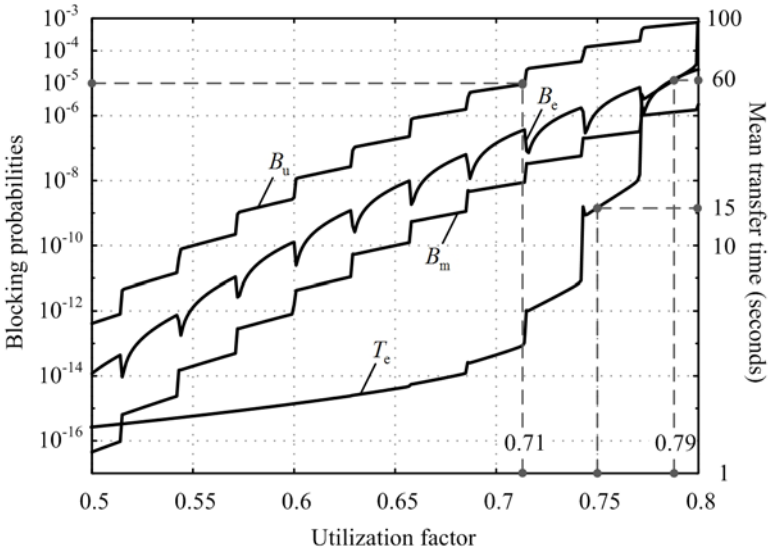



**Fig. 1.** Approximating performance measures of the triple play single-link network. The model parameters' values: $C = 100000$, $UTIL \in [0.5, \ldots, 0.8]$, $\gamma_u = 0.2$, $\gamma_m = 0.1$, $\gamma_e = 0.7$, $b_u = 2000$, $b_u = 10000$, $b_e = 1$, $\mu_u^{-1} = 10800$, $\mu_m^{-1} = 3600$, $\mu_e^{-1} = 81920$

the link more than on 75% if a telecom provider wants to guarantee the preferred threshold of 15 seconds to the data traffic average delay. If this time is not so critical for consumers then the link could be overloaded up to 80% to guarantee nevertheless the acceptable threshold of 60 seconds.

## 5 Conclusion

In this paper, we addressed a resource sharing problem in multi-service networks, and the mathematical model of triple play network was proposed for the first time. The model incorporates three traffic types – unicast streaming, multicast streaming and elastic, at that the presence of elastic traffic makes it impossible to analyze the model using known methods allowing a product form solution. Therefore we propose an approximate method and evaluate it accuracy with errors less than 1% for the average delay of elastic traffic's transmission. We also see an interesting question for future work is to develop a recursive algorithm to calculate performance measures of the triple play network model.

## References

1. Hens, F.J., Caballero, J.M.: Triple Play: Building the Converged Network for IP, VoIP and IPTV. Johns Wiley & Sons Ltd., Chichester (2008)
2. Cisco Visual Networking Index Usage Study. Cisco (2010), http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/Cisco_VNI_Usage_WP.html

3. Ross, K.W.: Multiservice Loss Models for Broadband Telecommunication Networks. Springer, New York (1995)
4. Basharin, G.P., Samouylov, K.E., Yarkina, N.V., Gudkova, I.A.: A New Stage in Mathematical Teletraffic Theory. Automation and Remote Control 70(12), 1954–1964 (2009)
5. Chan, W.C., Geraniotis, E.: Tradeoff between Blocking and Dropping in Multicasting Networks. In: Proc. of IEEE International Conference on Communications, Dallas, USA, vol. 2, pp. 1030–1034 (1996)
6. Karvo, J., Virtamo, J.T., Aalto, S., Martikainen, O.: Blocking of Dynamic Multicast Connections in a Single Link. In: Proc. of the 4-th International Conference on Broadband Communications, Stuttgart, Germany, pp. 473–483 (1998)
7. Gaidamaka, Y., Samouylov, K.: Analytical Model of Multicast Network and Single Link Performance Analysis. In: Proc. of the 6-th International Conference on Telecommunications, Zagreb, Croatia, pp. 169–175 (2001)
8. Bonald, T., Virtamo, J.: A Recursive Formula for Multirate Systems with Elastic Traffic. IEEE Communications Letters 9(8), 753–755 (2005)
9. Vassilakis, V., Moscholios, I., Logothetis, M.: Call-Level Performance Modelling of Elastic and Adaptive Service-Classes with Finite Population. IEICE Transactions on Communications 91-b(1), 151–163 (2008)
10. Samouylov, K.E., Gudkova, I.A.: Recursive Computation for a Multi-Rate Model with Elastic Traffic and Minimum Rate Guarantees. In: Proc. of the International Congress on Ultra Modern Telecommunications and Control Systems, Moscow, Russia, pp. 1065–1072 (2010)
11. Boussetta, K., Beylot, A.-L.: Multirate Resource Sharing for Unicast and Multicast Connections. In: Proc. of the 5-th International Conference on Broadband Communications, Hong Kong, pp. 561–570 (1999)
12. Avramova, Z., De Vleeschauwer, D., Wittevrongel, S., Bruneel, H.: Capacity Gain of Mixed Multicast/Unicast Transport Schemes in a TV Distribution Network. IEEE Transactions on Multimedia 11(5), 918–931 (2009)
13. Samouylov, K., Yarkina, N.: Blocking Probabilities in Multiservice Networks with Unicast and Multicast Connections. In: Proc. of the 7-th International Conference on Telecommunications, Zagreb, Croatia, pp. 423–429 (2005)
14. Gudkova, I.A., Plaksina, O.N.: Performance Measures Computation for a Single Link Loss Network with Unicast and Multicast Traffics. In: Balandin, S., Dunaytsev, R., Koucheryavy, Y. (eds.) ruSMART 2010. LNCS, vol. 6294, pp. 256–265. Springer, Heidelberg (2010)
15. Tan, H.-P., Núñez-Queija, R., Gabor, A.F., Boxma, O.J.: Admission Control for Differentiated Services in Future Generation CDMA Networks. Performance Evaluation 66(9-10), 488–504 (2009)
16. Karray, M.K.: Analytical Evaluation of QoS in the Downlink of OFDMA Wireless Cellular Networks Serving Streaming and Elastic Traffic. IEEE Transactions on Wireless Communications 9(5), 1799–1807 (2010)
17. Yashkov, S.F.: Mathematical Problems in the Theory of Shared-Processor Systems. Journal of Mathematical Sciences 58(2), 101–147 (1992)

# An Analytical Model for Streaming over TCP

Jinyao Yan[1,2], Wolfgang Mühlbauer[1], and Bernhard Plattner[1]

[1] Computer Engineering and Networks Laboratory, ETH Zurich, CH-8092, Switzerland
{jinyao,muehlbauer,plattner}@tik.ee.ethz.ch
[2] Communication University of China, 100024, Beijing, China
jyan@cuc.edu.cn

**Abstract.** Streaming over TCP has become popular as demonstrated by the example of YouTube. To cope with variability in data throughput, streaming applications typically implement buffers. Yet, for improving the quality of user experience, it is critical to dimension buffers and initial buffering delays appropriately.

In this paper, we develop an analytical framework that describes the dimensioning of appropriate buffers. To this end, we propose to rely on modeling congestion window sizes immediately before a triple duplicate or timeout event. We observe that such "bounds" on TCP window sizes follow a Gamma distribution.

Although being of general use due to its simplicity and accuracy, our proposed TCP model is particularly useful for TCP streaming. As confirmed by experiments, it allows to estimate the frequency of buffer overflow or underflow events if buffer sizes and initial buffering delays are known parameters in the proposed TCP streaming model, or conversely, to dimension the buffer appropriately.

## 1 Introduction

TCP is the dominant transport protocol in the Internet with the majority of applications including WWW and E-mail relying on it. Recent measurements [11,6] have shown that streaming over TCP has become popular as demonstrated by video-sharing websites such as YouTube. One important reason is that streaming applications over HTTP and TCP are less likely to be affected by firewalls. However, the throughput and performance of streaming (but also other) applications strongly depend on the congestion behavior of the underlying TCP [9,3,13].

To cope with variability in data throughput, streaming applications typically implement buffers, into which receivers insert incoming streaming content. To achieve a steady playout, data is removed from this buffer at a constant rate. Ideally, both *the initial buffering delay* and *the size of the buffer* should be as small as possible, yet large enough to avoid *buffer underflows* and *overflows*. While playback stops during underflows, playback video is choppy during overflows. An empirical evaluation in [14] for commercial multimedia software such as Skype, Google Talk and MSN Messenger shows that these applications do not dimension and adjust their buffer sizes very well.

In this paper, we develop an analytical framework that describes the dimensioning of appropriate buffers. To achieve this goal, we first derive an analytical TCP model that is tailored to TCP streaming applications. Our TCP model is to consider time intervals between consecutive timeout or triple duplicate events and refer to these as timeout periods (TOP) and triple duplicate periods (TDP), respectively. More precisely, we study

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 370–381, 2011.

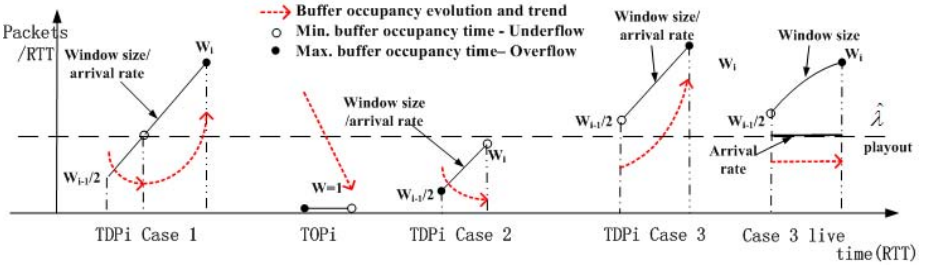



**Fig. 1.** Window bounds and buffer occupancy for TCP streaming – TDP Case 1: window sizes and arrival rates roughly match playout rate; TDP Case 2: window sizes and arrival rates are smaller than stream playout rate; TDP Case 3: window sizes and arrival rates are higher than the stream playout rate for stored streams while the arrival rate is constrained by the coding rate for live streams; TOP: window sizes during TOP are set to be one and the buffer occupancy is monotonously decreasing.

the "bounds" on TCP congestion window sizes, i.e., the window sizes immediately before a triple duplicate event or a timeout event. We find that these "bounds" follow a Gamma distribution which allows to present a surprisingly simple analytical solution for the distribution of TCP window bounds. Previous analytical TCP models [10,9,3,2] have mainly focused on average TCP sending rates [10,2], or on the overall distribution of congestion window sizes [9,3].

After presenting our TCP model, we illustrate its use for the specific application of TCP video streaming. To this end, we develop a TCP streaming model with emphasis on the relationship between TCP arrival rate (namely TCP window size per round-trip time), buffer occupancy, and playout rate. Please note that our TCP streaming analytical framework differs from the model in [13] in terms of the TCP "bounds" model, the constrained buffer size and its occupancy analysis in our analytical framework. We observe that our concept of "bounds" on TCP congestion window sizes, allows to distinguish between three cases with respect to the times of minimum and maximum buffer occupancy within a TDP, see Figure 1. Starting from given the desired low underflow/overflow probabilities we demonstrate how to determine appropriate buffer size and initial buffering delay by using the distribution of TCP window bounds or, conversely, how to estimate underflow/overflow probabilities if the buffer size and initial buffering delay are known.

To summarize our contributions: First, we propose a new TCP window size model based on the bounds of TCP congestion window sizes at the end of TDPs or TOPs. The closed form solution is a simple, surprisingly accurate approximation, which we believe to be of general use. Second, we demonstrate in detail how to apply this model to the application of TCP streaming. Our analytical framework aligned with the classification of Figure 1 allows to describe the relationship among buffer sizes, initial buffering delay and overflow/underflow probabilities in an elegant and simple manner. Based on simulations and experiments, we verify our proposed TCP window size and streaming model, and show that it can provide useful insights for dimensioning streaming buffers.

The rest of this paper is structured as follows. Section 2 explains our model for the bounds distribution of TCP congestion window sizes, while Section 3 introduces its application to TCP streaming. In Section 4 we set up simulations and experiments to study the accuracy of our models. Finally, we conclude the paper in Section 5.

## 2 TCP Congestion Window Sizes

In this section, we study the variation of congestion window sizes, more precisely we consider bounds on congestion window sizes.

In the past, many analytical models [10,2,8] have been proposed that describe the behavior of TCP. The main focus of the literature has been on the TCP Reno, NewReno and SACK variants of TCP. These TCP implementations are used in BSD Unix systems or BSD-based systems such as Mac OS. They are also available and used in Linux although they are being replaced by new variants, e.g., CUBIC. Common to most TCP implementations is the detection of packet losses via triple duplicates or timeouts, and the adaptation of TCP window congestion window sizes to match the available bandwidth. In the following, we build our model based on the assumption that TCP Reno is used as TCP implementation. However, we believe that our key ideas, e.g., studying the distribution of TCP window bounds, can also be applied to other TCP variants.

Possibly, the most popular model is the one introduced by Padhye et al.[10]. They derived a simple formula of the TCP average rate for TCP friendly rate control algorithms [16,5]. Altman et al. [2] obtain the throughput of a TCP connection for general loss patterns. Finally, there has been work [9,3] to investigate the distribution of window sizes. Our TCP model differs from previous work since we are the first to derive an accurate distribution of bounds on TCP congestion window sizes.

In TCP Reno, the steady state throughput is determined by the congestion window size, which again depends on detected packet loss events. A packet loss can be detected by either triple-duplicate ACKs or timeout events. We refer to the time interval after a triple duplicate event until the next event as a triple duplicated period (TDP) and the time period after a timeout as timeout period (TOP).

According to this definition, a new TDP starts immediately after a triple-duplicate ACK event. Within this TDP, TCP will increase the congestion window size by $1/b$ until the next tripled duplicate event. $b$ is the number of packets that a single ACK acknowledges is set to 1 when delayed acknowledgement is not used. Whenever multiple packets have been lost and less than three duplicated ACKs have been received, a TO period (TOP) begins. During a TOP, a sender stops transmitting data packets for a certain timeout interval and retransmits non-acknowledged packets. Note that the timeout interval in a TOP increases exponentially up to $64T_0$.

### 2.1 Triple Duplicate ACKs

In general, we assume the probability of timeouts to be rather small, i.e., low network congestion. Therefore, we restrict our analysis to triple duplicates and to TDPs in the beginning. Afterwards, we extend our model by taking into account timeouts as well.

Similar to Padhye et al. [10] the duration of a *round* within a TDP period is defined as the duration between the transmission of packets and the reception of the first acknowledgment (ACK) in a congestion window. The duration of a round is equal to the round-trip time (RTT) and is assumed to be independent of the window size. According to the additive increase/multiplicative decrease algorithm, the window size increases by one packet per round and reduces to half of its size immediately after receiving a triple duplicate ACK. Let $W_i$ be the window size at the end of the $i$th TDP ($TDP_i$) after initial buffering delay. Hence, $W_{i-1}/2$ is the window size at the beginning of $TDP_i$, and $W_i$ and $W_{i-1}/2$ are the upper and lower bounds for the TCP window sizes, and the number of rounds during $TDP_i$ is $W_i - W_{i-1}/2 + 1$. We refer to $Y_i$ as the number of packets successfully sent in $TDP_i$, and compute it as follows:

$$\begin{aligned} Y_i &= \sum_{k=1}^{X_i}(\frac{W_{i-1}}{2} + k - 1) = \frac{(W_i^2 - \frac{W_{i-1}^2}{4})}{2} + \frac{W_i + \frac{W_{i-1}}{2}}{2} \\ &\approx \frac{(W_i+1)^2 - \frac{(W_{i-1}+1)^2}{4}}{2}, (W_i >> 1) \end{aligned} \tag{1}$$

where $X_i = W_i - \frac{W_{i-1}}{2} + 1$. Now, we can model the distribution of $W_i$ as a Markov process and obtain the stationary distribution of window size $W_i$, and thus also of $W_{i-1}/2$ at the beginning of the same TDP. Transforming Eq.(1) results in the status transition of $W_i$:

$$(W_i+1)^2 - \frac{(W_{i-1}+1)^2}{4} \approx 2 \cdot Y_i, \qquad i = 1, 2, ... \tag{2}$$

We assume each packet has the probability of loss $p$ and $p$ is stationary for a certain time scale. We assume further the packet losses are independent as the correlation between subsequent losses is highly reduced by the high statistical multiplexing on high speed links [4]. Thus, $P\{Y_i = n\} = (1-p)^n \cdot p$, $(n = 1, 2, \dots)$, and it is a geometric distribution for packet loss rates $p \ll 1$. Although $Y_i$ and $W_i$ are generally discrete variables, we treat them as continuous as they could become non-integers in the following derivation. Then, each $Y_i$ of $i$th TDP $(i = 1, 2, \dots)$ follows an exponential distribution:

$$P\{Y_i = x\} = p \cdot e^{x \cdot ln(1-p)} = p \cdot e^{-p \cdot x}, (x \geq 0, p \ll 1) \tag{3}$$

Note that $Y_1, Y_2, \dots, Y_i$ are independent and all follow exponential distributions. We define a new variable $\psi$ as the sum of the weighted $Y_i$ variables, in particular: $\psi = 2 \cdot \sum_{k=1}^{i} \frac{Y_k}{4^{i-k}}$. According to statistical theory [7], the distribution for sequentially independent exponentially distributed variables with the same scale parameter $\theta = 1/p$ is a Gamma distribution with the sum of the shape parameters $s$, and the scale parameter $\theta = 1/p$.

$$s = 2(\frac{1}{4^{i-1}} + \frac{1}{4^{i-2}} + \frac{1}{4^{i-3}} + ... + 1)) = 8/3, (i \to \infty) \tag{4}$$

Thus, the gamma distribution of $\psi$ is $\Gamma(8/3, 1/p)$, and

$$\begin{aligned}\psi &= 2 \cdot \sum_{k=1}^{i} \frac{Y_k}{4^{i-k}} \\ &= (W_i+1)^2 - (W_{i-1}+1)^2/4 + (W_{i-1}+1)^2/4 - (W_{i-2}+1)^2/4^2 + \ldots \\ &= (W_i+1)^2 \end{aligned} \tag{5}$$

Let $F_\psi$ is the cumulative distribution function (CDF) of $\psi$ namely $F_\psi(W) \equiv Pr(\psi \leq W)$. Then, the CDF of $W_i$

$$F(W) \equiv Pr(W_i \leq W) = Pr(\psi \leq (W+1)^2) = F_\psi((W+1)^2)$$

Therefore, the CDF of $W_i$ is,

$$F(W) = \frac{\gamma(8/3, p(W+1)^2)}{\Gamma(8/3)} = 0.6646\gamma(8/3, p(W+1)^2), W > 0 \tag{6}$$

***Remark:*** Compared with the expected value of $W_i$ in Eq.(14) in [10], our solution is the first to provide the distribution of $W_i$ (the bounds of TCP window sizes).

### 2.2 Timeouts

Now, we extend our analysis to timeouts, where the window size is fixed to one. Let $Q$ be the probability that a packet loss is recognized via a timeout. Then, the closed form for the distribution of TCP window (upper) bounds is:

$$\hat{F}(W) = (1-Q)F(W) + Q, W \geq 1 \tag{7}$$

According to Padhye et al. [10] the probability $Q$ can be approximated as follows:

$$Q \approx min(1, 3\sqrt{\frac{3p}{8}}) \tag{8}$$

By substituting Eq.(8) into Eq.(7), we finally obtain:

$$\hat{F}(W) \approx (1 - min(1, 3\sqrt{\frac{3p}{8}}))F(W) + min(1, 3\sqrt{\frac{3p}{8}}), W \geq 1 \tag{9}$$

We point out that the distribution of Eq.(6) is an approximation of Eq.(9) if the probability of a timeout is very low. In Sec. 3.3 we will leverage our insights to describe the relationship between buffer sizes and the likelihoods of overflow and underflow events in TCP streaming.

## 3 TCP Streaming System

### 3.1 System Model

Figure 2 illustrates our general model of TCP video streaming consisting of a TCP connection, a streaming buffer, and a decoder. We assume that the sender transmits video

packets over a single TCP connection. While TCP streaming applications with a single long-lived connection still dominate, e.g., progressive downloading over HTTP/TCP in YouTube, our model can be extended for future TCP streaming applications such as Dynamic Adaptive Streaming over HTTP (DASH) [12] where the client may set up several TCP connections to the HTTP server. In our model, the receiver is equipped with a buffer in front of a video decoder/player. This streaming buffer is used to accommodate the small time-scale fluctuations of the TCP sending rate. However, it must not be confused with TCP/IP buffers in the context of flow control. The decoder waits until the streaming buffer is filled to a certain degree before displaying video.

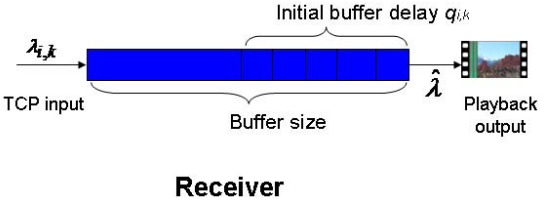



**Fig. 2.** TCP streaming model

To model arrival rates and any variations in the TCP connections, we apply the TCP model proposed in Section 2. Let $\lambda_{i,k}$ be the arrival rate (namely, the window size per *round*) of video packets at round $k$ of $TDP_i$, and $\hat{\lambda}$ be the video encoding/playback rate. It is assumed that $\hat{\lambda}$ is constant.

Since quality degradations are generally caused by a) buffer underflows that result in playout disruptions or freezings of images; b) buffer overflows that result in the loss of data or non-smooth playout; c) initial buffering delays as well as re-buffers delay to fill the buffer before starting or continuing the playout. Intuitively, the longer the buffer delay, the less likely buffer underflow events are. The larger the buffer size, the less likely buffer overflow events are.

In the following, we define the startup delay (i.e., initial buffering delay), the buffer underflow probability, buffer overflow probability, and used buffer size for video streaming applications:

**Initial buffering delay** is the time between the arrival of the first streaming packet in the buffer and the time when it is forwarded to the decoder. For simplicity, we assume that the playout of packets does not start until the buffer has been filled with at least $q_0$ packets. Hence, we define the initial buffer delay $D$ as the time to fill the buffer up to initial buffering occupancy $q_0$ at the coding rate $\hat{\lambda}$, namely $D = \frac{q_0}{\hat{\lambda}}$.

**Buffer underflow or overflow probability** is the likelihood that the buffer runs empty (underflow) or cannot accommodate any additional packets (overflow) within a specific TDP or TOP.

**Buffer size** limits the number of bytes and packets that can be stored in the buffer, and is denoted by $B$.

### 3.2 Buffer Occupancy

Now, we study buffer occupancy. The arrival rate (namely window size per *round*) at round $k$ of $TDP_i$ is $\lambda_{i,k}$ packets/RTT, and $\hat{\lambda}$ packets are drained at round $k$ of $TDP_i$. We can get $\lambda_{i,k}$,

$$\lambda_{i,k} = W_{i-1}/2 + k - 1 \tag{10}$$

For simplicity, both the playout rate and the encoding rate are $\hat{\lambda}$ and constant. The playout buffer occupancy at (right after) round $k$ of $TDP_i$ is $q_{i,k}$ given by

$$\begin{aligned} q_{i,k} &= q_{i,k-1} + \lambda_{i,k} - \hat{\lambda} = q_{i,0} + \sum_{n=1}^{k}(\lambda_{i,n} - \hat{\lambda}) \\ &= q_{i,0} + \frac{k^2}{2} + \frac{(W_{i-1} - 2\hat{\lambda} - 1)k}{2} \end{aligned} \tag{11}$$

where $q_{i,0}$ is the playout buffer occupancy at the beginning of TDP $i$, namely the buffer occupancy before round 1 starts. Moreover, $k$ ranges from $0, 1, 2, \ldots, X_i$, where $X_i$ is the round number when the $i$th triple duplicate event is detected. We denote round 0 or $k = 0$ as the beginning of TDP $i$ before round 1 starts. The playout buffer occupancy at round 0 of TDP $i$ is given as follows,

$$q_{i,0} = q_{1,0} + \sum_{m=0}^{i-1}\sum_{k=1}^{X_m}[\lambda_{m,k} - \hat{\lambda}] \approx q_{1,0} + i * \bar{k}(\bar{\lambda} - \hat{\lambda}) \tag{12}$$

where $i \to \infty$, $\bar{\lambda}$ is the average throughput of TCP and $\bar{k}$ is the average rounds of a TDP. Therefore, the playback buffer occupancy $q_{i,k}$ of $TDP_i$ is

$$q_{i,k} \approx q_{1,0} + i * \bar{k}(\bar{\lambda} - \hat{\lambda}) + \frac{k^2}{2} + \frac{(W_{i-1} - 2\hat{\lambda} - 1)k}{2} \tag{13}$$

There exist the following options with respect to the time of maximum buffer occupancy $q_{max}$ and the minimum buffer occupancy $q_{min}$ during $TDP_i$ (see Figure 1):

$$\frac{\partial q_{i,k}}{\partial k} = 0, \text{or } k = 0, \text{or } k = X_i \tag{14}$$

On the one hand, there is a buffer overflow during $TDP_i$ when the maximum buffer occupancy of $TDP_i$ reaches the buffer size $B$, namely $q_{max} \geq B$. On the other hand, there is an underflow during $TDP_i$ when the minimum buffer occupancy is 0, namely $q_{min} \leq 0$. We have

$$P_u = P\{q_{min} \leq 0\} \tag{15}$$

$$P_o = P\{q_{max} \geq B\} \tag{16}$$

Moreover, the window sizes and consequently also its bounds during a TOP are 1 and the successful arrival is 0. All packets are lost in a TOP, and no packet is successfully delivered to the receiver buffer. The playout buffer occupancy during TOP is monotonously decreasing and given by

$$q_{i,k} = q_{i,k-1} - \hat{\lambda} = q_{i,0} - k * \hat{\lambda} \tag{17}$$

As a receiving node is either in a TDP or a TOP, the buffer underflow probability at time $t$ is defined by the sum of conditional probabilities, such that,

$$P\{q_{min} \leq 0\} = P\{q_{min} \leq 0 | t \in TDP\} P\{t \in TDP\} + P\{q_{min} \leq 0 | t \in TOP\} P\{t \in TOP\} \quad (18)$$

Similarly, the overflow probability for a given buffer size $B$ is,

$$P\{q_{max} \geq B\} = P\{q_{max} \geq B | t \in TDP\} P\{t \in TDP\} + P\{q_{max} \geq B | t \in TOP\} P\{t \in TOP\} \quad (19)$$

We will derive the solutions for the likelihoods of buffer overflow and underflow events for a given buffer size and initial buffering delay in Sec. 3.3 by using the distribution of TCP window bounds in Sec. 2.

### 3.3 TCP Streaming Buffer Performance

In this section, we study the buffer underflow probability and overflow probability given the buffer size, the initial buffering delay, and TCP window bounds in TCP streaming. We show that a closed form of TCP congestion window bounds is useful for TCP streaming applications to find the right buffer sizes and initial buffering delays given the desired probabilities of buffer overflow and underflow.

Due to space limitations, we consider only one TCP streaming application where the playout rate matches the TCP average sending rate (average window size), namely $\bar{\lambda} = \hat{\lambda}$ and Case 1 in Figure 1. Our analysis easily extends to the other cases in Figure 1 and general TCP streaming applications. For more details we refer to our technical report [15].

In this application , we assume $\frac{W_i}{2} < \hat{\lambda} < W_i$ and $P\{t \in case1\} = F(2\hat{\lambda}) - F(\hat{\lambda}) \to 1$. Thus $P\{t \in TOP\} \to 0$. Then, the minimum buffer occupancy $q_{min}$ of a $TDP_i$ is at round $k$ where $\frac{\partial q_{i,k}}{\partial k} = 0$. We get $k = \hat{\lambda} - \frac{W_{i-1}}{2} + 1$ and substitute into Eq.(13).

The **underflow probability** of the buffer of $TDP_i$ is the underflow probability of $q_{min}$ at $k = \hat{\lambda} - \frac{W_{i-1}}{2} + 1$,

$$\begin{aligned} P_u &= P\{q_{min} \leq 0\} \\ &= P\{q_{i,0} - \frac{W_{i-1}^2}{8} + \frac{2\hat{\lambda}+1}{4} W_{i-1} - \frac{\hat{\lambda}^2 + \hat{\lambda}}{2} \leq 0\} \\ &= P\{W_{i-1} \leq (2\hat{\lambda}+1) - \sqrt{8q_{i,0}+1}\} \end{aligned}$$

When the playout rate matches the TCP average sending rate, $q_{i,0}$ approximately equals $q_{1,0}$ in Eq.(12). Substituting the CDF of TCP window bounds, we finally obtain

$$P_u = P\{q_{min} \leq 0\} = F(2\hat{\lambda} + 1 - \sqrt{8q_{1,0}+1}) \quad (20)$$

where

$$q_{1,0} \approx q_0 + \frac{1}{18p} + \frac{0.2}{\sqrt{p}}$$

The **overflow probability** of $TDP_i$, namely the overflow probability of $q_{max}$ at round $k = X_i$ (equivalently, $k = 0$),

$$\begin{aligned} P_o &= P\{q_{max} > B\} \\ &= P\{q_{min} + \frac{W_i^2}{2} + W_i(1/2 - \hat{\lambda}) + \hat{\lambda}^2 - \hat{\lambda} > B\} \\ &= P\{W_i > \hat{\lambda} - 1/2 + \sqrt{2(B - q_{min}) + 1/4}\} \\ &= 1 - F(\hat{\lambda} - 1/2 + \sqrt{2(B - q_{1,min}) + 1/4}) \end{aligned} \tag{21}$$

where $q_{i,min}$ approximately equals $q_{1,min}$ in Eq.(12) when the playout rate matches the TCP average sending rate. And,

$$q_{1,min} \approx q_0 - \frac{1}{36p} + \frac{0.2}{\sqrt{p}}$$

Knowing desired underflow/overflow probabilities we can now determine appropriate buffer sizes and initial buffering delays by using the distribution of TCP window bounds.

Given an underflow probability for a TCP streaming application with rate $\hat{\lambda}$, **the initial buffering occupancy** $q_0$ and **buffering delay** $D$ is

$$q_0 = \frac{(2\hat{\lambda} + 1 - F^{-1}(P_u))^2 - 1}{8} - \frac{1}{18p} - \frac{0.2}{\sqrt{p}} \tag{22}$$

$$D = \frac{q_0}{\hat{\lambda}} \tag{23}$$

Given an overflow probability for a TCP streaming application with rate $\hat{\lambda}$ and given the initial buffering occupancy, the needed **buffer size** $B$ is

$$B = \frac{(1/2 - \hat{\lambda} - F^{-1}(1 - P_o))^2 - 0.25}{2} + q_0 - \frac{1}{36p} + \frac{0.2}{\sqrt{p}} \tag{24}$$

## 4 Evaluation

Based on simulations and experiments, we now verify our proposed TCP window bounds model. In addition, we study how well our analytical solutions estimate underflow and overflow probabilities in TCP streaming scenarios.

### 4.1 TCP Window Model

Our goal is to compare the analytical solution for the distribution of TCP window bounds against congestion window sizes observed in simulations. To this end, we rely on ns-2 simulations [1] for a dumbbell topology with one bottleneck link. We establish TCP connections using this bottleneck link and leverage the ns-2 error model to

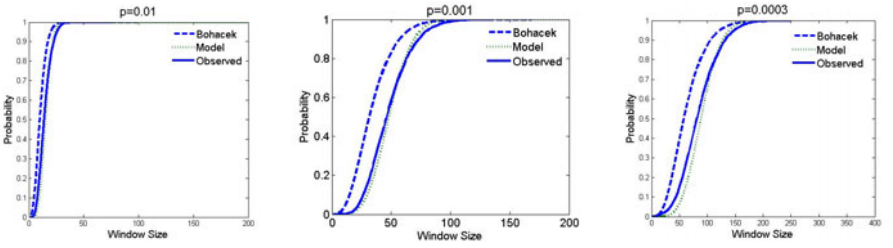



**Fig. 3.** CDF of window size and the upper bounds

generate random packet drops at the bottleneck. Stationary packet loss probabilities are supposed to reproduce packet losses on high speed links with high statistical multiplexing [4]. To observe a large sample of window sizes, we run each simulation instance for one million seconds.

Figure 3 presents the CDF of the observed upper bounds of TCP congestion window sizes for three different loss rates, namely $p = 0.01, 0.001, 0.0003$. In addition, it shows the results for our analytical solution (see Eq.(6)) and Bohacek's general distribution for all window sizes that is based on stochastic differential equations [3]. To the best of our knowledge, Bohacek's distribution is the only solution for TCP window sizes in the past, in particular, for all TCP window sizes. Indeed, we find that the curve showing the observed bounds for TCP congestion window sizes is very close to the Gamma distribution, i.e., our analytical solution. Crosschecking with different TCP variants such as Reno, NewReno, or SACK does not change this picture and creates almost identical results. Finally, there is only a very small likelihood in our experiments for timeouts where the window size would be fixed to one. Overall, the simulations suggest that our TCP model for bounds on congestion window sizes is reasonably accurate.

### 4.2 Buffer for TCP Streaming

Now, we study the degree to which our analytical solutions can estimate underflow and overflow probabilities in TCP streaming scenarios. Due to space limitations, we consider only one TCP streaming application. For more details we refer to our technical report [15].

We rely on the same topology as in Section 4.1. To study the behavior of a streaming application in the presence of packet losses, we generate 19 FTP flows in addition to our TCP streaming flow, and run the flows for a time period of $1,000$ seconds. All flows use the common bottleneck link that is configured to have 20 Mbps capacity and 100 ms delay. In such a setup we find that the packet loss rate $p$ for the streaming application is stationary and is around $0.25\%$ (RTT: $0.229367$ ms). Moreover, the correlation between subsequent losses is small due to the high multiplexing on the bottleneck link. We run the simulation 20 times using random seeds. With respect to access links, we assume a capacity of 100 Mbps and a delay of 1 ms.

For the TCP streaming application, we rely on a coding/playout rate of 1 Mbps, use an initial TCP window size of 20 packets, and set the packet size to 1200 bytes. Before we start decoding packets, we fill half of the buffer, namely $q_0 = \frac{B}{2}$. Thus, we

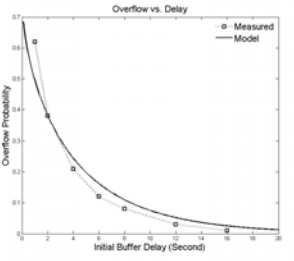



**Fig. 4.** Overflow probabilities for various buffer sizes and delays ($q_0 = \frac{B}{2}$, $\sigma_{est} = 0.053$)

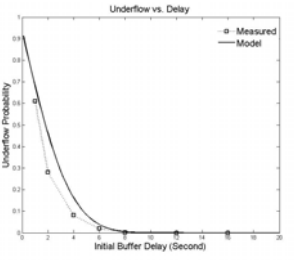



**Fig. 5.** Underflow probabilities for various delays ($\sigma_{est} = 0.080$)

adjust both the initial buffering delay and the buffer size at the same time. Overall, our goal is to study the events of overflow and underflow and their likelihoods for various buffer sizes and initial buffering delays ($q_0/\hat{\lambda}$). To this end, we compare the observed frequency of overflow and underflow events with the probabilities that we expect based on our analytical model.

Indeed, Fig. 4 and Fig. 5 show that the measured results are close to our model and analytical results. Small values for the standard error of the estimate $\sigma_{est}$ confirm this finding, underlining the expressiveness of our analytical framework. From Fig. 4 and Fig. 5 we also see that overflow events are more likely to occur than underflow events if the initial buffering delay $q_0$ is set to half of the buffer size in cases where available TCP bandwidth exactly matches the coding rate. This is somewhat counter-intuitive. One may expect the overflow and underflow probabilities to be roughly the same in such cases. However, we explain this by the fact that at the time when playback starts, the window sizes are likely to be closer to the minimum buffer occupancy than to the maximum buffer occupancy of a $TDP$. After all, buffer occupancy is non-linear with respect to the round number $k$ in a $TDP$.

## 5 Conclusion

In this paper, we have developed an analytical framework that describes the dimensioning of appropriate buffers for TCP streaming. To this end, we first infer a closed form for the distribution of TCP window bounds, relying on the Gamma distribution. We evaluate our solution for dimensioning buffers in TCP streaming. As confirmed by experiments, our proposed analytical framework allows to estimate the frequency of buffer overflow or underflow events if buffer sizes and initial buffering delays are known parameters in the proposed TCP streaming model, or conversely, to dimension the buffer size and the initial buffering delay appropriately. With respect to future work, we plan to evaluate our proposed model for TCP streaming applications in real environments, and to apply our key ideas to other TCP variants such as TCP CUBIC.

**Acknowledgements.** The first author is supported in part by Swiss National Science Foundation under grant No.200020-121753, National Science Foundation of China under grant No.60970127 and New Century Excellent Talents in Chinese University under grant No.NCET-09-0709.

## References

1. ns-2, www.isi.edu/nsnam/ns/
2. Altman, E., Avrachenkov, K., Barakat, C.: A stochastic model of tcp/ip with stationary random losses. IEEE/ACM Trans. Netw. 13, 356–369 (2005)
3. Bohacek, S., Shah, K.: Tcp throughput and timeout - steady state and time-varying dynamics. In: Proceedings of GLOBECOM 2004, pp. 1334–1340 (December 2004)
4. Bolot, J.-C.: End-to-end packet delay and loss behavior in the internet. In: Conference Proceedings on Communications Architectures, Protocols and Applications, SIGCOMM 1993, pp. 289–298. ACM, New York (1993)
5. Floyd, S., Handley, M., Padhye, J., Widmer, J.: Equation-based congestion control for unicast applications. In: Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, SIGCOMM 2000, pp. 43–56. ACM, New York (2000)
6. Guo, L., Tan, E., Chen, S., Xiao, Z., Spatscheck, O., Zhang, X.: Delving into internet streaming media delivery: a quality and resource utilization perspective. In: Proceedings of IMC 2006, pp. 217–230. ACM, New York (2006)
7. Hogg, R.V., Craig, A.T.: Introduction to Mathematical Statistics, 4th edn. Macmillan, New York (1978)
8. Mathis, M., Semke, J., Mahdavi, J., Ott, T.: The macroscopic behavior of the tcp congestion avoidance algorithm. SIGCOMM Comput. Commun. Rev. 27, 67–82 (1997)
9. Misra, V., Gong, W.B., Towsley, D.: Stochastic differential equation modeling and analysis of tcp windowsize behavior. In: Proceedings of IFIP WG 7.3 Performance (November 1999)
10. Padhye, J., Firoiu, V., Towsley, D.F., Kurose, J.F.: Modeling tcp reno performance: a simple model and its empirical validation. IEEE/ACM Trans. Netw. 8, 133–145 (2000)
11. Saxena, M., Sharan, U., Fahmy, S.: Analyzing video services in web 2.0: a global perspective. In: Proceedings of NOSSDAV 2008, pp. 39–44. ACM, New York (2008)
12. Stockhammer, T.: Dynamic adaptive streaming over http: standards and design principles. In: Proceedings of MMSys 2011, pp. 133–144. ACM, New York (2011)
13. Wang, B., Kurose, J., Shenoy, P., Towsley, D.: Multimedia streaming via tcp: An analytic performance study. ACM Trans. Multimedia Comput. Commun. Appl. 4, 16:1–16:22 (2008)
14. Wu, C.C., Chen, K.T., Huang, C.Y., Lei, C.L.: An empirical evaluation of voip playout buffer dimensioning in skype, google talk, and msn messenger. In: Proceedings of NOSSDAV 2009, pp. 97–102. ACM, New York (2009)
15. Yan, J., Muehlbauer, W., Plattner, B.: Analytical framework for streaming over tcp. ETH Zurich (2010), ftp://ftp.tik.ee.ethz.ch/pub/publications/TIK-Report-333.pdf
16. Yan, J., Katrinis, K., May, M., Plattner, B.: Media- and tcp-friendly congestion control for scalable video streams. IEEE Transactions on Multimedia 8(2), 196–206 (2006)

# Characterising University WLANs within Eduroam Context

Marangaze Munhepe Mulhanga, Solange Rito Lima, and Paulo Carvalho

University of Minho, Department of Informatics, 4710-057 Braga, Portugal
solange@di.uminho.pt

**Abstract.** The *eduroam initiative* is assuming an ever growing relevance in providing a secure, worldwide roaming access within the university WLAN context. Although several studies have focused on educational WLAN traffic characterisation, the increasing variety of devices, mobility scenarios and user applications, motivate assessing the effective use of eduroam in order to sustain consistent network planning and deployment. Based on recent WLAN traffic traces collected at the University of Minho (Portugal) and University of Vigo (Spain), the present work contributes for identifying and characterising patterns of user behaviour regarding, for instance, the location and activity sector of users. The results of data analysis quantify the impact of network access location on the number of associated users, on the number and duration of sessions and corresponding traffic volumes. The results also illustrate to what extent users take advantage of mobility in the WLAN. Complementing the analysis on a monthly basis, a fine grain study of WLAN traffic is provided through the identification of users' behaviour and patterns in small timescales.

**Keywords:** Network Traffic Characterisation, WLAN, Eduroam.

## 1 Introduction

In recent years, there has been a significant growth of Wireless Local Area Networks (WLANs), with increasing influence on people's day life and productivity. The low to moderate cost of personal devices and the need for easy and ubiquitous access to information are factors that influence the growing use of wireless technology.

Network Access Points (APs) tend to be critical points in WLANs due to multiple aspects such as user mobility, traffic dynamics, location and density of devices, which impact on network performance. From the user perspective, WLAN are expected to provide seamless connectivity, supporting multiple types of applications and services, such as VoIP, video conferencing, Web Services, with distinct quality of service (QoS) requirements. Understanding traffic characteristics and the usage of network resources is an essential step to assure quality of service (QoS) and improve quality of experience (QoE) of end users when accessing network services.

In this context, this paper presents a traffic analysis and characterisation study involving WLANs of two European Universities - University of Minho (Portugal) and University of Vigo (Spain), within *eduroam (Education Roaming) initiative* [1]. The analysis was carried out based on real traffic traces gathered between April and June

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 382–394, 2011.

of 2010, which corresponds to a typical academic term. The study assesses several network usage metrics related to AP utilisation, session characterisation, access location influence, and user mobility patterns. This analysis, when compared with other case studies, aims to provide guidelines for planning future WLANs deployment.

This paper is organised as follows: related work reporting university WLAN studies is discussed in Section 2; the University campi and corresponding WLAN infrastructure are briefly described in Section 3; the process of data collection is presented in Section 4; the results of traffic analysis are discussed in Section 5; and the main conclusions are included in Section 6.

## 2 Related Work

Traffic analysis and characterisation has been the matter of relevant and extensive research over the years. However, within the University WLANs context, there are several works addressing this topic.

Tang and Baker [2], based on data collected during twelve weeks in one building of Stanford University, aimed to understand the behaviour of WLAN users, answering questions regarding the benefits of mobility, and the volume and characteristics of traffic involved. QoS metrics, such as delay and bandwidth were also measured.

Kotz and Essien study [3], reporting the analysis of data traffic collected during eleven weeks on the campus of Dartmouth University, complemented the study in [2], extending the analysis to all buildings on campus. Later on, the work reported in [4] states that the applications used over the already mature WLAN changed dramatically.

Balazinska and Castro [5] analysed a four week trace gathered at a corporate 802.11b WLAN, encompassing three buildings hosting computer science and electrical engineering research groups. The study focuses on population characteristics, load distribution across APs, user activity levels, and user mobility. This study found that: users' average transfer rates follow a power law; load is unevenly distributed across access points and is more influenced by which users are present than by the number of users; the APs location plays an important role in the aggregate load observed; and users spend a large fraction of their time at a single location.

The motivation for the study carried out by Schwab and Bunt [6] was to understand usage patterns in the University of Saskatchewan campus, comprising a small number of access points (18) strategically placed, in order to plan its expansion. For this purpose, usage data was collected over the period of one week in January 2003, recording address and protocol information for every packet sent and received on the wireless network. The trace was analysed to answer questions about where, when, how much, and for what the wireless network was being used.

Papadopouli et al [7] investigated roaming activity at aggregated level in the University of North Carolina infrastructure. Based on syslog data from three monitoring periods (between October 2004 and April 2005). The authors identify the regions with high roaming activity and derive topological models of the University infrastructure, involving 488 APs. The study also discusses the impact of the spatial and temporal growth of the wireless infrastructure, discussing the nonlinear correlation between the number of roaming events between two APs and their geographic distance.

Kumar et al [8] study classifies users into social groups and investigates the WLAN usage behaviour of these groups in the USC University campus (MobiLib). Based on a month long WLAN trace, the authors analyse aspects such as the differences on the average session duration for male and female users across the campus.

Kim and Helmy [9] study, based on traffic traces collected from Dartmouth University WLAN during 4 years (between 2001 to 2004 and 2005 to 2006), try to understand how changes in wireless devices and network affects WLAN users, and influence location prediction. The study highlights the drastic change in the number of APs and growing of mobile user community, and defines the number of distinct APs a user has visited as mobility metric.

The analysis of the above studies provides useful insights on WLANs usage patterns and important guidelines for further study on this research topic. However, the mentioned works, report results on distinct aspects of WLANs usage based on traffic traces collected before 2006. Facing the constant evolution of wireless devices, the increasing number of users, the variety of available applications and services, and human behaviour regarding wireless technologies, it is essential to keep an up-to-date analysis of today's university wireless networks usage, in particular, within the eduroam context. These aspects will provide useful feedback regarding WLAN planning and future deployments. The present study is a further step in this direction, taking as case study the analysis of the WLANs from University of Minho and University of Vigo using traffic traces collected between May and June of 2010.

## 3 Case Study: Eduroam at University of Minho

The University of Minho (UMinho), located in the north of Portugal, was founded in 1973 and started its academic activity in 1975. Currently, with a population of nearly 15,000 students, 1,200 teachers, and 600 technical and administrative staff, it is one of the biggest Portuguese universities. The academic and scientific activities at UMinho are developed in two campi: the campus of Gualtar in Braga and the campus of Azurém in Guimarães. The students' accommodation buildings have capacity to accommodate 1400 students, around 60% in Braga and 40% in Guimarães.

Both campi participate in the eduroam initiative, which provides wireless network access for research and education to the university population and visitors. The WLAN infrastructure comprises a total of 429 APs, 310 located in Braga and 119 in Guimarães. The WLAN technology used in both campi is based on the standards IEEE 802.11b and 802.11g and, more recently, 802.11n. 802.11n is deployed in strategic locations, such as libraries, due to its advantages both in terms of transmission data rates and spatial coverage [10,11]. IP addresses are assigned to wireless devices via DHCP. Authentication is performed by a Radius server.

A partial view of the UMinho network infrastructure is illustrated in Figure 1. The core of network operation is located in Gualtar, where the main network services are assured to users inside and outside the campus (e.g., in residences, student associations, Azurém campus), providing a 10Gbps access to the Internet (through *Fundação para a Computação Científica Nacional* - FCCN). A 768Mbps link interconnects Gualtar

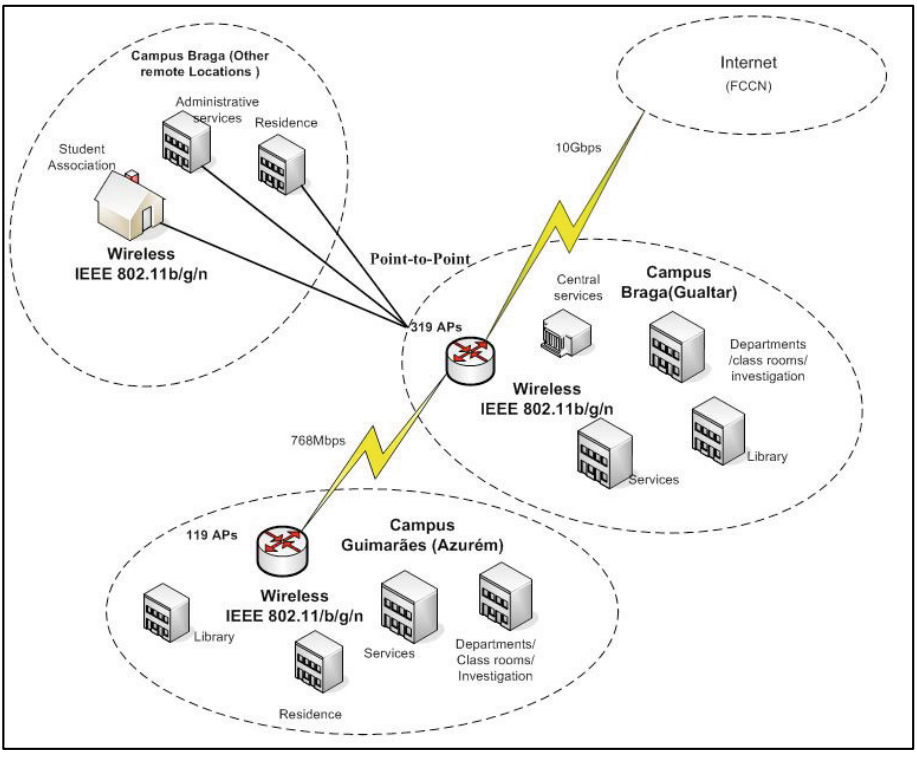



**Fig. 1.** Partial view of UMinho network infrastructure

and Azurém campi. The Communications Services headquarters, responsible for the operation and management of the whole network infrastructure and services, are located in Gualtar.

The University of Vigo (UVigo), located in the autonomous community of Galicia, in Spain, was established in 1990. Currently, it has about 30,000 students and approximately 1,800 teachers. The UVigo is located in three cities: Vigo, Ourense and Pontevedra. Each campus has several research centers, schools, residences, sport centers and other service buildings (libraries, administrative offices, canteens, etc.). The WLAN infrastructure is similar to the one at UMinho, diverging on the number of network resources. For instance, Vigo campus has 183 APs distributed across 18 buildings.

## 4 Data Collection: Challenges and Good Practices

Collecting data in large scale WLANs is in itself a challenge due to aspects such as dense topologies and mobility of users. Taking UMinho eduroam network as an example, the WLAN infrastructure is complex involving hundreds of APs, switches and routers, thousands of users distributed across different buildings and cities, using private or public IP addressing. Therefore, managing and monitoring network operation and collecting traces of WLAN usage is not a straightforward task.

### 4.1 Data Collection Strategies and Tools

This section provides information on strategies and tools used to assist the data collection process. This process involved the articulation of several technologies, namely:

- *SQL database*: for each AP, the information of authenticated users is stored on an SQL database. This information is directly accessed from a Radius server, which maintains data of eduroam authenticated users. These data include the MAC addresses from wireless devices of authenticated users, and corresponding events, such as association, disassociation, reassociation, roaming, handoff, authentication or deauthentication. Besides, this database allows accounting of multiple records, including the number and duration of user associations, initial and final APs involved in a session, and the traffic volume in packets and bytes for AP. Through an SQL server it is possible to obtain the WLAN traffic history.
- *DHCP logs*: private or public IP addresses are dynamically assigned to wireless devices on campus by a DHCP server. The DHCP logs contain IP addressing assignments to MAC addresses, including a timestamp of that occurrence.
- *SNMP-based tool*: a proprietary tool, developed by the Communications Services, based on PERL and using SNMP primitives, allows monitoring APs at regular intervals (configured for poling each five minutes). This tool performs traffic collection of associated users regardless its authentication status. Network history and statistics can be obtained querying the SQL server. Data can be visualised through a PERL CGI and HTTP-Apache server.
- *Port Mirroring*: for analysis at protocol level, traffic was captured using `tcpdump` via port mirroring. This traffic gathering process was performed at main router providing connectivity between Gualtar and Azurém campi, resorting to Colasoft's Capsa Network Analyzer for traffic analysis.
- *Data Confidentiality*: Processing traffic traces poses confidentiality concerns. It is known that packet header fields (e.g. IP and MAC addresses) or payload allow to access information about network structure, network services location, and user activities. Therefore, privacy policies must be enforced to avoid open processing of network traffic. The use of anonymisation tools (http://www.caida.org/tools/taxonomy/anonymization.xml) to pre-process network traces, e.g. replacing consistently MAC addresses, allowed to perform WLAN analysis without compromising data confidentiality.

At UVigo, it was used CiscoWorks Wireless LAN Solution Engine (WLSE), which allows managing Cisco aironet APs in WLANs.

Data collection was conducted in two different time periods: the first one took place at UMinho during April and the second one at UVigo, during May/June. The results presented in this paper are mainly focused on eduroam infrastructure at UMinho.

## 5 Results of Data Analysis

A first step toward WLANs characterisation is to group and classify APs according to their location. University buildings at UMinho are usually associated with a main activity. For instance, the Department of Civil Engineering (DEC) encompasses teacher offices, laboratories and classrooms dedicated to Civil Engineering academic and research

activities. APs in this building can be grouped and classified as belonging to DEC. Using this first classification criterion, 30 distinct locations were identified. Additionally, APs were also grouped in type of activity sector, resulting in six distinct sectors, namely Social, Residential, Services, Libraries, Research, and Academic. A third classification criterion was based on the number of distinct users registered in each AP, which leads to the definition of five distinct groups based on the level of APs usage.

The following sections detail the results of characterising and analysing traces from the Azurém Campus, regarding: (i) the number of associations to each AP and corresponding location; (ii) the traffic volumes and the top 10 locations contributing to this parameter; (iii) the average duration of user sessions; (iv) the degree of users mobility; and (v) a fine grain analysis of per day traffic.

### 5.1 Analysis of Associated Users and Location

Concerning the analysis of WLAN data at UMinho, at first, we have identified the number of different mobile users associated with each of 119 APs (during April). The obtained values ranged from 928 different users on the busiest AP to 2 users on the more underutilised AP. Attending to the asymmetry of WLAN users in using the APs across the campus, and to provide a finer analysis of the utilisation patterns, five distinct activity groups were considered. As illustrated below, Group 1 comprises the total of APs with 500 or more users, illustrating zones of higher user preference:

```
Group 1: APs with number of users ≥ 500
Group 2: APs with 300 ≤ number of users < 500
Group 3: APs with 100 ≤ number of users < 300
Group 4: APs with 50 ≤ number of users < 100
Group 5: APs with 0 ≤ number of users < 50
```

Considering these intervals, we found that AP utilisation follows approximately a normal distribution centered on Group 3 with 38% of AP associations. A total of 13.4% APs handle more than 500 users, whereas around 16% handle less that 50 users. This analysis also indicates that 56% of APs on campus support between 100 and 500 users. Figure 2a) shows the values recorded for the different intervals.

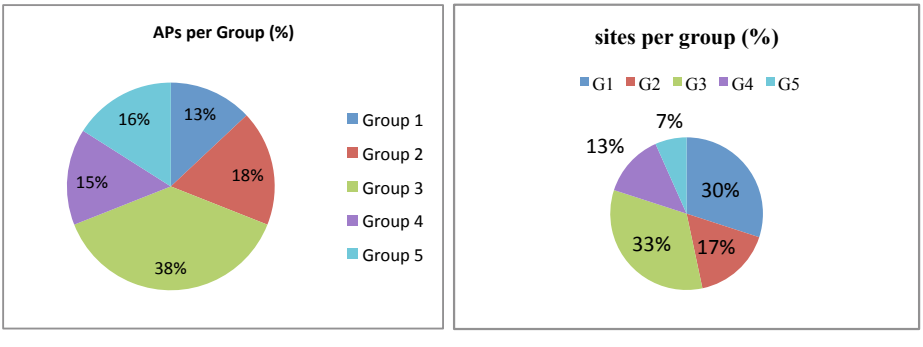



**Fig. 2.** a) Distribution of APs per group; b) Distribution of sites per group (%)

Taking into account the results above, for each group, the spatial distribution of APs on campus was assessed. Figure 2b) illustrates the obtained results when considering 30 spatial areas (sites). As shown, the busiest APs cover about 30% of the total area, while less utilised APs (Groups 4 and 5) around 30%.

### 5.2 Analysis of Traffic Volume

The analysis of WLAN traffic volume (eduroam) on campus was carried out considering the APs grouped in six classes, according to their location:

(i) Social: in bars, canteens and sports facilities;
(ii) Residential: in student accommodation premises;
(iii) Services: in administrative, technical and student support services;
(iv) Library: in libraries;
(v) Research: in research laboratories;
(vi) Academic: in departments and schools.

Figure 3 shows the distribution of the volume of inbound and outbound traffic by type of location, with a clear dominance of inbound traffic. As illustrated, the residential and academic sectors are responsible for most of inbound traffic, with nearly 80% of the total. Regarding the volume of outbound traffic, residential and academic sectors are again the major contributors to the overall traffic load. In absolute terms, for the period under study, the traffic generated by eduroam users was 620GB and 5120GB of outbound and inbound traffic, respectively. In order to detail the analysis of traffic volumes, accounting data was processed to determine the location of most active APs.

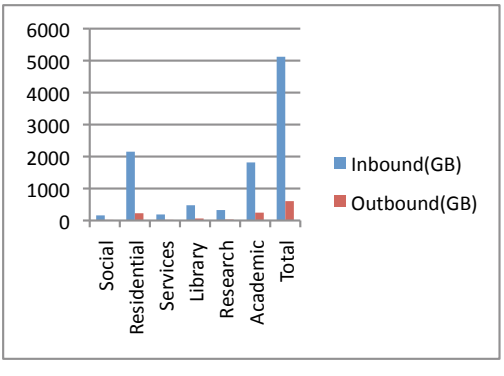



**Fig. 3.** Distribution of traffic volume by activity sector

### 5.3 Traffic Volume: Top 10 APs

The Top 10 ranking identifies the 10 APs exhibiting higher traffic volume. These APs represent approximately 2.3 TB of inbound traffic, i.e., 45% of the total traffic. As shown in Figure 4, in the Top 10 group, 50% of APs belong to Residential sector and

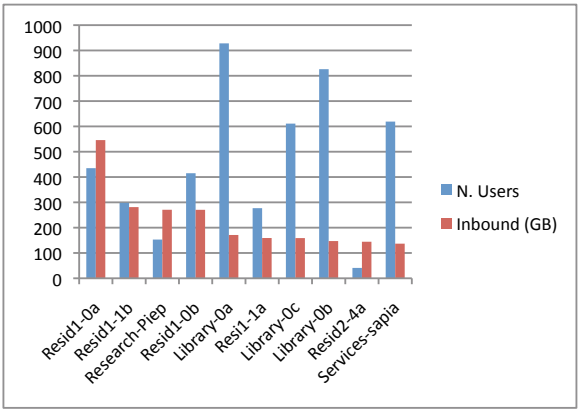



**Fig. 4.** Top10 - Traffic volume per AP

30% to Library sector. The Library APs are also part of group 1, i.e., group with the largest number of associated users. This Top 10 also includes APs belonging to Services and Research.

### 5.4 Analysis of Sessions

An important aspect of users' pattern behaviour is the time users remain associated to each AP. This time analysis, based on the duration of user sessions, resorts to aggregated data comprising the total number of sessions and their average duration. Crossing this information with the number of users registered per AP, it is possible to evaluate the average time of each user session. Based on this calculation, it was observed that the AP with the highest monthly associated time per user was approximately 58 hours, resulting from the sum of its individual sessions.

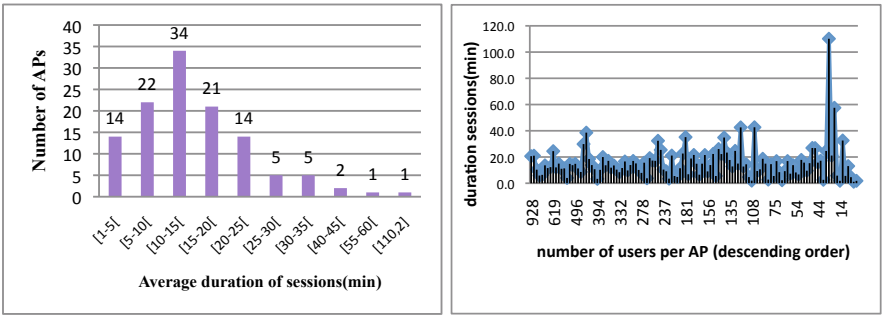



**Fig. 5.** a) Number of APs per average time duration of sessions; b) Distribution of duration of sessions

The average duration of sessions in APs varies between 1 and 110 minutes. Figure 5a) shows a histogram representing the number of APs for distinct session time intervals. As illustrated, five representative time intervals were identified, showing that, from a total of 119 APs, 34 (28%) handle sessions with average duration from 10 to 15 minutes. Based on this information it was found that: (i) APs with a low number of users (Groups 4 and 5) support a large number and duration of sessions. These observations correspond mainly to APs associated with the Residential sector and other APs in locations of reduced mobility; (ii) the average length of sessions does not exhibit major variations, especially in groups 1 and 2. The largest deviation occurs in all AP with few users. There are sessions of high duration depending on the AP location (e.g. home or research center) and other APs with many users and sessions of reduced duration. Figure 5b) shows the distribution of the average duration of sessions per AP in descending order of number of users.

### 5.5 Daily Analysis

A fine grain analysis of WLAN traffic is provided through the identification of users' behaviour and patterns in small timescales. Taking a weekly traffic trace (from April 2010) as an example, traffic data was analysed on a daily basis, hour-by-hour. This section reports the results obtained regarding: (i) the variation on number of users and on number of sessions; (ii) the aggregated traffic volume and packet size characteristics, for time periods of low, medium and high traffic load.

**Users and Sessions Accounting.** As illustrated in Figure 6a), the variation of the number of users during working days follows a pattern with a daily peak at 11/12 and 16 pm, with about 400-550 wireless users. The low activity period occurs at night, with a minimum number of users at 4 am (about 15-20 users), mainly from student residences. The variation on the number of sessions presents a similar behaviour. As illustrated in Figure 6b), during busy hours, the average number of sessions per user is around 4 (about 2350 sessions for 550 users), increasing to 7 overnight (e.g., about 100 sessions for 15 users).

**Traffic Volume and Packet Length.** The analysis of inbound traffic volume between Gualtar-Azurém during working days exhibits two peak hours and three activity time periods. As shown, the peak hours occur at noon and 4pm; the time period with high traffic load occurs between 9am to 6pm, reaching an inbound rate of 100Mbps; from 6pm to midnight this rate decreases to 40Mbps, keeping this level up to 9am. Regarding the link capacity between the two campi, WLAN traffic volumes represent 4% to 10% of the available capacity.

Packet size can be an important aspect on the performance of network equipment, with special impact on queuing behaviour. The analysis at packet level within the WLAN reveals two main groups of packet length: centred on 64 Bytes and 1518 Bytes, as illustrated in Figure 7. While smaller packets represent 35% of the total number of packets and 2,76% of the total number of bytes transmitted, larger packets represents 40% of the total packet count and more than 75% of the bytes transmitted.

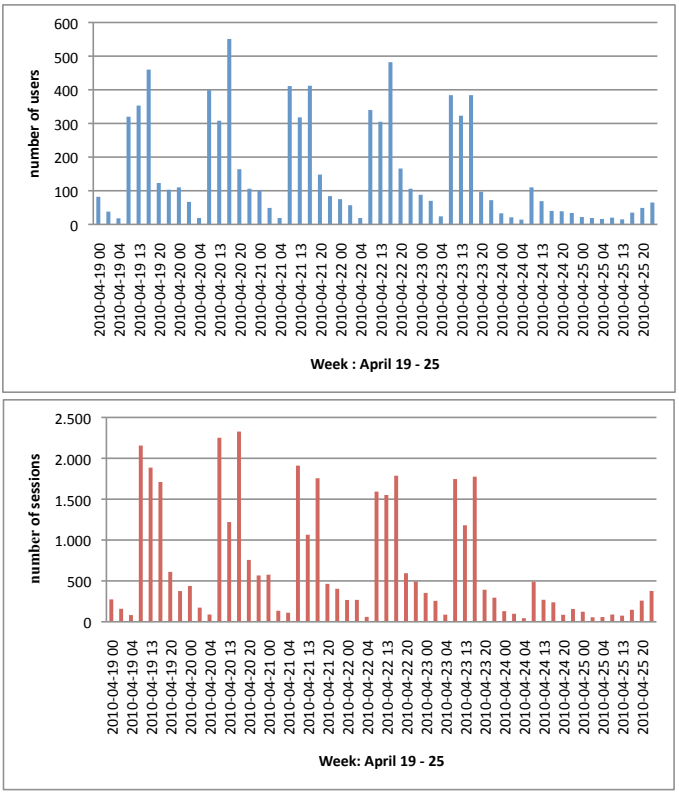



**Fig. 6.** a) Variation on the number of users; b) Variation on the number of sessions

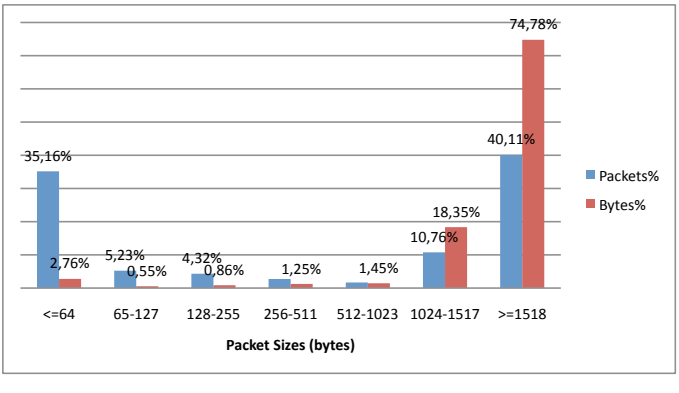



**Fig. 7.** Packet sizes and bytes (%)

### 5.6 Mobility of Users

Allowing user mobility is considered one of the main advantages of WLANs. On campus, users move to different locations, e.g. classrooms, canteens, accommodations, sport centers, meaning that mobile devices should necessarily go through the process of association with at least two APs.

To assess to what extent users take advantage of mobility in the WLAN, we identified (by MAC address) the users that presented high mobility patterns (during April and May). In this way, we analysed the number of associations from each user with different APs. The results show that most of authenticated users (90% of a total of 3480 users) exhibited effective mobility and only a reduced percentage (9%) used a single AP to access the WLAN.

Attending to the high percentage of users associated with two or more distinct APs, a more detailed analysis was carried out, identifying the number of users per mobility scenario, grouped in intervals of number of associated APs. As shown in the histogram

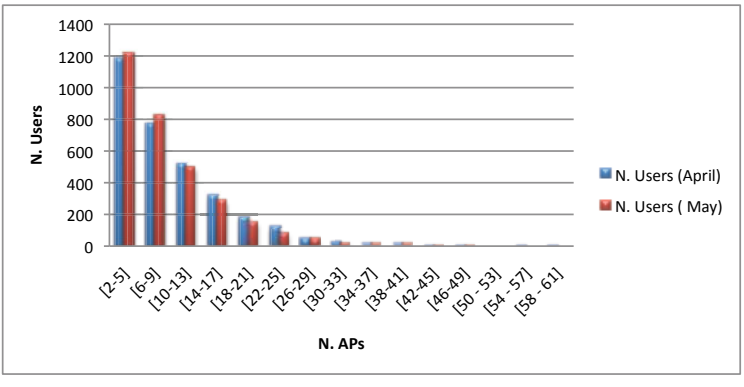



**Fig. 8.** Users mobility

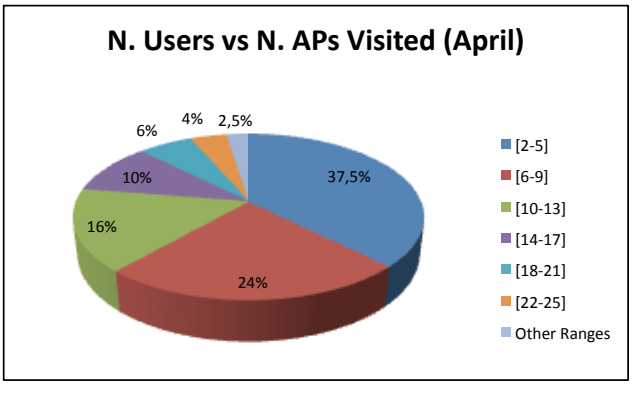



**Fig. 9.** Relative view of users mobility

of Figure 8, a significant number of users (around 37,5% of the total) visited from 2 to 5 APs. The tail of the histogram indicates that users may reach a maximum of [58,61] distinct APs. This means that, monthly, a large number of different locations on campus may be visited per a single user. Figure 9 provides a complementary view of the mobility degree expressing the relative values obtained in April. For users with the highest mobility indicators (corresponding to the slice "other ranges"), approximately 37% were identified as the same users in April and May.

### 5.7 UVigo WLAN Considerations

This section presents briefly the results from analysing WLAN traffic at UVigo. The study was focused on Vigo Campus, which encompasses 183 APs distributed across 18 buildings. As in UMinho, in order to identify the more relevant sites for analysis, the number of registered associations in the 183 APs during May/June was analysed. From the more representative sites regarding WLAN usage, it is clear that libraries, social areas and free spaces on campus are the preferred locations for students to access the wireless network. Although the quantification of results is clearly distinct due to the number of APs and users involved, the patterns of WLAN usage follow closely the ones reported in this paper for UMinho WLANs.

## 6 Conclusions

This paper has focused on the characterisation of University WLANs usage, within the eduroam initiative. The study, based on real traffic traces collected from UMinho and UVigo WLANs, contributes to understand to what extent users take advantage of this technology, within the academic context. The study has covered and assessed several network usage metrics related to AP utilisation, session characterisation, access location influence, and user mobility patterns. These metrics are relevant to better understand both the real utilisation of the WLAN infrastructure and the characteristics in terms of user behaviour.

Results from traffic analysis showed that the number of users associated with an AP depends heavily on the activity sector where those APs are located (e.g. libraries). The number and duration of sessions also varies depending on location and type of activity. In fact, the duration of sessions is more related to the activity area where APs are located than with the number of associated users. As regards the overall traffic volume, both residential and academic sectors are clearly the main contributors. The mobility analysis reveals that a significant number of users access the WLAN from a reduced number of APs in campus, exhibiting a stable behaviour in location preference. However, the mobility indicators also reveal that a considerable number of users use a large number of different locations on campus for accessing the network.

The current results, based on recent data and focused on user behaviour, constitute a useful feedback for planning, dimensioning and managing WLAN within eduroam context. Future work includes expending the characterisation of traffic on UMinho WLAN, including aspects such as the study of mobility patterns, protocols and applications per activity sector and user location.

**Acknowledgements.** The authors would like to thank the Communications Services of the University of Minho and University of Vigo for the technical support provided.

## References

1. eduroam: Eduroam Initiative (2003), http://www.eduroam.org
2. Tang, D., Baker, M.: Analysis of a local-area wireless network. In: MOBICOM, pp. 1–10 (2000)
3. Kotz, D., Essien, K.: Analysis of a Campus-wide Wireless Network. In: Proceedings of ACM Mobicom, pp. 107–118. ACM Press, New York (2002)
4. Henderson, T., Kotz, D., Abyzov, I.: The Changing Usage of a Mature Campus-Wide Wireless Network. In: MOBICOM 2004 (2004)
5. Balazinska, M., Castro, P.: Characterizing mobility and network usage in a corporate wireless local-area network. In: Proceedings of the 1st International Conference on Mobile Systems, Applications and Services, MobiSys 2003, pp. 303–316. ACM, New York (2003)
6. Schwab, D., Bunt, R.: Characterising the Use of a Campus Wireless Network. In: INFOCOM (2004)
7. Papadopouli, M., Moudatsos, M., Karaliopoulos, M.: Modeling Roaming in Large-scaleWireless Networks Using Real Measurements. In: Proceedings of the 2006 International Symposium on on World of Wireless, Mobile and Multimedia Networks, WOWMOM 2006, pp. 536–541. IEEE Computer Society, Washington, DC (2006)
8. Kumar, U., Yadav, N., Helmy, A.: Gender-based feature analysis in campus-wide WLANs. SIGMOBILE Mobile Computer Communications Review 12, 40–42 (2008)
9. Kim, J., Helmy, A.: Analyzing mobility evolution in WLAN users: how predictable are we? SIGMOBILE Mobile Computer Communications Review 14, 10–12 (2010)
10. Gast, M.: 802.11 Wireless Networks: The Definitive Guide, 2nd edn. O'Reilly, Sebastopol (2005)
11. IEEE: IEEE Std. 802.11-1999, Part 11: Wireless LAN Medium AccessControl (MAC) and Physical Layer (PHY) specifications, Reference number ISO/IEC 8802-11:1999(E) (1999)

# Internet Traffic Source Based on Hidden Markov Model

Joanna Domańska[1], Adam Domański[2], and Tadeusz Czachórski[1,2]

[1] Institute of Theoretical and Applied Informatics,
Polish Academy of Sciences,
Bałtycka 5, 44-100 Gliwice, Poland
{joanna,tadek}@iitis.gliwice.pl
[2] Institute of Informatics,
Silesian Technical University,
Akademicka 16, 44-100 Gliwice, Poland
adamd@polsl.pl

**Abstract.** This article shows how to use Hidden Markov Models to generate self-similar traffic. The well-known Bellcore traces are used as a training sequence to learn HMM model parameters. Performance of trained model are tested on the remaining portions of the sequences.Then we can use the HMM trained with the Bellcore data as the traffic source model.

## 1 Introduction

The necessity of computer modeling appears in many areas of computer networks design and use:

- in the initial design phase of the network mechanisms, to allow a realistic assessment of the quality and comparison the proposed mechanism with the existing solutions,
- at the advanced stage of design, when we know the basic features of the new product, to allow:
  - precisely determine the most profitable parameters of the prototype for the planned tasks,
  - modeling the behavior of a specific mechanism in the specific application (e.g. for wide area network, or for the integration the new network elements with the existing solutions without the need to build a full prototype),
- during the use phase, to adapt the network devices configuration and the network protocols parameters to the specific purposes.

Due to the above-mentioned reasons, for the proper implementation of the planned system it is necessary to create the realistic model of the packets (frames, cells ...) traffic. The number of the parameters of the modeled traffic should be determine in the way that modeling results were comparable to those obtained for the traffic observed directly in the network.

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 395–404, 2011.

In the traditional queuing models it is assumed that the input stream of customers (packets, frames, cells ...) is characterized by the interarrival time distribution. The interarrival times are independent and represent the values of the same random variable, hence the generated traffic was characterized by the short-term dependencies. However, the network traffic measurements have shown that these dependencies are long-term. This feature is associated with the self-similarity of the stochastic processes [14]. The problem of self-similarity has been widely described in section 2.

The conventional modeling methods does not take into account the characteristics of self-similarity. Consequently new methods of modeling these sources were founded [5], [24], [2], [4], [7]. Their advantage is undoubtedly a good description of network traffic with low number of parameters. However, they do not offer the possibility of using well-known techniques of the queuing theory to estimate the performance of the computer networks. Therefore we need to develop the methods that allow the use of classical methods of modeling to generate the self-similarity traffic [6], [27], [28].

One of the first attempts to use Markovian modeling, in [27], [28] the authors propose the use of discrete time Markov chains (DTMC) to modulate the packet arrival process. Depending on the value of the model parameters, the traffic generated by the model displays pseudo-LRD characteristics over finite time scales. In [3], the authors use a Batch Markov Arrival Process (B-MAP) generated by a non-ergodic CTMC with an absorbing state and N transient states. The results show a better agreement with the generated traffic compared with the simple Poisson and MMPP generators. The MMPP model proposed in [31] aims at generating traffic with multifractal scaling behavior. The well-known Bellcore traces are fitted with the proposed model, and a number of tests are performed to evaluate the accuracy of the fitting. The MMPP models in [1], [29] are shown to provide good matches of LRD properties under large time scales. The authors of this article used the SSMP self-similar markovian model to generate traffic in a finite time scale [34], [35].

Hidden Markov Models (HMMs) are used in several areas of computer science. Recently the interest in HMM-based models has grown, and HMM models have been proposed as a tool for several network traffic related research problems [17], [18]. In [20], [32] HMM models have been used to model the states of packet channels via corresponding loss probabilities and end-to-end delay distributions. Similar works have been proposed to model wired [21] and wireless [16] packet channels. To the best of our knowledge, only a few modeling works using HMMs to model selfsimilar traffic sources are present in literature.

Section 3 briefly describes the issue of the Hidden Markov Models (HMMs). This section also describes how the Hidden Markov Model is used to create an Internet traffic source. Section 4 presents the analysis of the HMM source in terms of the self-similarity of the generated traffic. Some conclusions are given in section 5.

## 2 Characteristics of the Internet Traffic

Clasically, the traffic intensity, seen as a stochastic process, was represented in queueing models by short term dependencies [12]. However, the analysis of measurements shows that the traffic has also long-terms dependencies and has self-similar character. It is observed on various protocol layers and in different network structures [2,3,5,8,9,11].

The term "*self-similar*" was introduced by Mandelbrot [13] in description of proceses in the field of hydrology and geophysics. It means that a change of time scale does not influence statistical properties of the process. A stochastic process $X_t$ is self-similar with Hurst parameter $H(0.5 \leq H \leq 1)$ if for a positive factor $g$ the process $g^{-H}X_{gt}$ has the same distribution as the original process $X_t$, [14]. Mathematically, the difference between short-range dependent and long-range dependent (self-similar) processes is as follows [15]:

For a short-range dependent process:
- $\sum_{r=0}^{\infty} \mathrm{Cov}(X_t, X_{t+\tau})$ is convergent,
- spectrum at $\omega = 0$ is finite,
- for large $m$, $\mathrm{Var}\,(X_k^{(m)})$ is asymptotically of the form $\mathrm{Var}\,(X)/m$,
- the aggregated process $X_k^{(m)}$ tends to the second order pure noise as $m \to \infty$;

For a long-range dependent process:
- $\sum_{r=0}^{\infty} \mathrm{Cov}(X_t, X_{t+\tau})$ is divergent,
- spectrum at $\omega = 0$ is singular,
- for large $m$, $\mathrm{Var}\,(X_k^{(m)})$ is asymptotically of the form $\mathrm{Var}\,(X)m^{-\beta}$,
- the aggregated process $X_k^{(m)}$ does not tend to the second order pure noise as $m \to \infty$,

where the spectrum of the process is the Fourier transformation of the autocorrelation function and the aggregated process $X_k^{(m)}$ is the average of $X_t$ on the interval $m$:

$$X_k^{(m)} = \frac{1}{m}(X_{km-m+1} + ... + X_{km}) \qquad k \geq 1.$$

There are several methods used to check if a process is self-similar. The easiest one is a visual test: one can observe the behaviour of the process through the scales of time. The other one is the estimation of aggregated index of dispersion $IDC$ or aggregated coefficient of variation $CV$. The aggregated index of dispersion is equal to the variance of the number of arrivals within the interval $m$ divided by the average number of arrivals during the same interval:

$$IDC(m) = \frac{\mathrm{Var}\,(mX_k^{(m)})}{E(mX_k^{(m)})}$$

and $CV$ is

$$CV(m) = \frac{\sqrt{\mathrm{Var}\,(mX_k^{(m)})}}{E(mX_k^{(m)})}$$

For a self-similar processes, *IDC* increases on several time scales and *CV* is much more than 1 for small time intervals. Estimation of Hurst parameter is the most frequently used method to check if a process is self-similar: for non-self-similar processes $H = 0.5$; for $0.5 < H \leq 1$ process is self-similar; the closer $H$ is to 1, the greater the degree of persistance of long-range dependence. The parameter can be estimated by various methods, among others by the analysis of variance-time plot [14]. The variation of aggregated self-similar process is equal to:

$$\mathrm{Var}\,(X_k^{(m)}) \approx \mathrm{Var}\,(X)\, m^{-\beta}, \quad \text{or} \quad \log \mathrm{Var}\,(X_k^{(m)}) \approx \log \mathrm{Var}\,(X) - \beta \log m$$

so the log-log plot of $\mathrm{Var}\,(X_k^{(m)})$ versus $m$ is a straight line with slope $-\beta$, $0 < \beta < 1$, and $H = 1 - \beta/2$.

Self-similarity of a process means that the change of time scale does not influence the process: the original process and the scaled one are statistically the same. It results in long-range dependence and makes possible the occurrence of very long periods of high (or low) traffic intensity. These features have a great impact on a network performance. They enlarge the mean queue lengths at buffers and increase the probability of packet losses, reducing this way the quality of services provided by a network [19]. According to Stallings [19], "Self-similarity is such an important concept that, in a way, it is surprising that only recently has it been applied to data communications traffic analysis". As mentioned above, many empirical and theoretical researches have shown the self similar characteristics of the network traffic. That is why it is necessary to take into account this feature when you have to create a realistic model of traffic sources.

## 3 Hidden Markov Model

Hidden Markov Models (HMMs) [22] is a statistical modelling tool for systems with hidden internal states that can be observed and measured only indirectly. These models have numerous applications in computer science. Recently the interest in Hidden Markov Models (HMMs) has grown and HMM-based models have been proposed in several network traffic related research problems.

*Hidden Markov Model* (HMM) may be viewed as a probabilistic function of a (hidden) Markov chain [22]. This Markov chain is composed of two variables:

- the hidden-state variable, whose temporal evolution follows a Markov-chain behavior ($x_n \in \{s_1, \ldots, s_N\}$ represent the (hidden) state at discrete time $n$ with $N$ being the number of states)
- the observe variable which stochastically depends on the hidden state ($y_n \in \{o_1, \ldots, o_M\}$ and represents the observable at discrete time $n$ with $M$ being the number of observables)

An HMM is characterized by the set of parameters:

$$\lambda = \{\mathbf{u}, \mathbf{A}, \mathbf{B}\}$$

where:

- $\mathbf{u}$ is the initial state distribution, where $u_i = Pr(x_1 = s_i)$
- $\mathbf{A}$ is the $N \times N$ state transition matrix, where $A_{i,j} = Pr(x_n = s_j | x_{n-1} = s_i)$
- $\mathbf{B}$ is the $N \times M$ observable generation matrix, where $B_{i,j} = Pr(y_n = o_j | x_n = s_i)$

Given a sequence of observable variables $y = (y_1, y_2, \ldots, y_L)$ referred to as the *training sequence*, we want to find the set of parameters such that the likelihood of the model $L(\mathbf{y}; \lambda) = Pr(\mathbf{y}|\lambda)$ is maximum. We solved it via the Baum-Welch algorithm, a special case of the Expectation-Maximization algorithm [23], that iteratively updates the parameters in order to find a local maximum point of the parameter set.

We used the well-known Bellcore trace of Internet traffic: OctExt.TL [2]. Each line of this file contains a floating-point time stamp (representing the time in seconds since the start of a trace) and an integer length (representing the Ethernet data length in bytes). We translated the sequence of time stamps into the sequence of inter-arrival times. Then we apply a scheme using *Vector Quantization* (VQ) to translate the obtained sequence of inter-arrival times into a sequence of symbols, and training a HMM for this sequence. The quantization algorithm used is Linde-Buzo-Gray (LBG) algorithm of VQ [25]. Vector Quantization is a clustering technique commonly used in compression, image recognition and stream encoding [26]. It is the general approach to map a space of vector valued data to a finite set of distinct symbols, in a way to minimize distortion associated with this mapping.

We consider an HMM in which the state and the observable variables are discrete. A little portion of the sequences was used as the training sequence to learn model parameters. Performance of trained model are tested on the remaining portions of the sequences.

Then we can use the HMM trained with the Bellcore data as the Internet traffic source model. The Fig. 1 and 2 show respectively the example series of inter-arrival times which are obtained from the Bellcore trace and from our HMM traffic source.

## 4 Analysis of HMM Traffic from the Self-Similarity Point of View

The analysis presented in this section is based on the data generated by the HMM internet traffic source which is described in section 3.

Figures 3 i 4 confirm the presence of long-term dependencies in the generated traffic. The index of dispersion IDC increases with the time scale while the coefficient of variation CV is much greater than 1 for small time scale. For the comparison the figures present also the aggregate index of dispersion and the aggregate coefficient of variation for a Poisson process which represents the process with short-term dependencies.

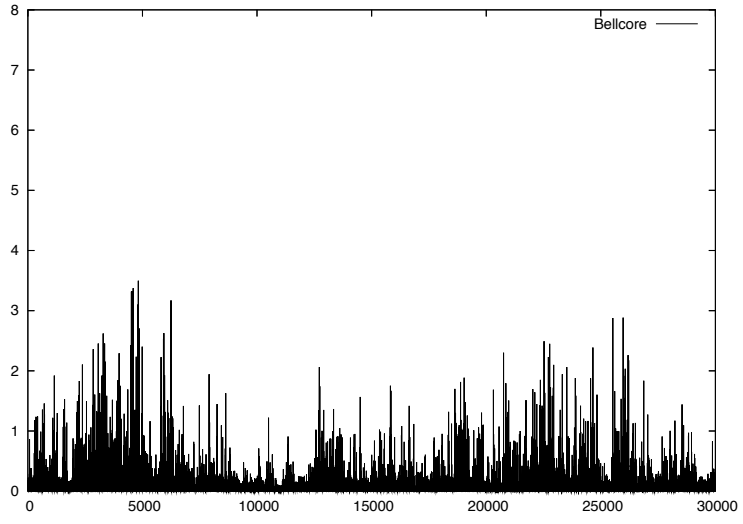



**Fig. 1.** The sequence of inter-arrival times [s] for Bellcore trace

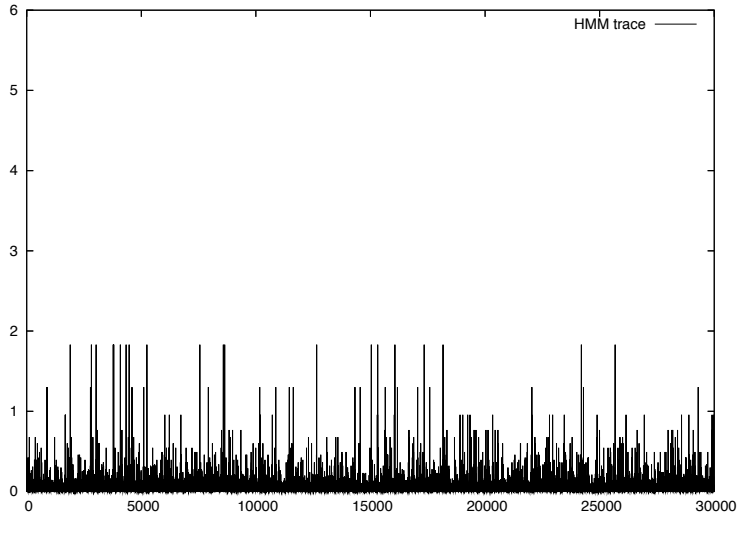



**Fig. 2.** The sequence of inter-arrival times [s] for HMM traffic source trace

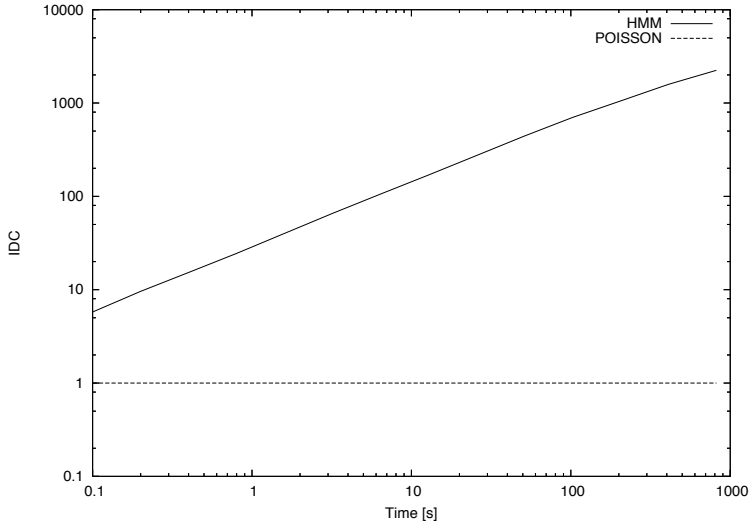



**Fig. 3.** Index of dispersion for HMM traffic source trace

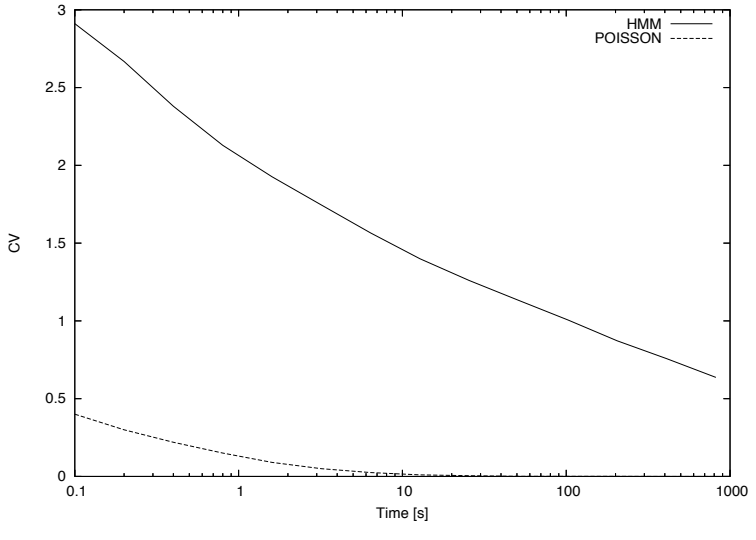



**Fig. 4.** Coefficient of variation for HMM traffic source trace

The degree of self-similarity of the process, expressed as a Hurst coefficient, has been calculated using the variance method (see section 2). Figure 5 shows the dependence of the variance versus time scale (on a logarithmic scale). The slope of the straight line (estimated by the least squares method) is $-0.3$, which gives the Hurst coefficient equal to 0.85. For the omparison, we plotted in the same figure the dependence of the variance versus time scale for a Poisson process. As might be supposed, the slope of this straight line is $-1$, which gives Hurst coefficient equal to 0.5 (that means no self-similarity).

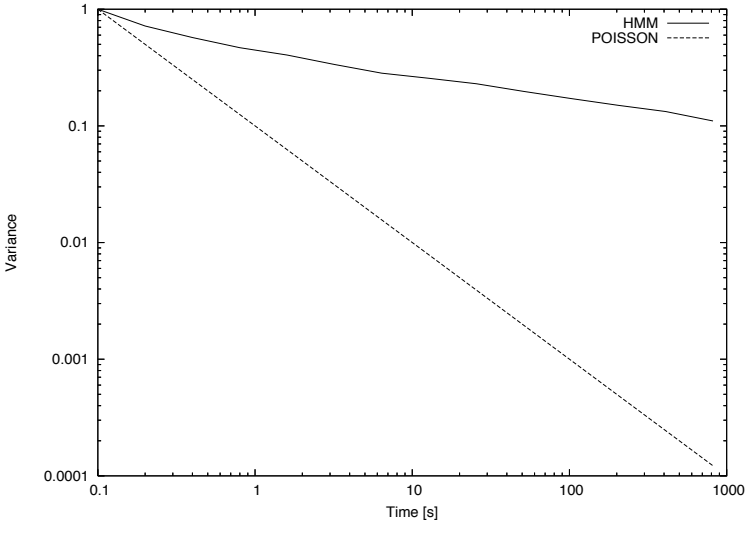



**Fig. 5.** Variance-time plot for HMM traffic source trace

The analysis presented in this section confirm that the developed by the authors HMM traffic sources can generate the traffic which exhibit the self-similarity.

Our HMM traffic generator can be use not only for modeling, but also for generation the real traffic in network connection – e.g. for hardware tests, as devices can be loaded with traffic having characteristics identical with generated with real application.

## 5 Summary

This article has demonstrated that it is possible to generate the self-similar traffic using Hidden Markov Model. Our HMM traffic source was created on the base of the well-known Bellcore Internet traffic trace.

The authors further work will focus on developing HMM sources based on the most recent measurements of the Internet traffic, made available to researchers by the CAIDA organizations [30].

Another important direction for further research is to determine how exactly the generated traffic is fitted to real traffic. In the literature one can find some studies to determine which parameters of the input stream have the greatest impact on the occupancy and the loss of packets in buffers of limited capacity. Although the major output of this study is that higher moments of the process (higher than the second) have a small impact on the processes in queues [10], the question remains still open.

**Acknowledgements.** This research was partially financed by Polish Ministry of Science and Higher Education project no. N N516441438.

## References

1. Andersen, A.T., Nielsen, B.F.: A Markovian approach for modeling packet traffic with long-range dependence. IEEE Journal on Selected Areas in Communications 16(5), 719–732 (1998)
2. Willinger, W., Leland, W.E., Taqqu, M.S.: On the self-similar nature of ethernet traffic. IEEE/ACM Transactions on Networking (1994)
3. Klemm, A., Lindemann, C., Lohmann, M.: Modeling IP traffic using the batch Markovian arrival process. Performance Evaluation 54(2), 149–173 (2003)
4. Veitch, D., Abry, P., Flandrin, P., Chainais, P.: Infinitely Divisible Cascade Analysis of Network Traffic Data. In: Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2000), Istanbul, Turkey, vol. 1 (June 2000)
5. Erramilli, A.: Chaotic maps as models of packet traffic. ITC 14 (June 1994)
6. Baiocchi, A., Melazzi, N.B., Listanti, M., Roveri, A., Winkler, R.: Loss Performance Analysis of an ATM Multiplexer Loaded with High-Speed ON-OFF Sources. IEEE-JSAC 9(3) (April 1991)
7. De Vendictis, A., Baiocchi, A.: Wavelet Based Synthetic Generation of Internet Packet Delays. In: Proceedings of International Teletraffic Conference ITC17, Salvador, Brasil (December 2001)
8. Paxson, V., Floyd, S.: Wide Area Traffic: A Failure of Poisson Modeling. IEEE/ACM Transactions on Networking (June 1995)
9. Crovella, M., Bestavros, A.: Self-similarity in World Wide Web Traffic: Evidence and Possible Causes. IEEE/ACM Transactions on Networking (December 1997)
10. Li, S.Q., Hwang, C.L.: Queue response to input correlation functions: continuous spectral analysis. IEEE//ACM Trans. Networking 1(3), 678–692 (1993)
11. Garret, M., Willinger, W.: Analysis, modeling and generation of self-similar VBR video traffic. In: ACM SIGCOMM, London (September 1994)
12. Kleinrock, L.: Queueing Systems, vol. II. Wiley, New York (1976)
13. Mandelbrot, B., Ness, J.V.: Fractional Brownian Motions, Fractional Noises and Applications. SIAM Review 10 (October 1968)
14. Beran, J.: Statistics for Long-Memory Processes. Chapman and Hall, Boca Raton (1994)
15. Cox, D.R.: Long-range dependance: A review. Statistics: An Appraisal (1984)
16. Iannello, G., Palmieri, F., Pescap, A., Salvo Rossi, P.: End-to-end packet-channel Bayesian model applied to heterogeneous wireless networks. In: IEEE GLOBECOM, pp. 484–489 (November 2005)

17. Dainotti, A., Pescape, A., Salvo Rossi, P., Palmieri, F., Ventre, G.: Internet Traffic Modeling by means of Hidden Markov Models. Computer Networks 52(14), 2645–2662 (2008)
18. Colonnese, S., Rinauro, S., Rossi, L., Scarano, G.: H.264 Video Traffic Modeling via Hidden Markov Process. In: 17th European Signal Processing Conference (EUSIPCO 2009), Glasgow, Scotland, August 24-28 (2009)
19. Stallings, W.: High-Speed Networks: TCP/IP and ATM Design Principles. Prentice-Hall, Englewood Cliffs (1998)
20. Salamatian, K., Vaton, S.: Hidden Markov Modeling for network communication channels. In: ACM SIGMETRICS 2001, vol. 29, pp. 92–101 (2001)
21. Salvo Rossi, P., Romano, G., Palmieri, F., Iannello, G.: Joint end-to-end loss-delay Hidden Markov Model for periodic UDP traffic over the Internet. IEEE Transactions on Signal Processing 54(2), 530–541 (2006)
22. Rabiner, L.R.: A tutorial on Hidden Markov Models and selected applications in speech recognition. Proceedings of the IEEE 77(2) (1989)
23. Bilmes, J.A.: A Gentle Tutorial od the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models. University of Berkeley (1998)
24. Norros, I.: On the use of fractional Brownian motion in the theory of connectionless networks. Technical contribution, TD94-33 (September 1994)
25. Linde, Y., Buzo, A., Gray, R.: An Algorithm for Vector Quantizer Design. IEEE Transactions on Communication Com-28 (1980)
26. Romaszewski, M., Głomb, P.: 3D Mesh Approximation Using Vector Quantization. Advances in Soft Computing 57 (2009)
27. Robert, S.: Modélisation Markovienne du Trafic dans Réseaux de Communication. PhD thesis, Ecole Polytechnique Fédérale de Lausanne, Nr 1479 (1996)
28. Robert, S., Boudec, J.Y.L.: New models for pseudo self-similar traffic. Performance Evaluation 30(1-2) (1997)
29. Salvador, P., Valadas, R., Pacheco, A.: Multiscale fitting procedure using markov modulated poisson processes. Telecommunication Systems Journal 23(1-2), 123–148 (2003)
30. The Cooperative Association for Internet Data Analysis, http://www.caida.org
31. Horvath, A., Telek, M.: A Markovian Point Process Exhibiting Multifractal Behavior and its Application to Traffic Modeling. In: Proceedings of Fourth International Conference on Matrix-analytic Methods in Stochastic Models, Adelaide, Australia (July 2002)
32. Wei, W., Wang, B., Towsley, D.: Continuous-time Hidden Markov Models for network performance evaluation. Performance Evaluation 49(1-4), 129–146 (2002)
33. Domańska, J.: Procesy Markowa w modelowaniu natężenia ruchu w sieciach komputerowych. PhD thesis, IITiS PAN, Gliwice (2005)
34. Domańska, J., Domański, A., Czachórski, T.: The Drop-From-Front Strategy in AQM. In: Koucheryavy, Y., Harju, J., Sayenko, A. (eds.) NEW2AN 2007. LNCS, vol. 4712, pp. 61–72. Springer, Heidelberg (2007)
35. Domański, A., Domańska, J., Czachórski, T.: The impact of self-similarity on traffic shaping in wireless LAN. In: Balandin, S., Moltchanov, D., Koucheryavy, Y. (eds.) NEW2AN 2008. LNCS, vol. 5174, pp. 156–168. Springer, Heidelberg (2008)

# New Synchronization Method for the Parallel Simulations of Wireless Networks

Sławomir Nowak, Mateusz Nowak, and Paweł Foremski

Institute of Theoretical and Applied Informatics,
Polish Accademy of Science,
44-100 Gliwice, Poland
{emanuel,mateusz,pjf}@iitis.pl

**Abstract.** We present a new synchronization method for parallel discrete event simulation for wireless networks. The method merges paradigms of time-stepped and event-driven simulations to achieve reduction of messages exchanged between local processes of simulation. The method is particularly suitable for wireless network simulation, in which objects share the same physical medium and parallel simulation involves a significant overhead for interprocess communication. The article describes the method and compare it to conservative and optimistic synchronization methods, on the basis of results of simulations of selected wireless network model.

**Keywords:** Parallel simulation, wireless network simulation, synchronization.

## 1 Introduction

*Discrete event simulations* (DES) has appeared as the most convenient approach for the performance evaluation of network protocols and architectures. However with the increasing complexity of simulation scenarios the demands on computational resources also significantly increase. This problem is particularly important for the simulation of wireless networks, due to the complexity of the physical layer, shared medium and possible interferences.

Several wireless network simulators have been proposed. Examples are NS2[1], GTNetS [2], and some popular extensions of OMNeT++ [3]: MobilityFramework [4], INET packages [5], MiXiM [6] etc.

The popular approach is to simplify the physical layer model to reduce the number of events and computational complexity. Simplifying the complexity of the model, however, leads to less accurate simulation but the scale of simulated model increases. Different simulation tools are different degrees of detail of physical layer implementation and the complexity of the wireless physical layer enforces the use of simplified models, as the tradeoff between the accuracy and the scalability of simulators. The impact of the physical layer modeling accuracy on both the computational cost and the confidence in simulations was investigated and presented in the literature, among others [7], [8].

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 405–415, 2011.

### 1.1 Parallel Simulations

A single simulation of a network model with thousands of nodes can occupy considerable time to obtain statistically trustworthy results, and many simulation studies require several simulation runs.

Independent replicated simulation runs method is an alternative, but often one simulation run may depend on the results of earlier runs as input.

Using *parallel discrete event simulation* (PDES) over several processors, cores or hosts, it is possible to achieve a considerable speedup. The simulation's scenario is divided in a number of *logical processes* (LPs), each of them has its own clock (LVT, *local virtual time*) and executes a part of the scenario. LPs share demanded resources among several computers or processors [9].

Even though distributing the model over several computers the simulation execution with given synchronization algorithm may result in slower execution than on a single workstation, but at least it is possible to run the model.

The challenging problem with distributed simulation is to synchronize the LPs to ensure each LP processes events in right time stamp order. The case where the incoming event's timestamp is less than LVT of the given LP is called *causality error*.

There are two kinds of algorithms to handle event synchronization: conservative algorithms and optimistic ones. In the *conservative algorithms*, a causality error can never happen. It is obtained by avoiding violating the local causality constraint. In the most popular conservative synchronization (known as Chandy/Misra/Bryant algorithm) "null messages", containing lower timestamp bound of future messages the LP sends, are exchanged between objects for ensuring the given event timestamp is safe to process [10].

In *optimistic algorithms* the causality error may happen, but they are detected and recovered using a withdrawal mechanism (called *rollback*), which moves the state of LP to the moment when processing of delayed event is safe. As rollback withdraws action taken by LP, also events sent in rollbacked period must be cancelled. This leads to rollbacks on another LPs. Consequently, it is necessary to store last states of objects. Rollback operation is time-costly, and necessity of storing past states of LP binds with high memory overhead. The representative optimistic synchronization algorithm is the Time Warp [11].

Parallel simulation of wireless networks, while maintaining accuracy physical layer model, is a particularly difficult issue. It is mostly because of the shared type of the medium and the resulting intensity of communication between objects.

In conventional network simulators of higher layers of network protocols each data unit is represented by a single event, with timestamp set to the delivery time. However, to model the full complexity of the wireless physical layer it is required to represent each data frame by two events: the beginning and end of the transmission, separately for each object using a shared medium. When receiving a frame the interference is evaluated in order to determine which signals are interfering with each other. This set of interfering signals can be very large for large scale simulations and require a very frequent synchronization between LPs and are often useless as frequent communication between LPs severely lowers efficiency of simulation.

Although some of the simulation tools supports for parallel simulation (eg, PDNS – distributed version of NS-2 [12] or OMNeT++ [22]) in the case of complex simulation models (e.g. INET for TCP/IP and wireless network simulations) is not possible or is very difficult. The authors attempted to use a package such as INET simulation in parallel, but the results obtained have proved ineffective [13].

There are also available dedicated simulation tools for simulation of wireless and mobile networks in parallel, e.g. MoVeS [14], GloMoSim [15] and some related work was done [16], [17] etc.

Available tools and methods however, often limited the accuracy of the simulation at the physical layer, compensating for the increased scalability.

In this situation it is reasonable to develop new or modified synchronization methods to improve performance of parallel simulation of wireless communication, while maintaining the accuracy of detailed model of the physical layer. It seems that this is possible by taking into account specific character of wireless communication protocols. In previous work the authors attempted to develop a concept of a new PDES synchronization method of wireless networks with non-zero rate of lost frames, e.g. wireless networks. Thanks to treating straggler messages (all or some of them) as messages informing of damaged frames, better performance of distributed simulation was expected [18], however, failed to achieve the required accuracy of the simulation for sequential simulation.

In the next sections the new method of time stepped synchronization for wireless networks in described. The developed simulation tool and the simple model and scenario is presented. Next, obtained simulation results are presented to proof the accuracy of parallel simulation in relation to sequence version and evaluation of communication overhead of synchronization are described. At the end of the paper we summarize the work and present the related future work.

## 2 New Time Stepped Synchronization Method

In the proposed synchronization method, as in other synchronization algorithms, simulation scenario is partitioned into a number of LP. Each LP handles events, arranged by increasing timestamps (*future events set*, FES). The allocation of simulation objects to each LP is defined in the initialization phase.

The characteristic feature of the proposed method is to introduce synchronization time steps. During the time step each LP operates as an independent, sequential DES, and barrier synchronization occurs only at the end of each time step.

Events created by given LP are directed to internal (local) objects via *local future event set (L-FES)*, and to objects managed by others LP's via *external future event set* (E-FES). Each LP has therefore two separate future events sets: L_FES, which stores the events directed to local objects, and E_FES, for events directed to objects outside the LP.

During the time step LP processes events from the L_FES, with timestamps of the designated boundaries of the current time step. Events addressed to objects in other LP are stored in E_FES. They are exchanged with other LP's during the synchronization phase at the end of time step. The obvious advantage of this solution

is the possibility of fully parallel simulation performance by each LP during a given time step. However it's easy to observe, that delayed events, generated during the time step, but sent after its finish, must cause causality errors. The longer the time step is, the more causality errors occur. Thus the number of causality errors depends on the length of time step. Preliminary studies have shown that for a given scenario, it is possible to calculate time step small enough to obtain no causality errors. This step, however, takes small values in relation to a simulation time limit, which significantly reduces the effectiveness.

To increase the efficiency of time-stepped simulation we take advantage from specific character of network transmission. Events in network simulation are related to transmission of the frame. In wireless network simulation it is important to take into consideration both beginning and end of frame transmission, as this is the only way to detect frame collisions, interferences and other phenomena typical for this kind of communication. Therefore frame transmission is simulated with two events, related to start of transmission (FRAME_START event) and end of transmission (FRAME_END).

Basic consequence of a causality error is start of transmission of the frame, which – in correct simulation – should not be started due to busy channel. As external events are delayed, the LP sending such a erroneous frame discovers the fact the channel is busy at the end of time step, after sending FRAME_START event.

Therefore we introduce another type of event, FRAME_CANCEL. If the information about frame cancelation arrived to the destination before it generates the next event, simple cancellation of frame will restore correctness of simulation.

Possibility of causality error refers to the optimistic synchronization methods. There are still fundamental differences between them and proposed time-stepped simulation. Optimistic simulation assumes rollback, and operation of restoring state of LP and re-simulation of cancelled time-period. Time stepped simulation does not provide any re-simulations. Cancelling of frame has also only local effects, it does not imply subsequent withdrawal of next events, neither locally nor in other LPs.

For the method to work properly it is necessary to impose additional boundaries on time, in which causality errors can occur and implement appropriate causality error handling at the level of simulation objects. It is possible to restore the synchronization with the single FRAME_CANCEL event when:

$$\max T_k < \frac{1}{2}\min(t_{FRAME\_END} - t_{FRAME\_START}), \tag{1}$$

where $T_k$ is the length of time step $k$, and $t_{FRAME\_START}$ and $t_{FRAME\_END}$ means the timestamps of events related to respectively beginning and end of a frame transmission. In other words, the length of a single time step must be less than half the time of minimum frame transmission time in a given scenario.

Adoption of the length of time step consistent with (1) ensures that the transmission of the frame, started in time step $k$ will be delivered to the destination objects in step $k+2$ (or later).

Therefore, the causality error may occur in one, specific situation: The object $O_2$ located in $LP_2$ in time step $k$ at time $t_2$ checks the status of the wireless medium, and states that it can start sending a frame, so it sends a FRAME_START event. On the next time step $k+1$, the $O_2$ object is receiving a delayed FRAME_START event from simulation object $O_1$, located in the $LP_1$. The timestamp of the event $t_1 < t_2$. This means a causality error situation (see Fig.1).

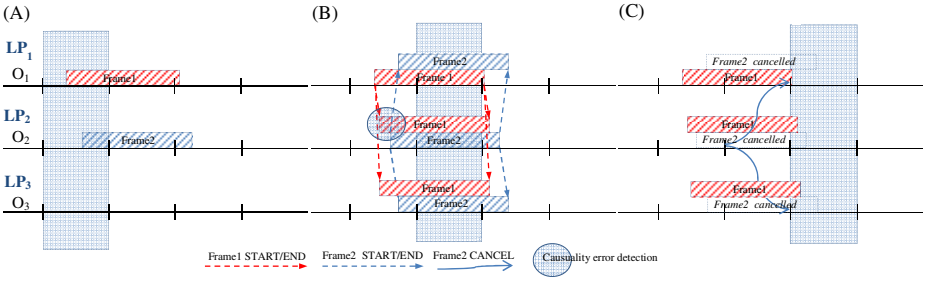



**Fig. 1.** The causality error in the synchronization method with the time step (A, B, C: subsequent time steps). (A) Object 1 ($O_1$) detects free channel and send Frame1. $O_2$ detects free channel, and sent Frame2, (B) Obj2 has detected the causality error, cause channel was not free for Frame2 transmission FRAME_CANCEL of Frame2 is sent. $O_3$ has detected both (1 and 2) frames, (C) $O_1$ and $O_3$ have received event FRAME_CANCEL before the FRAME_END of Frame2 will occur and restore the correct state of objects.

To restore correctness of simulation, object $O_1$ sends FRAME_CANCEL of previously sent event. This event is delivered to all objects which previously received erroneous FRAME_START.

If the condition (1) is fulfilled, the FRAME_CANCEL reaches the destination objects: (a) before the FRAME_START event is delivered at time step $k+1$ or (b) before the FRAME_END is delivered at the time step $k+2$.

In the general case of wireless networks simulations (in which objects receive FRAME_START and then record changes in the wireless medium parameters), this would allow the object to withdraw the state associated with the time errorneus frame without generating further events.

This solution involves the need to remember (with an accuracy of one time step) of the state of the object. The objects must be able to withdraw its state before receiving a FRAME_START event (with restoration of counters, timers, collisions indicators etc.). This is clearly an implementation overhead, and is associated with the implementation of object-level simulation and may be supported only slightly by a kernel of simulator. But unlike the optimistic synchronization methods synchronization may be restored by a single FRAME_CANCEL event instead of handling the possible subsequent chain of rollbacks procedures.

In this way we achieve a tradeoff between the low efficiency of conservative synchronization and high complexity of implementing the optimistic synchronization.

## 3 Simulation Evaluation

For verification of synchronization method presented above, simple pseudo-parallel simulator was implemented. Simple, however realistic model of wireless communication was taken for proofing correctness of simulation.

### 3.1 Simulation Tool

The simulation tool was developed to perform the time-stepped DES in time steps using described synchronization. Simulations was done both in sequential and the pseudo-parallel mode. In the pseudo-parallel mode LPs run in a fully independent way, but are called in sequence in the main simulation loop. A simple concept is as follows:

```
do
{
  foreach (LP lp in LPset)
  {
    lp.step(stepSize);
    simTime = simTime + stepSize;
  }
}
while (simTime < simulationTimeLimit);
```

This approach allows us for convenient analysis and debugging in both sequential and parallel way and lets verify the results step by step. The next step was to implement the selected model of the simulation model and determine its parameters to obtain the reference simulation scenario.

### 3.2 Model of Simple Wireless Network

The simplified model of communication in a wireless network was adopted, since the primary objective of the study was to evaluate the methods for synchronization. This model, however, remains consistent with the *Distributed Coordination Function* (DCF) of the IEEE 802.11-2007 standard [19], i.e. the most popular setup used in existing wireless networks.

Model simulates the DCF collision avoidance algorithm whereas each network node must ensure it can access the medium before transmission starts. Once started, transmission cannot be interrupted (eg. by interfering signal) until frame transmission finishes. It is even not possible to tell if any such attempts was made. This is consistent with radio devices found on the market (eg. Atheros chipset).

A few mechanisms modeled according the standard are used: *carrier sensing* (CS), *NAV timers* (NAV), *inter-frame spacing* (IFS) and *exponential back-off* (EBO). The IFS facility takes into account if the most recently received frame was successful (see 9.2.3 in [19]). For EBO, a simplified formula (2) for the contention window (CW) is used:

$$CW(r) = \min(2^{3+r} - 1, 1023). \quad (2)$$

In the above formula, $r$ is the number of retries for a given frame.

Nodes only exchange frames without using acknowledgements (ACK), what reflects a multicast transmission. In addition, the new IEEE 802.11n standard [20] introduces programmable support for disabling this facility. The RTS/CTS mechanism is not used – it has been poorly adopted by the ISP industry due to its performance and is rarely used nowadays.

Topology model includes $n$ hosts, acting as client-server pairs ($n/2$ pairs). The client requests a file transfer from a dedicated server and the server responds by sending data. At the application layer a protocol similar to the TFTP protocol [21] is used. UDP datagrams are used at the transport layer and IP packets at the network layer. Each TFTP data unit (TFTP-DU) is acknowledged by the customer using a TFTP acknowledged datagram (TFTP-ACK).

At the level of network layer we have two types of frames: TFTP-DU frames (sized 1518 bytes = 1452 bytes of useful data + related protocols headers) and TFTP-ACK frames (sized 68 bytes). Client objects send requests at random time intervals with uniform distribution (0, 1) s. The transmission rate is set to 1Mbps. The distance between any two hosts on the network is fixed (constant delay between objects is assumed).

### 3.3 Synchronization and Causality Errors Handling

The experiments adopt a model in which there are two lengths of frame. The duration of the shortest frame (TFTP acknowledged datagram) in accordance with the assumed rate of transmission is 544μs. The maximum length of time step, respecting (1) was set to 200us.

If the simulation object detects a causality error, sends a FRAME_CANCEL event and restores its state (counter of sending attempts, backoff timer, the status of last transmission, and relevant statistics). The object receiving FRAME_CANCEL also restores its state (statistics and the status of last transmission).

### 3.4 Parallel Simulation Accuracy

The results correctness was a necessary condition for the start of testing for further evaluation of the synchronization method.

As an example of a graph of average servers rate as a function of number of client-server pairs is presented (see Fig.2). The parallel version used 10 LPs. Simulation objects are partitioned equally between the LPs, with the principle that a client to work on another LP than dedicated server.

The evaluation of obtained results confirmed consistency of results with those obtained in a sequential manner. Minor deviations results from difficulty in withdrawing multiple randomly generated values. In terms of statistical correctness, however, the results should be regarded as compatible.

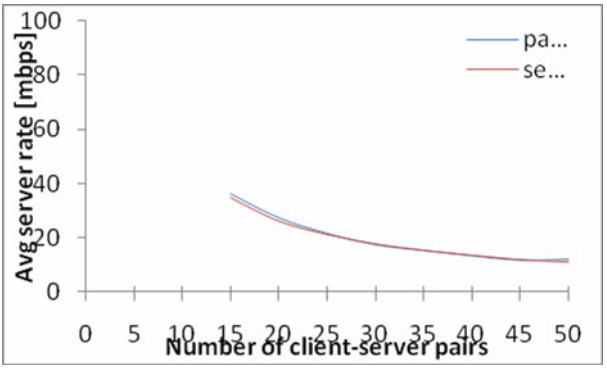



**Fig. 2.** Average servers rate as a function of the number of client-server pairs

### 3.5 Synchronization's Overhead Analysis

Quantitative comparison between novel time-stepped algorithm and classical synchronization methods is hard, as the synchronization model in time stepped, Null-Messages and Time Warp synchronization algorithms is completely different.

Overhead of Jefferson's Time warp [11] is tightly connected with rollback procedures. While normal operation, with no rollbacks, no administrative messages are sent over the network. Comparing to sequential simulation, Time Warp controlled LP must do network I/O, state savings and Global Virtual Time computations. Overhead of these operations is noticeable, but not high. However, when causality error occurs in given LP, simulation on the LP must be stopped. The rollback operation, based on identifying and copying of proper LP state, is performed. This is memory operation, so its overhead is impossible to compare with network overhead coming from sending of network messages. After doing local state restoration some messages, already sent to other LPs must be invalidated. This is done with so called anti-messages, which can cause further rollbacks on LPs receiving them. The bigger the difference between the simulated time value before and after rollback, the more anti-messages must be sent. Thus the efficiency of Time Warp is very strongly dependent on frequency and depth of rollbacks.

Parallel simulation with use of Null Messages is considered less complicated for implementation, but introducing bigger overheads. They come from network communication delays as well as from the algorithm itself (synchronization overhead). Number of messages sent over the network is much bigger than in case of Time Warp. After processing of each message the node computes its Earliest Output Time (EOT), which is lower bound on the timestamp of any message that LP may send. Value of EOT is then sent to every LP the sender has the link with. Thus a lot of messages is exchanged over the network, increasing communication overhead and their exact number depends strongly on number of links between LPs in the particular simulation scenario. Synchronization overhead is also hard to evaluate, as every LP waits for receiving of all messages needed for computation of time bounds allowing safe processing of events in FES. Fig. 3 presents the number of events exchanged in the model, number of events exchanged between LPs (not all event messages leave the LP) and number of FRAME_CANCEL messages in exemplary scenario, engaging 10 LPs, and with number of simulated network nodes varying from 15 to 50.

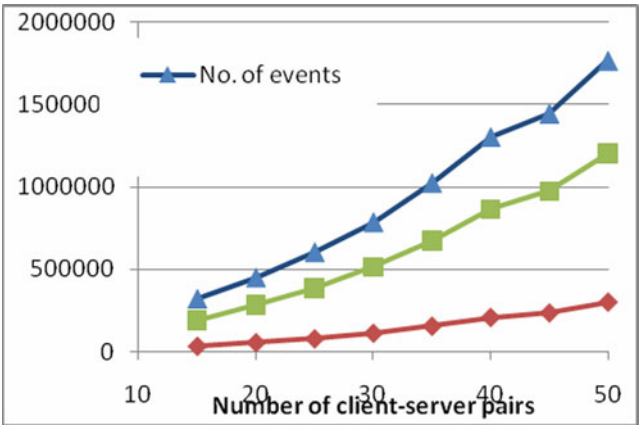



**Fig. 3.** Number of messages exchanged in the parallel simulator. Number of LPs: 10, number of object pairs varying from 15 to 50

Communication and simulation model used in time stepped simulation described in the paper is even more different. The number of network messages exchanged between LPs does not depend on the number of events. Contrary, network message is broadcasted to every other LP at the end of time step. As the event messages are combined into one, the more events are exchanged, the longer the messages are, but their number remain constant (at the application level – we do not consider packet splitting due to preserving MTU). Performance problems arise, as all LP work synchronously and communicate at the same time.

As we show above, efficiency comparison of different synchronization algorithms is not possible, as the number of factors influencing performance of parallel simulation controlled the algorithms is too large. One could however expect, that constant number of messages exchanged while using time-stepped method will cause that the method will be particularly effective for big networks simulations. If LP handle a few hundred or a few thousand network nodes, each communicating with other nodes, the number of single events exchanged in every time step becomes huge. Aggregation of communication may significantly increase performance of parallel simulation.

## 4 Future Works

As the next steps the implementation in a fully parallel architecture will be provided. It will allow the practical evaluation of the efficiency, as well as evaluating scenarios with a greater number of nodes.

The further developing of a method, implementing adoptable time step length, instead setting it arbitrarily at the beginning of the simulation is planned. Time step will be selected dynamically, based on the length of the currently transmitted frames.

An important element of the future work will be to develop and simulate more complex models of the physical layer, taking into account the modulation (the transmission of symbols instead of bits), radio propagation models, full interference model and MIMO.

## 5 Summary

The paper proposes a new method of parallel discreet event simulation synchronization. In relatively big simulation scenarios it reduces the communication overhead required for synchronization of parallel processes, leading to a potential increase in efficiency of parallel simulation.

The solution has been evaluated on the base of a simple model of wireless network. The correctness of the method in comparison to the sequential simulation has been proved. The disadvantage of the solution is an additional effort of the implementation and the need for resolving the causality errors at the level of the model. This implementation overhead can be significant depending on the complexity of the model.

With the assumptions adopted in the synchronization method the casuals errors related to frames transmission can be resolved in the next synchronization step (one frame "cancelling" step).

**Acknowledgments.** The work presented in the paper was supported by Polish Ministry of Science and Higher Education grant no. N N516 407138.

## References

1. The network simulator, ns-2, http://www.isi.edu/nsnam/ns/
2. GTNstS homepage, http://www.ece.gatech.edu/research/labs/MANIACS/GTNetS/
3. OMNeT++ homepage, http://inet.omnetpp.org/
4. Mobility Framework homepage, http://mobility-fw.sourceforge.net/
5. OMNeT++/INET homepage, http://inet.omnetpp.org/
6. MiXiM homepage, http://mixim.sourceforge.net
7. Ben Hamida, E., Chelius, G., Gorce, J.M.: Impact of the Physical Layer Modeling on the Accuracy and Scalability of Wireless Network. Simulation 85(9), 574–588 (2009)
8. Kotz, D., Newport, C., Gray, R.S., Liu, J., Yuan, Y., Elliott, C.: Experimental evaluation of wireless simulation assumption. In: Proceedings of the 7th ACM Internetional Symposium on Modeling Analysis and Simulation of Wireless and Mobile Systems (2004)
9. Fujimoto, R.: Parallel and Distributed Simulation Systems. John Wiley and Sons, Inc., Chichester (2000)
10. Chandy, K., Misra, J.: Distributed Simulation: A Case Study in Design and Verification of Distributed Programs. IEEE Transactions on Software Engineering 5(5), 440–452 (1979)
11. Jefferson, D., Sowizral, H.: Fast concurrent simulation using the Time Warp mechanism. In: SCS Conf. Distributed Simulation, San Diego, CA (USA), January 24-26, pp. 63–69 (1985)

12. Parallel/ Distributed ns homepage, http://www.cc.gatech.edu/computing/compass/pdns
13. Nowak, M., Nowak, S.: Parallel simulations with modified INET simulation package. Theoretical and Applied Informatics 19(2), 147–156 (2007)
14. Bononi, L., Felice, M., D'Angelo, G., Bracuto, M., Donatiello, L.: MoVES: A framework for parallel and distributed simulation of wireless vehicular ad hoc networks. Computer Networks 52, 155–179 (2008)
15. Zeng, X., Bagrodia, R., Gerla, M.: 15: A library for Parallel Simulation of Large-scale Wireless Networks. In: Proceedings of the 12th Workshop on Parallel and Distributed Simulation (PADS 1998), pp. 154–161 (1998)
16. Peschlow, P., Voss, A., Martini, P.: Good news for parallel wireless network simulations. In: Proceedings of the 12th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (2009), doi:10.1145/1641804.1641828
17. Liu, W., Chiang, C., Jha, V., Grela, M., Bagrodia, R.: Parallel Simulation Environment For Mobile Wireless Networks (1996), doi:10.1.1.47.7806
18. Nowak, M., Nowak, S.: Synchronisation concept for distributed simulation of networks with packet loss. Studia Informatice 30(1), 81–89 (2009)
19. Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE Standard 802, Part 11 (2007)
20. Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE Standard 802, Part 11, Amendment 5 (2009)
21. Sollins, K.: The TFTP protocol (Revision 2). IETF RFC 1350 (1992)
22. Varga, A., Sekercioglu, A.Y.: Parallel Simulation Made Easy With OMNeT++. In: Proc. 15th European Simulation Symposium and Exhibition (2003)

# Steady State Analysis of Three Node Client Relay System with Limited-Size Queues

Olga Galinina[1], Sergey Andreev[1],
Alexey Anisimov[2], and Alexandra Lokhanova[3]

[1] Tampere University of Technology (TUT), Finland
olga.galinina@tut.fi, sergey.andreev@tut.fi
[2] State University of Aerospace Instrumentation (SUAI), Russia
alexey.anisimov86@googlemail.com
[3] St. Petersburg Speech Technology Center (STC), Russia
alokhanova@yandex.ru

**Abstract.** In this paper, we present an accurate analytical model for the performance evaluation of the client relay wireless system, where users cooperate sending uplink data packets. We consider the simplified three-node system topology with limited-size queues. Using the embedded Markov chain technique based on the length of each packet queue and the contents of the relay queue, we obtain the equilibrium distribution and, therefore, the mean overall packet delay and the packet loss probability for all the nodes. The analytical model is compared with simulation to conclude on its accuracy.

**Keywords:** queueing system, embedded Markov chain, client relay, packet delay, packet loss.

## 1 Introduction

Recently, due to the standardization process of next-generation mobile broadband communication systems such as 3GPP LTE-Advanced [1], IEEE 802.16j [2] and IEEE 802.16m [3] cooperative networks receive increasing attention from the research community [4], [5], [6].

Different aspects of cooperative transmission have been discussed in the literature, as well as plenty of cooperative schemes with different assumptions have been proposed. However, limited attention has been paid to the cooperation between wireless clients themselves. Some recent works [7], [8] have demonstrated that spectral and energy efficiency may be improved by uplink client relay schemes. Therefore, cooperative transmission could become an effective solution to improve the performance of the future wireless networks.

In our previous paper [9], we have considered a prominent client relay technique and studied its delay performance by introducing a simple approximation. In this paper, we continue our work by analyzing a more practical cooperative system with limited-size queues. Here we calculate numerically the exact value of the mean overall delay using the steady state distribution.

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 416–427, 2011.

The rest of the paper is organized as follows. Section 2 includes specific details, parameters and assumptions comprising our analytical model. The queueing analysis is given in Section 3 including expressions for the queue length and overall delay. The validation of the model and the numerical results are discussed in Section 4, as well as the conclusion of this work. In the Appendix, we detail the important transition probabilities.

## 2 System Model

In this section, we briefly outline the assumptions of our client relay model. For more details, see our previous paper [9]. In order to make the analytical model mathematically tractable, we study the simplified network topology (see Figure 1) under the following conditions.

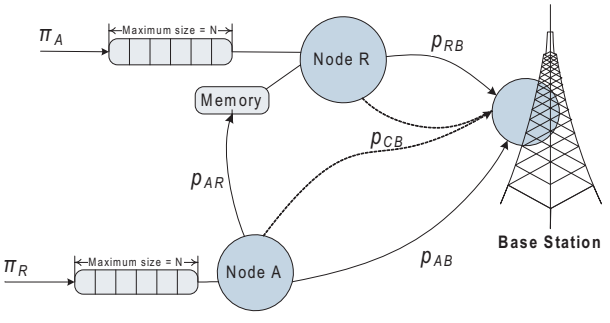



**Fig. 1.** Simplified client relay system operation

### 2.1 Modeling Assumptions

*System assumptions.*

1. The system time is divided into slots. We assume that the transmission of each data packet takes exactly one slot.

2. Source node $A$ generating new data packets with the mean arrival rate of $\lambda_A$ packets per slot is termed the *originator*. Similarly, source node $R$ generating new data packets with the mean arrival rate of $\lambda_R$ packets per slot is termed the *relay*. The sink node $B$ controlling the system and receiving data packets from both node $A$ and node $R$ is termed the *base station*.

3. We assume fair stochastic round-robin scheduler at the base station $B$ and immediate transmission of scheduling information to both source nodes over a separate channel.

4. Node $R$ is able to intercept transmissions from node $A$ and keep the packets from node $A$ for the subsequent retransmission. We also assume that simultaneous reception and transmission is not possible.

5. Both queues at the nodes $A$ and $R$ have constant limitation on their size. Without loss of generality, we assume that the queue size equals $N$ for both nodes. The node $R$ has an additional memory location to store a single intercepted packet from the node $A$.

*Traffic assumptions.*

6. Numbers of new data packets arriving to the node $A$ or $R$ during sequential slots are i.i.d. random variables. In order to obtain the discrete model and attain analytical tractability, we keep the arrival process and the service discipline discrete. Hence we assume Bernoulli arrival process.

*Channel assumptions.*

7. Communication channel is error-prone. It is based on the multi-packet reception channel model [10].

8. A data packet is received by the destination node successfully with the constant probability $p_{AB}$, $p_{RB}$, $p_{AR}$ and $p_{CB}$ ($p_{AR} > p_{AB}$, $p_{CB} > p_{AB}$) according to the link and node type. If the packet transmission is unsuccessful, the source retransmits the failed packet. The maximum number of allowable retransmission attempts is unlimited.

9. Feedback information is immediately available to both source nodes over a separate channel. In practice, this information is typically available in the downlink channel and this assumption only simplifies the understanding of the model.

### 2.2 Modeling Parameters

We consider the following system parameters.

- $p_{AB}$ is the probability of successful reception from $A$ at $B$ when $A$ transmits;
- $p_{RB}$ is the probability of successful reception from $R$ at $B$ when $R$ transmits;
- $p_{AR}$ is the probability of successful reception from $A$ at $R$ when $A$ transmits;
- $p_{CB}$ is the probability of successful reception from $A$ at $B$ when $A$ and $R$ cooperate;
- $\pi_A$ is the parameter of Bernoulli distribution describing the arrival process at the node $A$;
- $\pi_R$ is the corresponding parameter of Bernoulli distribution for the node $R$;
- $Q_A^{(t)}$ is the number of packets in the queue of the node $A$ at the moment $t$;
- $Q_R^{(t)}$ is the number of packets in the queue of the node $R$ at the moment $t$;
- $Q_M^{(t)}$ is the number of packets from the node $A$ in the additional memory location of the node $R$ at the moment $t$;
- $\delta_A$ is the mean packet delay of the node $A$;
- $\delta_R$ is the mean packet delay of the node $R$;
- $N$ is the maximum size of every queue.

## 3 Queueing Analysis

In this section, we establish a queueing model of the system and detail all possible transitions along with the corresponding transition probabilities.

The first step in the analysis is the choice of a convenient set of embedded points and the corresponding Markovian state description. Eventually, we chose the following embedded points (see Figure 2), which are located between the end of previous slot (when system estimates the number of packets in every queue) and the scheduler decision. Therefore, we study the behavior of the queues $A$, $R$ and the additional memory location $M$ at the embedded epochs.

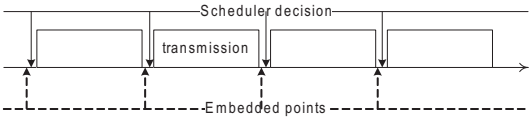



**Fig. 2.** Embedded points of the Markov chain for the considered system

### 3.1 The Transition Probabilities for the Cooperative System

In this section, we introduce the probability of transition, consider the transitions from state to state and also define the matrix representing transition probabilities of the embedded Markov chain.

Let $Q_A^{(t)}$ be a random variable representing the number of packets in the queue of node $A$ at the beginning of the frame $t$, $t > 0$. Similarly, let $Q_R^{(t)}$ and let $Q_M^{(t)}$ represent the number of packets in the queue of node $R$ and in the queue $M$ at the beginning of the frame $t$, respectively. Therefore, we may denote the system state as $S^{(t)} = (Q_A^{(t)}, Q_R^{(t)}, Q_M^{(t)})$. We also define the transition probability from the state $S^{(t)} = (Q_A^{(t)}, Q_R^{(t)}, Q_M^{(t)})$ to the state $S^{(t+1)} = (Q_A^{(t+1)}, Q_R^{(t+1)}, Q_M^{(t+1)})$ as follows

$$p^{(t)}_{(i',j',k'),(i,j,k)} \doteq Pr\{Q_A^{(t)}=i, Q_R^{(t)}=j, Q_M^{(t)}=k | Q_A^{(t-1)}=i', Q_R^{(t-1)}=j', Q_M^{(t-1)}=k'\},$$
$$t > 0, i, j, i', j' \geq 0, k, k' = 0, 1. \quad (1)$$

In this case, $S^{(t)}$ is a Markov process (chain) with stationary transition probabilities. Consequently, we may define the corresponding state transition graph as shown in Figure 3, where every vertex represents a state and edges represent transitions between the states. For convenience, we simplify the complete graph by merging the two states for $Q_M = 0$ and 1 into one state.

Notice that since not more than one packet (from node $A$ or $R$) is served per frame and a packet of either type can be received in general case, the system allows only the following transitions to the state $(i, j, k)$ (as shown by the graph): $(i-1, j-1, \cdot) \to (i, j, \cdot)$, $(i-1, j, \cdot) \to (i, j, \cdot)$, $(i, j-1, \cdot) \to (i, j, \cdot)$, $(i, j, \cdot) \to (i, j, \cdot)$, $(i+1, j, \cdot) \to (i, j, \cdot)$, $(i, j+1, \cdot) \to (i, j, \cdot)$, $(i-1, j+1, \cdot) \to (i, j, \cdot)$, $(i+1, j-1, \cdot) \to (i, j, \cdot)$. The sign '$\cdot$' details whether relay has a packet from $A$ to transmit or not.

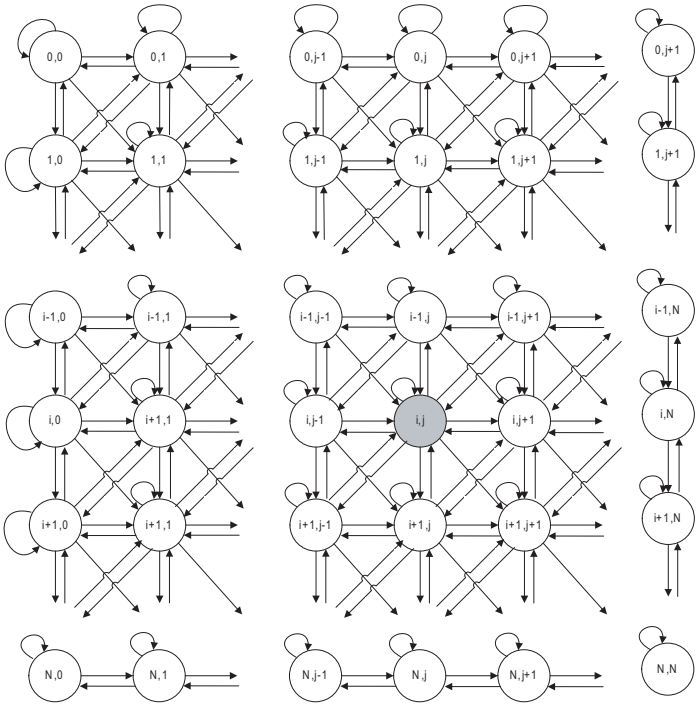



**Fig. 3.** State transition graph for the considered Markov chain

Under the assumption that the considered system reaches the steady state, let $p_{(i,j,k)}$ denote the equilibrium stationary probability of the event, when $Q_A^{(t)} = i$, $Q_R^{(t)} = j$, $Q_M^{(t)} = k$

$$p_{(i,j,k)} \doteq \lim_{t\to\infty}(Pr\{Q_A^{(t)} = i,\ Q_R^{(t)} = j,\ Q_M^{(t)} = k\}), i, j \geq 0, k = 0, 1. \qquad (2)$$

Since the considered Markov chain is time homogeneous, we omit the index $(t)$. See Figure 4 for the proposed grouping of transition probabilities and the corresponding links to the tables in the Appendix.

1. The first case describes the simplest state of the system. We consider the state $S = (0, 0, 0)$ and easily establish expressions for four corresponding transition probabilities $p_{(0,0,0),(0,0,0)}$, $p_{(1,0,0),(0,0,0)}$, $p_{(1,0,1),(0,0,0)}$ and $p_{(0,1,0),(0,0,0)}$. For details see Table 1 in the Appendix.

2. Then consider the boundary states, for instance, states $S = (i, 0, 0)$ and $S = (i, 0, 1)$, $0 < i < N$, and define the transition probabilities for them. The system can reach the state $(i, 0, \cdot)$ only from the states $(i-1, 0, \cdot)$, $(i, 0, \cdot)$, $(i+1, 0, \cdot)$, $(i, 1, \cdot)$, $(i-1, 1, \cdot)$ according to the state transition graph (Figure 3). The corresponding transmission probabilities are shown in Table 2 in the Appendix.

3. Now consider another type of the boundary states $S = (0, j, 0)$, $S = (0, j, 1)$, $0 < j < N$ and obtain the transition probabilities for them. We obviously do

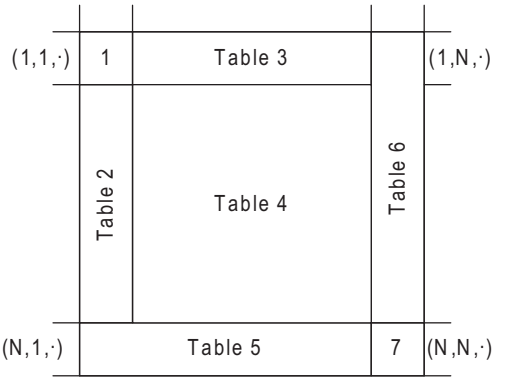



**Fig. 4.** State transition graph for the considered Markov chain

not consider the unreachable states $S = (0, j, 0)$, $j > 0$ since the queue $M$ is always empty unless the originator has a packet to transmit. The corresponding probabilities are listed in Table 3 in the Appendix.

4. The transition probabilities for the general state $S = (i, j, 0)$, $S = (i, j, 1)$, $0 < i, j < N$ can be described as shown in Table 4 in the Appendix.

5. In order to derive other boundary probabilities related to the states of the maximum queue size we have to make respective modifications as listed in Tables 5, 6 and 7.

We remind that the queue sizes are finite. As such, consider the elements of orthonormal basis of space $\mathbf{R}^{2(N+1)}$: $\left\{\mathbf{e_A}^{(i)}\right\}_{i=0}^{2N+1}$. Here $\mathbf{e_A}^{(i)} = (0, \ldots, 0, 1, 0, \ldots, 0)$, where 1 stands at the $i$-th position. Similarly, consider $\left\{\mathbf{e_R}^{(j)}\right\}_{j=0}^{N} \subset \mathbf{R}^{(N+1)}$. Further, let $\otimes$ stand for tensor product. We introduce vector $\boldsymbol{\theta} \in R^{2(N+1)^2}$ representing the stationary probabilities of the Markov chain given above as

$$\boldsymbol{\theta} = \sum_{i=0}^{N}\sum_{j=0}^{N} p_{(i,j,0)} \mathbf{e_A}^{(2i)} \otimes \mathbf{e_R}^{(j)} + \sum_{i=0}^{N}\sum_{j=0}^{N} p_{(i,j,1)} \mathbf{e_A}^{(2i+1)} \otimes \mathbf{e_R}^{(j)}. \tag{3}$$

The transition probability matrix $\boldsymbol{\Pi} \in R^{2N^2} \times R^{2N^2}$ is given as

$$\boldsymbol{\Pi} = \sum_{i,j=0}^{N} \sum_{i',j'=0}^{N} \sum_{k,k'=0}^{1} p_{(i,j,k),(i',j',k')} \left(\mathbf{e_A}^{(2i+k)} \otimes \mathbf{e_R}^{(j)}\right) \otimes \left(\mathbf{e_A}^{(2i'+k')} \otimes \mathbf{e_R}^{(j')}\right)^T \tag{4}$$

The stationary probabilities of the Markov chain can be determined from the following system of linear equations

$$\boldsymbol{\theta}\boldsymbol{\Pi} = \boldsymbol{\theta}, \quad \sum_{i=0}^{N}\sum_{j=0}^{N} p_{(i,j,0)} + \sum_{i=0}^{N}\sum_{j=0}^{N} p_{(i,j,1)} = 1. \tag{5}$$

### 3.2 The Mean Number of the Packets in Relay and Originator Outgoing Queues

In this subsection, we give expressions for the mean number of packets in the relay and originator outgoing queues. All the performance parameters can be derived from the steady-state probabilities. Obviously, the mean number of packets in the queue of the node $A$ (and similarly of $R$) at the beginning of every frame, $E[Q_A]$, can be computed from the stationary distribution as

$$\boldsymbol{E}[Q_A] = \sum_{i=0}^{N}\sum_{j=0}^{N} ip_{(i,j,0)} + \sum_{i=0}^{N}\sum_{j=0}^{N} ip_{(i,j,1)}, \tag{6}$$

$$\boldsymbol{E}[Q_R] = \sum_{i=0}^{N}\sum_{j=0}^{N} jp_{(i,j,0)} + \sum_{i=0}^{N}\sum_{j=0}^{N} jp_{(i,j,1)}. \tag{7}$$

Let us denote the loss probability at the node $A$ for a particular state as

$$p^{(A)}_{loss(N,j,k)} \doteq Pr\{\text{a new packet arrives at } A \text{ but becomes discarded} \mid Q_A = N, Q_R = j, Q_M = k\}, 0 \le j \le N, k = 0, 1. \tag{8}$$

Obviously, $p^{(A)}_{loss(i,j,k)} = 0$ if $i < N$. According to the formula of total probability, the loss probability (the proportion of the discarded packets) for the node $A$ follows from the above expression as

$$\boldsymbol{P}^{(A)}_{loss} = \frac{1}{\pi_A}\sum_{j=0}^{N}\sum_{k=0}^{1}\left(p^{(A)}_{loss(N,j,k)} \cdot p_{(N,j,k)}\right). \tag{9}$$

Similarly, we may establish the loss probability for the node $R$. We omit it here for the sake of brevity. The expressions for $p^{(A)}_{loss(N,j,k)}$ and $p^{(R)}_{loss(i,N,k)}$ could be obtained using the corresponding transition probabilities described above.

### 3.3 The Mean Delay of Relay and Originator Packets

We define the overall delay $\delta_A$ of the node $A$ (and similarly $\delta_R$ of the node $R$) as the time interval spent from packet arrival into the queue of the node up to the end of its successful transmission to the base station $B$. According to Little's law in the form $E[Q_A] = \pi_A\delta_A$, where $\pi_A$ denotes the intensity of the arrival process, we calculate the system delay as follows

$$\boldsymbol{\delta}_A = \frac{1}{\pi_A}\left(\sum_{i=0}^{N}\sum_{j=0}^{N} ip_{(i,j,0)} + \sum_{i=0}^{N}\sum_{j=0}^{N} ip_{(i,j,1)}\right), \tag{10}$$

$$\boldsymbol{\delta}_R = \frac{1}{\pi_R}\left(\sum_{i=0}^{N}\sum_{j=0}^{N} jp_{(i,j,0)} + \sum_{i=0}^{N}\sum_{j=0}^{N} jp_{(i,j,1)}\right). \tag{11}$$

## 4 Numerical Results and Conclusion

In this section, we compare the results of our analytical model with the simulation. An event-driven simulator that accounts for the discussed system features was developed and described in [11]. In Figure 5, we compare analytical and simulation results for the mean overall packet delay and for the different values of the parameter $N$. Additionally, we present the loss probability plot in Figure 6.

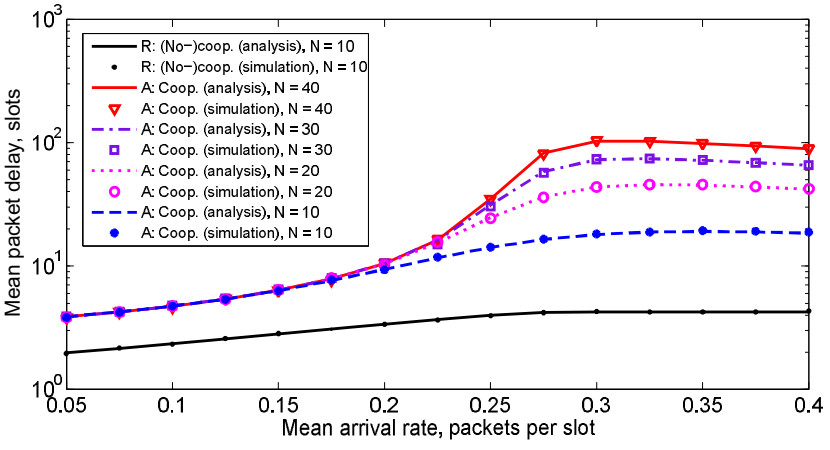



**Fig. 5.** Numerical results: overall delay for the nodes $A$ and $R$ vs. arrival rate

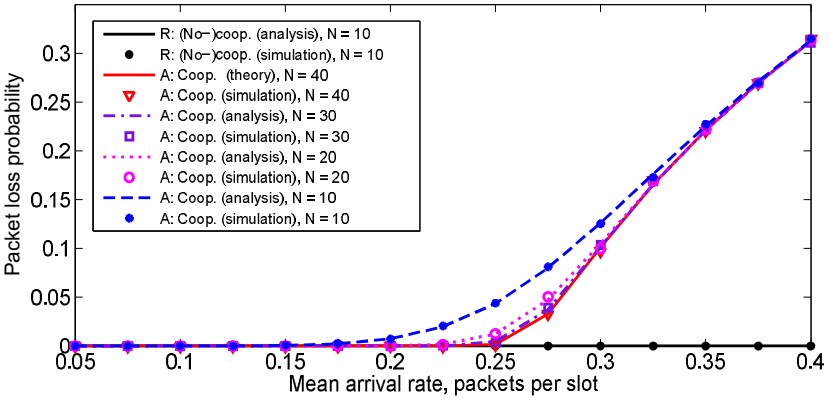



**Fig. 6.** Numerical results: loss probability for the nodes $A$ and $R$ vs. arrival rate

The proposed analytical approach to the overall mean packet delay evaluation demonstrates perfect agreement with simulation results. We thus conclude that our analysis is very precise in predicting the mean delay values in the client relay wireless system with limited-size queues.

**Acknowledgments.** This work is supported by the Russian Foundation for Basic Research (projects #10-08-01071-a and #11-08-01016-a) as well as by the Branch of Nano- and Information Technologies of Russian Academy of Sciences (project 2.3).

## References

1. LTE Release 10 & beyond (LTE-Advanced)
2. IEEE Std 802.16j-2009, IEEE Standard for Local and metropolitan area networks – Multihop Relay Specification
3. IEEE Std 802.16m. Part 16: Air Interface for Broadband Wireless Access Systems – Advanced Air Interface
4. Ibrahim, A., Sadek, A., Su, W., Liu, K.: Cooperative communications with relay-selection: when to cooperate and whom to cooperate with? IEEE Transactions on Wireless Communications 7, 2814–2827 (2008)
5. Jing, Y., Jafarkhani, H.: Single and multiple relay selection schemes and their achievable diversity orders. IEEE Transactions on Wireless Communications 8, 1414–1423 (2009)
6. Yang, Y., Hu, H., Xu, J., Mao, G.: Relay technologies for WiMax and LTE-advanced mobile systems. IEEE Communications Magazine 47, 100–105 (2009)
7. Hossain, E., Kim, D.I., Bhargava, V.K.: Cooperative Cellular Wireless Networks. Cambridge University Press, Cambridge (2011); Technology & Engineering
8. Tannious, R., Nosratinia, A.: Spectrally-efficient relay selection with limited feedback. IEEE Journal on Selected Areas in Communications 26, 1419–1428 (2008)
9. Andreev, S., Galinina, O., Vinel, A.: Performance evaluation of a three node client relay system. International Journal of Wireless Networks and Broadband Technologies 1, 73–84 (2011)
10. Rong, B., Ephremides, A.: On opportunistic cooperation for improving the stability region with multipacket reception. In: NET-COOP Conference, pp. 45–59 (2009)
11. Pyattaev, A., Andreev, S., Galinina, O., Koucheryavy, Y.: System-level evaluation of opportunistic client cooperation in wireless cellular networks. In: IEEE ICCCN Conference (2011)

## Appendix

In the Appendix, we detail the system states and transitions between them. For convenience, we group the states according to the number of packets $(i, j)$ in queues $A$ and $R$. Some transitions are omitted due to the fact that states might be unreachable or the corresponding probability may equal zero.

**Table 1.** Probabilities for the state $(0, 0, 0)$

| From | To | Probability |
|---|---|---|
| $(0,0,0)$ | $(0,0,0)$ | $(1-\pi_A)(1-\pi_R)$ |
| $(1,0,0)$ | $(0,0,0)$ | $(1-\pi_A)(1-\pi_R)p_{AB}$ |
| $(1,0,1)$ | $(0,0,0)$ | $(1-\pi_A)(1-\pi_R)p_{CB}$ |
| $(0,1,0)$ | $(0,0,0)$ | $(1-\pi_R)(1-\pi_A)p_{RB}$ |

**Table 2.** Probabilities for the states $(i, 0, 0)$ and $(i, 0, 1)$

| From | To | Probability |
|---|---|---|
| $(i,0,0)$ | $(i,0,0)$ | $(1-p_{AR})(1-p_{AB})(1-\pi_A)(1-\pi_R)+p_{AB}\pi_A(1-\pi_R)$ |
| $(i,0,1)$ | $(i,0,0)$ | $p_{CB}\pi_A(1-\pi_R)$ |
| $(i,0,0)$ | $(i,0,1)$ | $p_{AR}(1-p_{AB})(1-\pi_A)(1-\pi_R)$ |
| $(i,0,1)$ | $(i,0,1)$ | $(1-p_{CB})(1-\pi_A)(1-\pi_R)$ |
| $(i,1,0)$ | $(i,0,0)$ | $\frac{1}{2}(1-\pi_A)(1-\pi_R)p_{RB}$ |
| $(i,1,1)$ | $(i,0,1)$ | $\frac{1}{2}(1-\pi_A)(1-\pi_R)p_{RB}$ |
| $(i+1,0,0)$ | $(i,0,0)$ | $p_{AB}(1-\pi_A)(1-\pi_R)$ |
| $(i+1,0,1)$ | $(i,0,0)$ | $p_{CB}(1-\pi_A)(1-\pi_R)$ |
| $(i-1,0,0)$ | $(i,0,0)$, $i>1$ | $\pi_A(1-p_{AR})(1-p_{AB})(1-\pi_R)$ |
| $(i-1,0,0)$ | $(i,0,1)$, $i>1$ | $\pi_A p_{AR}(1-p_{AB})(1-\pi_R)$ |
| $(i-1,0,1)$ | $(0,0,1)$, $i>1$ | $\pi_A(1-p_{CB})(1-\pi_R)$ |
| $(i-1,1,0)$ | $(i,0,0)$, $i>1$ | $\frac{1}{2}\pi_A(1-\pi_R)p_{RB}$ |
| $(i-1,1,1)$ | $(i,0,1)$, $i>1$ | $\frac{1}{2}\pi_A(1-\pi_R)p_{RB}$ |
| $(0,0,0)$ | $(1,0,0)$ | $\pi_A(1-\pi_R)$ |
| $(0,1,0)$ | $(1,0,0)$ | $\pi_A(1-\pi_R)p_{RB}$ |

**Table 3.** Probabilities for the states $(0, j, 0)$

| From | To | Probability |
|---|---|---|
| (0,j,0) | (0,j,0) | $(1-\pi_A)(1-\pi_R)(1-p_R)+(1-\pi_A)p_R\pi_R$ |
| $(1,j,0)$ | $(0,j,0)$ | $\frac{1}{2}(1-\pi_A)p_{AB}(1-\pi_R)$ |
| $(1,j,1)$ | $(0,j,0)$ | $\frac{1}{2}(1-\pi_A)p_{CB}(1-\pi_R)$ |
| $(0,j+1,0)$ | $(0,j,0)$ | $(1-\pi_A)(1-\pi_R)p_R$ |
| $(0,j-1,0)$ | $(0,j,0)$, $j>1$ | $(1-\pi_A)\pi_R(1-p_R)$ |
| $(1,j-1,0)$ | $(0,j,0)$, $j>1$ | $\frac{1}{2}\pi_R(1-\pi_A)p_{AB}$ |
| $(1,j-1,1)$ | $(0,j,0)$, $j>1$ | $\frac{1}{2}\pi_R(1-\pi_A)p_{CB}$ |
| $(0,0,0)$ | $(0,1,0)$ | $(1-\pi_A)\pi_R$ |
| $(1,0,0)$ | $(0,1,0)$ | $\pi_R(1-\pi_A)p_{AB}$ |
| $(1,0,1)$ | $(0,1,0)$ | $\pi_R(1-\pi_A)p_{CB}$ |

**Table 4.** Probabilities for the states $(i, j, 0)$ and $(i, j, 1)$

| From | To | Probability |
|---|---|---|
| $(i,j,0)$ | $(i,j,0)$ | $\frac{1}{2}(1-\pi_A)(1-\pi_R)(1-p_{RB}) +$ $+\frac{1}{2}(1-\pi_A)(1-\pi_R)(1-p_{AB})(1-p_{AR}) +$ $+\frac{1}{2}\pi_A p_{AB}(1-\pi_R)+\frac{1}{2}\pi_R p_{RB}(1-\pi_A)$ |
| $(i,j,0)$ | $(i,j,1)$ | $\frac{1}{2}(1-\pi_A)(1-\pi_R)(1-p_{AB})p_{AR}$ |
| $(i,j,1)$ | $(i,j,0)$ | $\frac{1}{2}\pi_A(1-\pi_R)p_{CB}$ |
| $(i,j,1)$ | $(i,j,1)$ | $(1-\pi_A)\left((1-\pi_R)\left(1-\frac{1}{2}p_{CB}-\frac{1}{2}p_{RB}\right)+\frac{1}{2}\pi_R p_{RB}\right)$ |
| $(i,j-1,0)$ | $(i,j,0)$, $j>1$ | $\frac{1}{2}\pi_R((1-\pi_A)((1-p_{AB})(1-p_{AR})+(1-p_{RB}))+p_{AB}\pi_A)$ |
| $(i,j-1,0)$ | $(i,j,1)$, $j>1$ | $\frac{1}{2}(1-\pi_A)\pi_R(1-p_{AB})p_{AR}$ |
| $(i,j-1,1)$ | $(i,j,0)$, $j>1$ | $\frac{1}{2}\pi_A p_{CB}\pi_R$ |
| $(i,j-1,1)$ | $(i,j,1)$, $j>1$ | $\frac{1}{2}(1-\pi_A)\pi_R(1-p_{CB})+\frac{1}{2}(1-\pi_A)\pi_R(1-p_{RB})$ |
| $(i-1,j,0)$ | $(i,j,0)$, $i>1$ | $\frac{1}{2}\pi_A(1-\pi_R)((1-p_{RB})+(1-p_{AB})(1-p_{AR}))+\pi_R p_{RB})$ |
| $(i-1,j,0)$ | $(i,j,1)$, $i>1$ | $\frac{1}{2}\pi_A(1-\pi_R)(1-p_{AB})p_{AR}$ |
| $(i-1,j,1)$ | $(i,j,1)$, $i>1$ | $\pi_A(1-\pi_R)\left(1-\frac{1}{2}p_{CB}-\frac{1}{2}p_{RB}\right)+\frac{1}{2}\pi_A\pi_R p_{RB}$ |
| $(i-1,j-1,0)$ | $(i,j,0)$, $i,j>1$ | $\frac{1}{2}\pi_A\pi_R(1-p_{RB})+\frac{1}{2}\pi_A\pi_R(1-p_{AB})(1-p_{AR})$ |
| $(i-1,j-1,0)$ | $(i,j,1)$, $i,j>1$ | $\frac{1}{2}\pi_A\pi_R(1-p_{AB})p_{AR}$ |
| $(i-1,j-1,1)$ | $(i,j,1)$, $i,j>1$ | $\pi_A\pi_R\left(1-\frac{1}{2}p_{CB}-\frac{1}{2}p_{RB}\right)$ |
| $(i,j+1,0)$ | $(i,j,0)$ | $\frac{1}{2}(1-\pi_A)(1-\pi_R)p_{RB}$ |
| $(i,j+1,1)$ | $(i,j,1)$ | $\frac{1}{2}(1-\pi_A)(1-\pi_R)p_{RB}$ |
| $(i-1,j+1,0)$ | $(i,j,0)$ | $\frac{1}{2}\pi_A(1-\pi_R)p_{RB}$ |
| $(i-1,j+1,1)$ | $(i,j,1)$ | $\frac{1}{2}\pi_A(1-\pi_R)p_{RB}$ |
| $(i+1,j,0)$ | $(i,j,0)$ | $\frac{1}{2}(1-\pi_A)p_{AB}(1-\pi_R)$ |
| $(i+1,j,1)$ | $(i,j,0)$ | $\frac{1}{2}(1-\pi_A)p_{CB}(1-\pi_R)$ |
| $(i+1,j-1,1)$ | $(i,j,0)$, $j>1$ | $\frac{1}{2}(1-\pi_A)p_{AB}\pi_R$ |
| $(i+1,j-1,1)$ | $(i,j,0)$, $j>1$ | $\frac{1}{2}(1-\pi_A)p_{CB}\pi_R$ |
| $(i+1,0,0)$ | $(i,1,0)$ | $(1-\pi_A)p_{AB}\pi_R$ |
| $(i+1,0,1)$ | $(i,1,0)$ | $(1-\pi_A)p_{CB}\pi_R$ |
| $(i,0,0)$ | $(i,1,0)$ | $(1-\pi_A)\pi_R(1-p_{AB})(1-p_{AR})+\pi_A p_{AB}\pi_R$ |
| $(i,0,0)$ | $(i,1,1)$ | $(1-\pi_A)\pi_R(1-p_{AB})p_{AR}$ |
| $(i,0,1)$ | $(i,1,0)$ | $\pi_A p_{CB}\pi_R$ |
| $(i,0,1)$ | $(i,1,1)$ | $(1-\pi_A)\pi_R(1-p_{CB})$ |
| $(i-1,0,0)$ | $(i,1,0)$, $i>1$ | $\pi_A\pi_R(1-p_{AB})(1-p_{AR})$ |
| $(i-1,0,0)$ | $(i,1,1)$, $i>1$ | $\pi_A\pi_R(1-p_{AB})p_{AR}$ |
| $(i-1,0,1)$ | $(i,1,1)$, $i>1$ | $\pi_A\pi_R(1-p_{CB})$ |
| $(0,j,0)$ | $(1,j,0)$ | $\pi_A(1-\pi_R)(1-p_{RB})+\pi_A\pi_R p_{RB}$ |
| $(0,j-1,0)$ | $(1,j,0)$, $j>1$ | $\pi_A\pi_R(1-p_{RB})$ |
| $(0,j+1,0)$ | $(1,j,0)$ | $\pi_A(1-\pi_R)p_{RB}$ |
| $(0,0,0)$ | $(1,1,0)$ | $\pi_A\pi_R$ |

**Table 5.** Probabilities for the modified boundary states $(N, j, \cdot)$, $j < N$

| From | To | Probability |
|---|---|---|
| $(N,0,0)$ | $(N,0,0)$ | $(1-p_{AR})(1-p_{AB})(1-\pi_R)+p_{AB}\pi_A(1-\pi_R)$ |
| $(N,0,0)$ | $(N,0,1)$ | $p_{AR}(1-p_{AB})(1-\pi_R)$ |
| $(N,0,1)$ | $(N,0,1)$ | $(1-p_{CB})(1-\pi_R)$ |
| $(N,1,0)$ | $(N,0,0)$ | $\frac{1}{2}(1-\pi_R)p_{RB}$ |
| $(N,1,1)$ | $(N,0,1)$ | $\frac{1}{2}(1-\pi_R)p_{RB}$ |
| $(N,0,0)$ | $(N,1,0)$ | $\pi_R(1-p_{AB})(1-p_{AR})+\pi_A p_{AB}\pi_R$ |
| $(N,0,0)$ | $(N,1,1)$ | $\pi_R(1-p_{AB})p_{AR}$ |
| $(N,0,1)$ | $(N,1,1)$ | $\pi_R(1-p_{CB})$ |
| $(1,N,0)$ | $(0,N,0)$ | $\frac{1}{2}(1-\pi_A)p_{AB}$ |
| $(1,N,1)$ | $(0,N,0)$ | $\frac{1}{2}(1-\pi_A)p_{CB}$ |
| $(N,j,0)$ | $(N,j,0)$ | $\frac{1}{2}(\pi_A p_{AB}+\pi_R p_{RB}+(1-\pi_R)((1-p_{RB})+(1-p_{AB})(1-p_{AR})))$ |
| $(N,j,0)$ | $(N,j,1)$ | $\frac{1}{2}(1-\pi_R)(1-p_{AB})p_{AR}$ |
| $(N,j,1)$ | $(N,j,0)$ | $\frac{1}{2}\pi_A(1-\pi_R)p_{CB}$ |
| $(N,j,1)$ | $(N,j,1)$ | $(1-\pi_R)(1-\frac{1}{2}p_{CB}-\frac{1}{2}p_{RB})+\frac{1}{2}\pi_R p_{RB}$ |
| $(N,j-1,0)$ | $(N,j,0),j>1$ | $\frac{1}{2}\pi_R((1-p_{AB})(1-p_{AR})+(1-p_{RB}))+\frac{1}{2}\pi_A p_{AB}$ |
| $(N,j-1,0)$ | $(N,j,1),j>1$ | $\frac{1}{2}\pi_R(1-p_{AB})p_{AR}$ |
| $(N,j-1,1)$ | $(N,j,0),j>1$ | $\frac{1}{2}\pi_A p_{CB}$ |
| $(N,j-1,1)$ | $(N,j,1),j>1$ | $\pi_R\left(1-\frac{1}{2}p_{CB}-\frac{1}{2}p_{RB}\right)$ |
| $(N,j+1,1)$ | $(N,j,1)$ | $\frac{1}{2}(1-\pi_R)p_{RB}$ |
| $(N,j+1,0)$ | $(N,j,0)$ | $\frac{1}{2}(1-\pi_R)p_{RB}$ |

**Table 6.** Probabilities for the modified boundary states $(i, N, \cdot)$, $i < N$

| From | To | Probability |
|---|---|---|
| $(i,N,0)$ | $(i,N,0),\ j>1$ | $\frac{1}{2}\pi_A p_{AB}+\frac{1}{2}(1-\pi_A)(1-p_{RB}+p_{RB}\pi_R+(1-p_{AB})(1-p_{AR}))$ |
| $(i,N,0)$ | $(i,N,1),\ i>0$ | $\frac{1}{2}(1-\pi_A)(1-p_{AB})p_{AR}$ |
| $(i,N,1)$ | $(i,N,0),\ i>0$ | $\frac{1}{2}\pi_A p_{CB}$ |
| $(i,N,1)$ | $(i,N,1),\ i>0$ | $(1-\pi_A)\left(1-\frac{1}{2}p_{CB}-\frac{1}{2}p_{RB}+\frac{1}{2}\pi_R p_{RB}\right)$ |
| $(i+1,N,0)$ | $(i,N,0)$ | $\frac{1}{2}(1-\pi_A)p_{AB}$ |
| $(i+1,N,1)$ | $(i,N,0)$ | $\frac{1}{2}(1-\pi_A)p_{CB}$ |
| $(i-1,N,0)$ | $(i,N,0),\ i>1$ | $\frac{1}{2}\pi_A(1-p_{RB}+\pi_R p_{RB}+(1-p_{AB})(1-p_{AR}))$ |
| $(i-1,N,0)$ | $(i,N,1),\ i>1$ | $\frac{1}{2}\pi_A(1-p_{AB})p_{AR}$ |
| $(i-1,N,1)$ | $(i,N,1),\ i>1$ | $\pi_A\left(1-\frac{1}{2}p_{CB}-\frac{1}{2}p_{RB}+\frac{1}{2}\pi_R p_{RB}\right)$ |
| $(0,N,0)$ | $(1,N,0)$ | $\pi_A(1-p_{RB}+p_{RB}\pi_R)$ |
| $(0,N,0)$ | $(0,N,0)$ | $(1-\pi_A)(1-p_{RB}+p_{RB}\pi_R)$ |

**Table 7.** Probabilities for the modified boundary states $(N, N, \cdot)$

| From | To | Probability |
|---|---|---|
| $(N,N,0)$ | $(N,N,0)$ | $\frac{1}{2}(1-p_{RB}+\pi_A p_{AB}+\pi_R p_{RB})+(1-p_{AB})(1-p_{AR})$ |
| $(N,N,0)$ | $(N,N,1)$ | $\frac{1}{2}(1-p_{AB})p_{AR}(\pi_A+\pi_R-\pi_A\pi_R)$ |
| $(N,N,1)$ | $(N,N,0)$ | $\frac{1}{2}\pi_A p_{CB}$ |
| $(N,N,1)$ | $(N,N,1)$ | $\left(1-\frac{1}{2}p_{CB}-\frac{1}{2}p_{RB}\right)+\frac{1}{2}p_{RB}\pi_R$ |

# Discrete Markov Chain Model for Analyzing Probability Measures of P2P Streaming Network*

Aminu Adamu, Yuliya Gaidamaka, and Andrey Samuylov

Peoples' Friendship University of Russia,
Telecommunication Systems Department, Ordzhonikidze str. 3
115419 Moscow, Russia
{Aminu,ygaidamaka}@mail.ru, asam1988@gmail.com

**Abstract.** In this paper, we present a model of the data exchange process between users in P2P live streaming network with buffering mechanism - buffer occupancy model. The model is developed in terms of discrete Markov chain. We study the probability measures of the model – buffer position occupancy probability and probability of playback continuity. The model is very tractable and compacted formulas were obtained. While considering user churn in the model we examine how user departure affects buffer occupancy and playback continuity. For our numerical experiment we develop a simulation tool for the considered model.

**Keywords:** P2P network, live streaming, buffer occupancy, playback continuity, Markov chain model.

## 1 Introduction

P2P technology is being used in many networks for the provision of live video streaming services, in order to effectively utilize the available resources of each user in the network [1,2]. Any user in these networks uses his download bandwidth to receive video data from the network and his upload bandwidth to distribute video data to other users in the network, thus simultaneously playing the role of a client that downloads (receives) data and the role of a server which provides data. The focus of [3-6] was to derive one of the main QoE parameters of P2P live streaming network, i.e. the probability that all users receive video stream at a rate not less than the defined playback rate. One of the factors that affect the quality of perception at user level is playback continuity, i.e., a user is watching video without pauses during playback. To ensure a smooth video playback a buffering mechanism is used. In P2P live streaming networks video stream is divided into small blocks of data, known as video chunks, each video chunk has a length of about 1 second, and the end user's device, which may be a set-top-box or a personal computer, contains a buffer for storing these chunks of video data in order of their playback time. When a new user joins a video session, he first has to start downloading chunks from the network to fill his buffer

* This work was supported in part by the Russian Foundation for Basic Research (grant 10-07-00487-a).

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 428–439, 2011.

and only after that chunks from the buffer are used to play the video. If a user is missing a certain chunk, he will try to download that particular chunk from other users before its playback deadline.

In this paper we present a discrete Markov chain model for buffer state change in P2P live streaming networks along with a method for finding the probability of playback continuity. For the numerical analysis due to high dimensionality of the Markov chain state space we developed a simulation tool.

## 4 Data Exchange Process

In this section we describe the procedure for distribution of video data in P2P streaming networks, taking into account buffering mechanism. Consider a network with N users present in the network, and a single server, which transmit only one video stream, i.e. all users in the network, are receiving the same video stream. The process of video stream playback is divided into time slots, the length of each time slot corresponds to the playback time of one chunk. Assume that each user in the network has a buffer designed to accommodate M+1 chunks. The buffer positions are numbered from 0 to M: buffer 0-position is to store the most recent (freshest) chunk just received from the server, other $m$-positions, $m = 1, \cdots, \mathrm{M} - 1$ are to store chunks, already received during the past time slots or will be downloaded in the coming time slots, and buffer M-position is to store the oldest chunk that will be moved out from the buffer to the player for playback during the next time slot.

Now we will specify the actions that the server and users perform during each time slot. At the beginning of each time slot the server randomly selects a user from the network and uploads the most recent chunk into his buffer 0-position. Each of $\mathrm{N} - 1$ users that were not chosen by the server during the current time slot will perform the following actions. If there are empty positions in the user's buffer, i.e. there are missing chunks in his buffer, the user will choose another user from the network randomly (called a target user) in order to download one of the missing chunks from him. If it happens that, the target user has one of the missing chunks, then the attempt to download from the target user will be successful. If the target user has more than one of the missing chunks, then downloading strategy will define which chunk to download. Most commonly used strategies in P2P networks are Rarest First (RF) and Greedy (Gr) strategy. With Rarest First strategy users during any time slot will try to download the most recent (freshest) chunk, i.e. the most rarely encountered chunk in the network (chunk with least available copies), and with Greedy strategy users will select to download a chunk with a closest playback deadline [6,7]. A user will not download a chunk in the current time slot at all, if the target user he chose does not have any of the missing chunks, or if in the current time slot his buffer is filled (there are no empty positions). At the end of each time slot, chunks in the buffer of each user will shift one step forward, i.e. chunk in M-position will exit out of the buffer and move to the player for playback, the remaining chunks in other positions will shift one position to the right (towards the end of the buffer) to replace the position freed by its predecessor, and in this case buffer 0-position will be free to accommodate a new chunk from the server at the beginning of the next time slot.

In this paper we generalize the model proposed in [7] by considering user behavior. We assume that each user may at some time leave the network and stop exchanging data with other users, and likewise a user may join the network and begin exchanging data with other users. In the next section, we develop a mathematical model for data exchange between users in the form of discrete Markov chain describing the buffer states of all users.

## 3 Model Description

For a given network with N users and a single server, vector $\mathbf{z}(n) = \big(a(n), \mathbf{x}(n)\big)$ defines the state of each user (*n*-user), where $a(n)$ is user on-line indicator ($a(n) = 1$ if the user is on-line and $a(n) = 0$ otherwise) and $\mathbf{x}(n) = \big(x_0(n), x_1(n), \cdots, x_{\mathrm{M}}(n)\big)$ is the state of *n*-user's buffer. Here $x_m(n)$ is the state of *n*-user's buffer *m*-position: $x_m(n) = 1$, if *n*-user's buffer *m*-position is occupied with a chunk, otherwise $x_m(n) = 0$, $m = 0, \cdots, \mathrm{M}$. Each user in the network, uses buffer positions $m = 1, \cdots, \mathrm{M}$ to store the downloaded chunks from the network, and uses 0-position only to download a chunk from the server. Thus, the oldest chunk in the buffer, which will be sent to the video player for playback during the next time slot, is located in M-position during the current time slot, and chunk in *m*-position will be sent to the player for playback after M-*m* time slots.

Note that, if during any time slot M-position is filled, then *n*-user will watch the video stream without any pause. The introduced notations are illustrated in Fig.1.

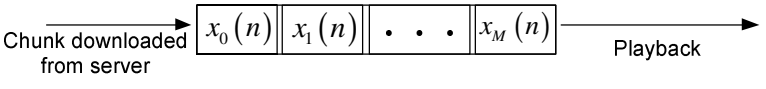



**Fig. 1.** Buffer state of *n*-user

Thus, the state of the system (on-line indicators and buffer states of all users in the network) is defined by $= \big(\mathbf{z}(n)\big) = (\mathbf{a}, \mathbf{X}) = \big(a(n), \mathbf{x}(n)\big)_{n=1,\ldots,\mathrm{N}}$, where the *n*-th row of the matrix $\mathbf{X}$ corresponds to the buffer state of *n*-user, and $\dim\mathbf{X} = \mathrm{N}(\mathrm{M}+1)$. Therefore, the state space of the system is given by $Z = \{0,1\}^{\mathrm{N}} \times \{0,1\}^{\mathrm{N}(\mathrm{M}+1)}$ and $|Z| = 2^{\mathrm{N}(\mathrm{M}+2)}$.

Denote by $M^0\big(\mathbf{x}(n)\big)$ and $M^1\big(\mathbf{x}(n)\big)$ the set of all empty and filled positions in *n*-user's buffer respectively, i.e. $M^0\big(\mathbf{x}(n)\big) = \big\{m\colon x_m(n) = 0,\ m = 1, \cdots, \mathrm{M}\big\}$, $M^1\big(\mathbf{x}(n)\big) = \big\{m\colon x_m(n) = 1,\ m = 1, \cdots, \mathrm{M}\big\}$, where $M^0\big(\mathbf{x}(n)\big) \cup M^1\big(\mathbf{x}(n)\big) = \{1,2, \cdots, \mathrm{M}\}$. Then $M^0\big(\mathbf{x}(n)\big) \cap M^1\big(\mathbf{x}(h)\big)$ will be the set of all positions in buffer to which *n*-user can download a chunk from target *h*-user, $n \neq h$. If $M^0\big(\mathbf{x}(n)\big) \cap M^1\big(\mathbf{x}(h)\big) \neq \emptyset$, then the index $m_\delta\big(\mathbf{x}(n), \mathbf{x}(h)\big)$ of the position to which *n*-user can download a chunk from *h*-user is determined by the downloading strategy in use, i.e.

$$m_\delta\big(\mathbf{x}(n), \mathbf{x}(h)\big) = \begin{cases} \min\big\{m\colon m \in M^0\big(\mathbf{x}(n)\big) \cap M^1\big(\mathbf{x}(h)\big)\big\}, & \text{with RF strategy;} \\ \max\big\{m\colon m \in M^0\big(\mathbf{x}(n)\big) \cap M^1\big(\mathbf{x}(h)\big)\big\}, & \text{with Gr strategy.} \end{cases}$$

Denote by $S\mathbf{x}(n)$ the shifting operator of vector $\mathbf{x}(n)$, meaning if $\mathbf{x}(n) = \left(x_0(n), x_1(n), \cdots, x_{\mathrm{M}-1}(n), x_{\mathrm{M}}(n)\right)$, then $S\mathbf{x}(n)=\left(0, x_0(n), \cdots, x_{\mathrm{M}-1}(n)\right)$.

Let $t_l$ be the shifting moment of buffer contents. When constructing the model in a discrete time it is assumed that if at the moment $t_l - 0$ a buffer is in the state $\mathbf{x}(n)$, then at the moment $t_l + 0$ it will be in the state $S\mathbf{x}(n)$.

We assume that a user can leave the network, or join the network only at the moment $t_l$. Denote by $a^l(n)$ the $n$-user's on-line indicator at the moment $t_l - 0$. If $n$-user joined the network at the moment $t_l$ or earlier and never left the network till the moment $t_l$, then $a^l(n) = 1$ and $a^l(n) = 0$ otherwise. Let $\alpha(n)$ be the probability of $n$-user joining the network and $\beta(n)$ the probability of $n$-user leaving the network:

$$P\{a^{l+1}(n) = 1 | a^l(n) = 0\} = \alpha(n);$$
$$P\{a^{l+1}(n) = 0 | a^l(n) = 0\} = 1 - \alpha(n);$$
$$P\{a^{l+1}(n) = 0 | a^l(n) = 1\} = \beta(n);$$
$$P\{a^{l+1}(n) = 1 | a^l(n) = 1\} = 1 - \beta(n).$$

For simplicity assume that all users join and leave the network with equal probabilities, i.e.

$$\alpha(n) = \alpha, \beta(n) = \beta, n = \overline{1, N}.$$

We also assume that when $n$-user leaves the network then the corresponding row in matrix $\mathbf{X}$ will reset i.e. $\mathbf{x}(n) = \mathbf{0}$.

According to the protocol for the distribution of data in P2P live streaming networks with a buffering mechanism, in the interval $[t_l, t_{l+1})$, which corresponds to the $l$-th time slot, the server and users perform the following actions.

1. At the moment $t_l$, an off-line user decides to join the network with probability $\alpha$ and an on-line user decides to leave the network with probability $\beta$.
2. At the moment $t_l$, for all users the shift of the buffer content takes place:
   - Chunk in buffer M-position if present will be sent for playback;
   - All other chunks in other buffer positions will be shifted one position to the right, i.e. towards the end of the buffer;
   - Buffer 0-position will be emptied.
3. At the moment $t_l + 0$, server chooses one user randomly and uploads a chunk for the current time slot to his buffer 0-position. If server has chosen $n$-user, then $x_0(n) = 1$ at the moment $t_{l+1} - 0$.
4. Each $\hat{n}$-user, $\hat{n} \neq n$ not chosen by the server will perform the followings:
   - If there are empty positions in $\hat{n}$-user's buffer, meaning $M^0\left(\mathbf{x}(\hat{n})\right) \neq \emptyset$, then $\hat{n}$-user will choose a target $h$-user randomly from the network to download a chunk in order to fill one of the empty positions, $h \neq \hat{n}$. Let $M^1\left(\mathbf{x}(h)\right)$ be the set of all $h$-user's buffer positions filled with chunks;
   - If $M^0\left(\mathbf{x}(\hat{n})\right) \cap M^1\left(\mathbf{x}(h)\right) \neq \emptyset$, then the $\hat{n}$-user will select in accordance with the downloading strategy $\delta$, $\delta \in \{RF, Gr\}$ the position $m_\delta\left(\mathbf{x}(\hat{n}), \mathbf{x}(h)\right)$ to which he will download a missing chunk from $h$-user;

- No action will be performed by the $\hat{n}$-user:
  a) If $M^0\big(\mathbf{x}(\hat{n})\big) \cap M^1\big(\mathbf{x}(h)\big) = \emptyset$, i.e. $\hat{n}$-user unsuccessfully chose a target $h$-user;
  b) If $M^0\big(\mathbf{x}(\hat{n})\big) = \emptyset$, i.e. there are no empty positions in the $\hat{n}$-user's buffer.

Denote by $\mathbf{Z}^l = (\mathbf{a}^l, \mathbf{X}^l)$ the network state at the moment $t_l - 0$, as shown in Fig.2. Note that the set $\{\mathbf{Z}^l\} := \{\mathbf{Z}^l,\ l \geq 0\}$ forms a Markov chain over state space $Z = \{0,1\}^{\mathrm{N}} \times \{0,1\}^{\mathrm{N(M+1)}}$, generally speaking the Markov chain is decomposable, with one class $\tilde{Z}$ of essential states, $\tilde{Z} \subset Z$.

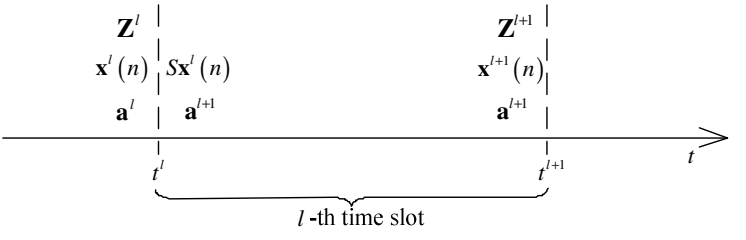



**Fig. 2.** State of Markov chain $\{\mathbf{Z}^l\}$ at $l$-th time slot

Let $\pi^l(\mathbf{Z})$ be the probability that Markov chain $\{\mathbf{Z}^l\}_{l\geq 0}$ during $l$-th time slot is in state $\mathbf{Z}$, i.e. $\pi^l(\mathbf{Z}) = P\{\mathbf{Z}^l = \mathbf{Z}\}$ and $\Pi^{l,l+1}(\mathbf{Z},\mathbf{Y})$ be the corresponding transition probability.

Note that the transition probability $\Pi^{l,l+1}(\mathbf{Z},\mathbf{Y})$ depends on $m_\delta\big(\mathbf{x}(n),\mathbf{x}(h)\big)$, i.e. on the strategy in use, and on $\alpha$ and $\beta$, i.e. on the user joining and leaving probabilities. The probability distribution $\pi^l(\mathbf{Z})$ satisfies the Kolmogorov-Chapman equations:

$$\pi^{l+1}(\mathbf{Y}) = \sum_{\mathbf{X}\in X} \pi^l(\mathbf{Z})\Pi^{l,l+1}(\mathbf{Z},\mathbf{Y}),\ \mathbf{Y}\in Z,\ l \geq 0\,. \tag{1}$$

## 4 Probability Measures Analysis

One of the main performance measures of P2P live streaming networks is the probability $V(n)$ of playback continuity, which is the probability that buffer M-position of $n$-user is filled with a chunk for playback at the end of any time slot. To find this probability, we introduce the $h_n^i(\mathbf{Z})$ function, which corresponds to the number of users who have a chunk in their buffer $i$-position, from which $n$-user can download in accordance with the downloading strategy δ when the network is in state $\mathbf{Z}$:

$$h_n^i(\mathbf{Z}) = \sum_{h=1,\cdots,\mathrm{N}:\, h\neq n,\ a^l(h)=1} \delta_{m_\delta(\mathbf{x}(n),\mathbf{x}(h)),i}\,,\ \mathbf{Z}\in Z,$$

where $\delta_{j,i} = \begin{cases} 1, & j = i\,, \\ 0, & j \neq i\,. \end{cases}$

Let $\mathrm{N}(\mathbf{a}^l) = \sum_{j=\overline{1,\mathrm{N}}} a^l(j)$ be the number of on-line users when the network is in state $\mathbf{Z}^l = (\mathbf{a}^l, \cdot)$. Now we define the probability $Q_n^l(i)$ that during the $l$-th time slot, the chunk which $n$-user can download to his buffer $i$-position is available in the network. Due to the dependency of this probability on the downloading strategy $\delta$ we can interpret $Q_n^l(i)$ as the probability that $n$-user will select $i$-position and successfully download a chunk from the target user during the $l$-time slot. If $\mathrm{N}(\mathbf{a}^l) \geq 2$, then one can obtain the following formula

$$Q_n^l(0) = 0, \;\; Q_n^l(i) = \sum_{\mathbf{Z} \in Z} \pi^l(\mathbf{Z}) \cdot \frac{h_n^i(\mathbf{Z})}{\mathrm{N}(\mathbf{a}^l) - 1}, \;\; i = 1, \cdots, \mathrm{M}. \tag{2}$$

Denote by $p_0^l(n,i)$ ($p_1^l(n,i)$) the probability that $i$-position of $n$-buffer is empty (filled) during $l$-th time slot. These probabilities are defined as follows:

$$P\{x_i^l(n) = 1\} = \sum_{\mathbf{Y} \in Z:\, \mathbf{y}(n) = \mathbf{x}(n), x_i(n) = 1, a^l(n) = 1} \pi^l(\mathbf{Y}) =: p_1^l(n,i),$$
$$P\{x_i^l(n) = 0\} = \sum_{\mathbf{Y} \in Z:\, \mathbf{y}(n) = \mathbf{x}(n), x_i(n) = 0, a^l(n) = 1} \pi^l(\mathbf{Y}) =: p_0^l(n,i).$$

The formula that relates the probability of $n$-buffer state during ($l$+1)-th time slot and the probability of $n$- buffer state during $l$-th time slot can be obtained from the following relation:

$$\begin{aligned} P\{x_{i+1}^{l+1}(n) = 1\} &= P\{x_i^l(n) = 1, a^l(n) = 1\} \cdot (1 - \beta) + \\ &+ P\{x_i^l(n) = 0, a^l(n) = 1\} \cdot (1 - \beta) \cdot Q_n^l(i) + \\ &+ P\{x_i^l(n) = 1, a^l(n) = 0\} \cdot \alpha + P\{x_i^l(n) = 0, a^l(n) = 0\} \cdot \alpha \cdot Q_n^l(i), \\ i &= 1, \cdots, M. \end{aligned} \tag{3}$$

By considering $P\{x_i^l(n) = 1, a^l(n) = 0\} = 0$ and $P\{x_i^l(n) = 0, a^l(n) = 0\} = 1$ in formula (3), we can obtain a recursive relation for calculating the buffer state probabilities:

$$\begin{aligned} p_1^l(n,0) &= \frac{1}{\mathrm{N}}, \\ p_1^{l+1}(n, i+1) &= p_1^l(n,i) \cdot (1 - \beta) + \\ &+ p_0^l(n,i) \cdot (1 - \beta) \cdot Q_n^l(i) + \alpha \cdot Q_n^l(i), \\ i &= 0, \cdots, \mathrm{M} - 1. \end{aligned} \tag{4}$$

Assume that the equilibrium distribution of the Markov chain $\{\mathbf{Z}^l\}$ exists. Denote by $p_1(n,i) = \lim_{l \to \infty} p_1^l(n,i)$ the probability that $i$-position of $n$-buffer is filled and by $p_0(n,i) = \lim_{l \to \infty} p_0^l(n,i)$ the probability that $i$-position of $n$-buffer is empty. Then from formula (4) we obtain the following:

$$
\begin{aligned}
& p_1\ (n,0) = \frac{1}{\mathrm{N}}, \\
& p_1\ (n,i+1) = p_1\ (n,i) \cdot (1-\beta) + \\
& \quad + p_0\ (n,i) \cdot (1-\beta) \cdot Q_n\ (i) + \alpha \cdot Q_n\ (i), \\
& i = 0, \cdots, \mathrm{M}-1 .
\end{aligned}
\tag{5}
$$

Note that the recursive relation (5) gives a method for calculating the probability $p_1\ (n, \mathrm{M})$, that M-position of *n*-buffer is filled. Denote by $V(n)$ the probability that *n*-user is watching video without pauses during playback, i.e. probability of playback continuity, then we have the following formula

$$V(n) = p_1\ (n, \mathrm{M}) . \tag{6}$$

Hence the desired probability measures of the considered model are obtained, and in the next section we provide some case studies for P2P live streaming network performance analysis.

## 5 Case Study

For our case study we have developed the simulation model, because of some computational difficulties using the formulas derived in the previous sections of the paper. Note that, to calculate the probability $p_1\ (n, \mathrm{M})$ using formula (5) first it is necessary to calculate $Q_n^l(i)$ using formula (2), but to do so one has to calculate the equilibrium probabilities $\pi^l(\mathbf{Z})$ of Markov chain $\{\mathbf{Z}^l\}_{l\geq 0}$ using formula (1). So the calculation of the probabilities $\pi^l(\mathbf{Z})$ has to be carried out over a state space of a dimension $|Z| = 2^{\mathrm{N}(\mathrm{M}+2)}$. We should notice that this state space dimension for P2P live streaming networks will be very high, because in real networks the number of concurrent users may reach up to $10^5$ and enumerate up to $10^2$ buffer positions. That is why calculations using formulas derived in this paper are rather difficult to be carried out. For our case study we chose N=1000 and M=40. It is not a subject of the paper to develop a numerical method to resolve the computational difficulties. Therefore we did our case study using a simulation model.

We begin with a case where all users are present in the network and do not depart, i.e. $\alpha = 0$ and $\beta = 0$. Fig. 3 presents the graphs of $Q_n\ (i)$ for both considered strategies (Rarest First and Greedy). Note that in [4] approximate formulas for $Q_n\ (i)$ were proposed for both strategies, whereas in this paper we present the exact formulas for $Q_n\ (i)$. Moreover the result of our calculations with exact formulas is much different from those with the approximate formulas presented in [4].

In Fig. 3 we can notice that the graph of the $Q_n\ (i)$ probability for RF strategy increases on the interval of *i*=[0,10], because the number of target users with these

buffer positions filled increases. But on the interval of $i$=[11,39] this probability for RF strategy goes down, because most users have already managed to fill these buffer positions in the past in accordance with the RF strategy. For Greedy strategy the probability $Q_n\ (i)$ on the interval of $i$=[0,35] increases because for most users these positions are empty, with Greedy strategy users target chunks with the closest playback deadline, but on the interval of $i$=[36,39] this probability decreases because most users have already been able to download chunks for these buffer positions previously.

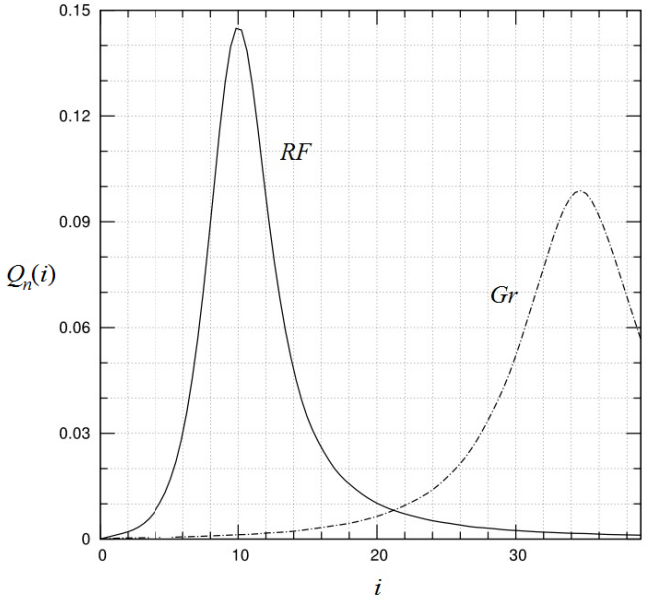



**Fig. 3.** Buffer position selection probability, $\alpha = 0$ and $\beta = 0$

Having studied the buffer position selection probability $Q_n\ (i)$ it is interesting to study the buffer position occupancy probability $p_1\ (n,i)$ for both strategies, i.e. the probability that a buffer position is filled. The buffer position occupancy probability enables us to evaluate playback continuity. The graph of $p_1\ (n,i)$ for both strategies is provided in Fig. 4.

From the graphs in Fig.4 one can notice that $p_1^{RF}(n,i) > p_1^{Gr}(n,i)$, for all buffer positions. Moreover one can observe that the probability of playback continuity $V(n) = p_1\ (n,\mathrm{M})$ (the probability that the last buffer position is filled with a chunk) for RF strategy is always higher than that of Greedy strategy, $V^{RF} > V^{Gr}$. Based on these arguments it is apparent that the RF strategy is better in terms of playback continuity.

It is interesting to examine the effect that the user churn has on user's buffer condition. Consider a network with users joining and leaving the network. Assume

that an off-line user joins the network with probability $\alpha > 0$ and an on-line user leaves the network with probability $\beta > 0$ as described in section 3 of the paper. We further focus our analysis on RF strategy due to its obvious advantage over the Greedy strategy in terms of playback continuity.

It is important to note that, the connection of new users with empty buffers to the network, likewise the disconnection of users from the network with partially or fully filled buffer decreases the number of target users with $i$-position filled. For further analysis we fix the joining probability $\alpha$, and vary the leaving probability $\beta$.

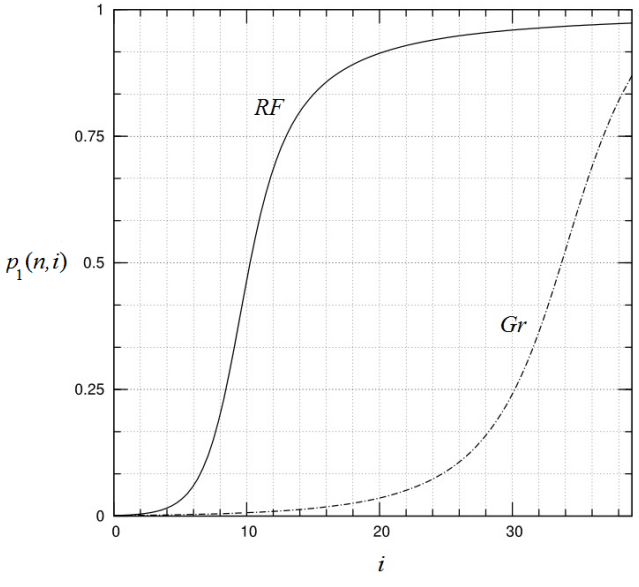



**Fig. 4.** Buffer position occupancy probability, $\alpha = 0$ and $\beta = 0$

From the graphs in Fig. 5 one can notice that as the probability $\beta$ increases (meaning more user departure) the probability $Q_n(i)$ of selecting and successfully downloading chunk to buffer position decreases. However, the graphs in Fig. 5 have almost the same pattern with the graph for RF strategy in Fig. 3. The behavior of the graphs in Fig. 5, shows that with a fixed joining probability, the increase of leaving probability leads to low circulation of video chunks in the network. Thus the proportion of users with more empty positions will rise, thereby reducing the probability of successful download. Note that the graph of $Q_n(i)$ for $\beta = 0{,}05$ has almost aligned due to the high frequency of user departure and therefore few chunks are available for downloading.

Similar to the behavior of graphs for $Q_n(i)$ in Fig. 5, with the increase of probability $\beta$, the average number of users $\mathrm{N}_{avg}$ in the network decreases (Fig. 6), likewise the probability of playback continuity $V$ also decreases (Fig. 7).

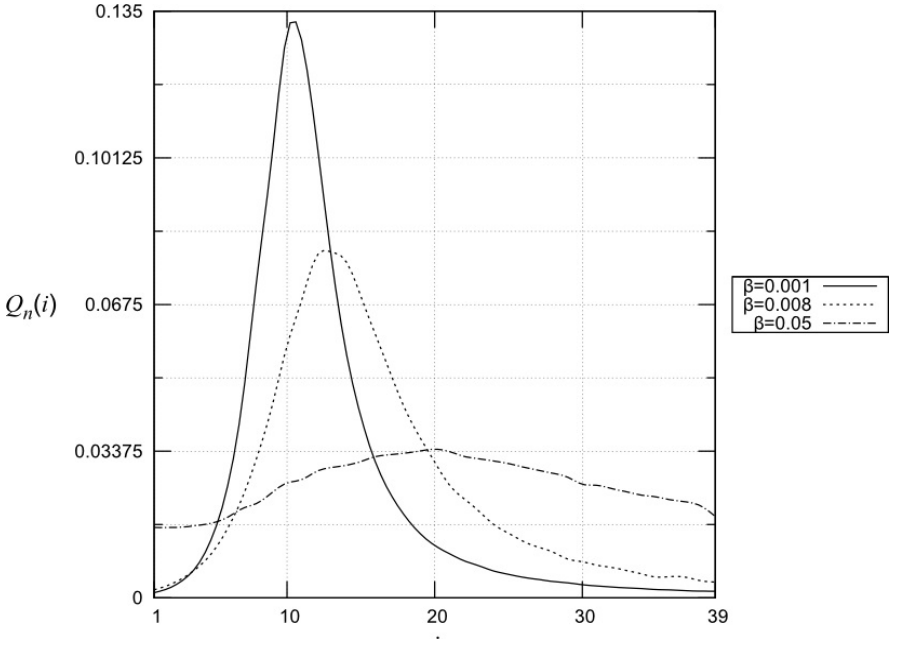



**Fig. 5.** Buffer position selection probability, $\alpha = 0{,}001$

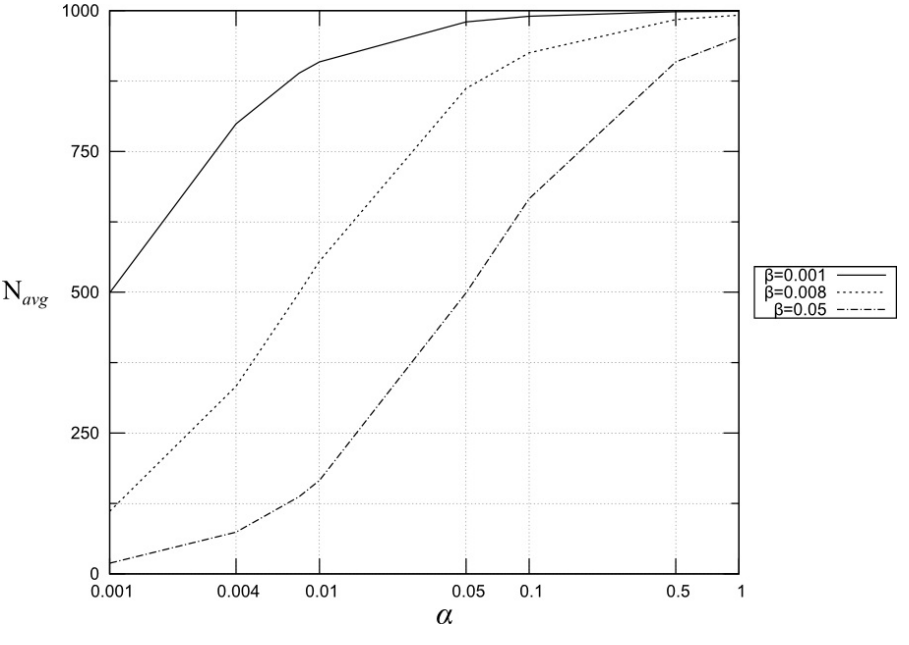



**Fig. 6.** Average number of users in the network

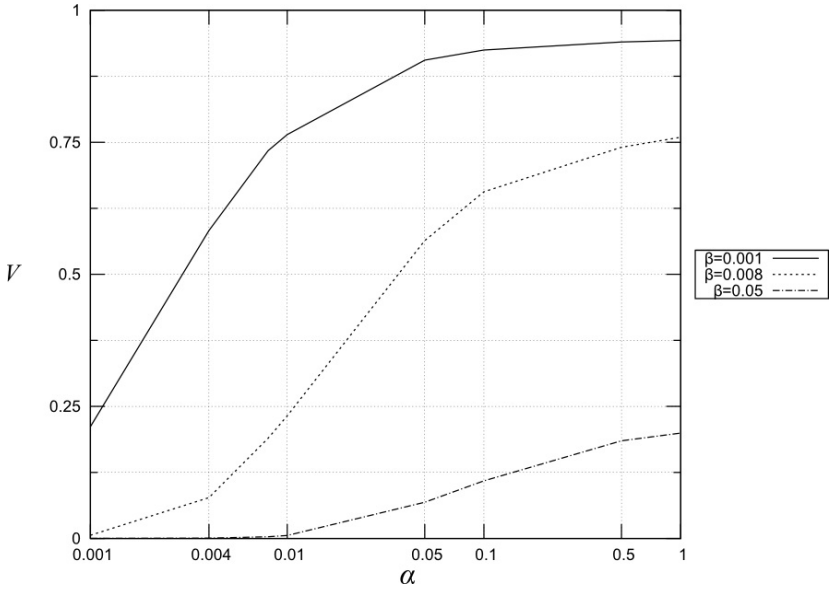



**Fig. 7.** Probability of playback continuity

## 6 Conclusion

In this paper the model of data exchange between users of P2PTV live streaming network was presented. The model was developed in terms of discrete Markov chain, through which the formulas for the analysis of the system performance measures were obtained. For the state space of small dimension N=3, M=3 and $|Z| = 2^{15}$ in case when α=0 and β=0, we obtained the transition matrix for Markov chain and calculated the probability measures using the exact formulas (1). The calculations results of the exact formulas coincide with the results of simulation. The direction for our further studies will be to develop a numerical method for a network with high dimension state space and to develop a model taking into account the server upload rate and the upload rate of users. Besides the playback continuity, we will concentrate our future work on the analysis of another performance metric – startup latency, based on both downloading strategies, i.e. $\delta \in \{RF, Gr\}$.

We thank professor K.E. Samouylov for comments that greatly improved the manuscript.

## References

1. Shen, X., Yu, H., Buford, J., Akon, M.: Handbook of Peer-to-Peer Networking, p. 1421. Springer, Heidelberg (2010)
2. Setton, E., Girod, B.: Peer-to-Peer Video Streaming, p. 150. Springer, Heidelberg (2007)

3. Adamu, A., Gaidamaka, Y., Samuylov, A.: Analytical Modeling of P2PTV Network. In: Proc. of the 2d International Congress on Ultra Modern Telecommunications and Control Systems (IEEE ICUMT 2010), Moscow, Russia, October 18-20, pp. 1115–1120 (2010)
4. Hei, X., Liang, C., Liang, J., Liu, Y., Ross, K.: A measurement study of a large-scale P2P IPTV System. IEEE Trans. Multimedia 9(8), 1672–1687 (2007)
5. Wu, D., Liu, Y., Ross, K.W.: Queuing Network Models for Multi-Channel Live Streaming Systems. In: Proc. of the 28th Conference on Computer Communications (IEEE Infocom 2009), Rio de Janeiro, Brazil, April 19-25, pp. 73–81 (2009)
6. Zhou, Y., Chiu, D.M., Lui, J.C.S.: A Simple Model for Analyzing P2P Streaming Protocols. In: Proc. of IEEE Int. Conf. IN Network Protocols (ICNP 2007), October 19, pp. 226–235 (2007)
7. Aminu, A., Gaidamaka, Y., Samuylov, A.: Markov chain model for analyzing P2P streaming network. T-Comm - Telecommunications and Transport (5) (2011)

# A Mathematical Framework for the Multidimensional QoS in Cognitive Radio Networks

Jerzy Martyna

Institute of Computer Science, Jagiellonian University,
ul. Prof. S. Łojasiewicza 6, 30-348 Cracow, Poland

**Abstract.** In cognitive radio networks multiple secondary users share a single channel and multiple channels are simultaneously used by a single secondary user (SU) in order to satisfy their rate requirement. In such environment, we attempt to evaluate the relations between different dimensions of the multiple QoS. By means of the mathematical modelling framework for studying the versions QoS constraints in the CR network introduced here, we have determined the throughput, the delay, and the bandwidth in a uniform formula. As an example application, the loss rate of the secondary users in the CR network as a function of various parameters, etc. are presented for a realistic network setup.

**Keywords:** cognitive radio networks, multidimensional QoS, performance evaluation.

## 1 Introduction

Cognitive radio (CR) networks were introduced in order to improve the utilization efficiency of the existing radio spectrum [8], [1]. The key of this concept lies in the dynamic adjusting of the spectrum in such way that the under-utilized spectrum is better applied than without this technique. The CR networks are assumed to be able to provide the high bandwidth to mobile users via heterogeneous wireless architectures and dynamic spectrum access techniques.

In a CR network the unlicensed users (also called the secondary users - SU) are allowed to opportunistically access the radio spectrum allocated to the licensed users (also called the primary users - PU) without causing any harmful interference to the licensed users. Several mechanizms are proposed for the implementation of this mechanism. Among others, the so-called *dynamic spectrum access* (DSA) for accessing the idle licensed channels was proposed by N. Shah [10]. Detailed DSA schemes are explained in the paper by J. Zhao [13]. In other solutions [4], [11] a distributed protocol MAC for the implementation of the CR networks was proposed. However, the authors of these papers ignored the QoS requirement of the SU or coexistence of multiple SUs in a channel.

The problem of the optimal radio spectrum assignment to the SUs in the CR networks was also studied by L. Cao [3] and Zheng et al. [15]. In the first paper local bargaining was proposed. In the other a graph-theoretical model for

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 440–449, 2011.

characterization of the spectrum access was proposed and a heuristic to find the fair spectrum suggested. Based on this model, they designed several centralized heuristics in order to find fair spectrum allocation. The problem problem of the optimal radio spectrum assignment to SUs in the CR networks was also studied in the papers [9], [6]. Distributed spectrum allocation methods were presented in the paper by Q. Zhao [14]. However, none of these works formulated the basic dependencies between the basic QoS parameters in the sensing policies and the resource allocation.

The main goal of the paper is to develop a mathematical framework for the QoS constraints modelling in the CR networks. We formulate the uniform formula in order to connect the throughput, the delay, and the loss rate of the data packets in the CR network. Within this framework, we also propose two solution approaches in order to find probability that the time interval $\tau$ is more than a given threshold and the depedence between the throughput and the parameter $\delta$. It allows us to obtain the maximum throughput of the SU in the CR network for the required service.

The rest of this paper is as follows. In section 2 we formulate our model of the CR network. Section 3 provides the multidimensional QoS in the CR network. In section 4 we present a numerical example. Section 5 concludes the paper.

## 2 Modelling of the Multidimensional QoS in Cognitive Radio Networks

In this section we introduce a multi-dimensional QoS framework in the CR networks.

We consider the downlink of a CDMA-based system as a model of the CR network as depicted in Fig. 1. In this figure a two-tier macro/microcell structure is presented. The cognitive microcell uses the same frequency band to transmit non-real time data as the macrocell. The data transmission from the mobile station is disturbed by the macrocell base station $BS_i, (i = 0, 1, \ldots, 6)$. The transmission between mobile station and its microcell base station is realized when the macrocell interference falls bellow a given threshold. Thus, the microcell sends its data with a transmission rate dependent on the macrocell interference level [5]. For such two-tier cognitive system we find a microcell transmission window in order to guarantee the successful transmission with a defined QoS level of data packets.

The received signal-to-inference-plus-noise ratio (SINR) of the $i$-th secondary user in the microcell (the microcell serves $N$ users) can be expressed on a linear scale as

$$SINR_i(t) = \frac{g_i(t)p_i(t)}{\sum_{j=1,j\neq 1}^{N} \alpha_{ij} g_j(t)p_j(t) + \overline{I}(t) + \overline{N}(t)} \tag{1}$$

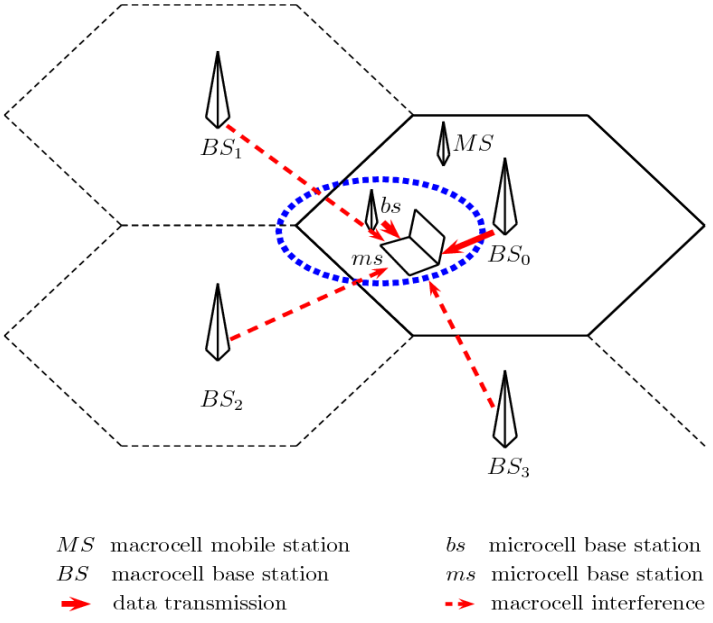



**Fig. 1.** Two-tier macro/microcell structure.

where $g_i(t)$ is the time-varying channel gain between the mobile base station in the microcell and the $i$-th user, $p_i(t)$ is the time-varying transmission power of the base station in the microcell for $i$-th user, $\alpha_{ij}$ is the code cross-correlation between users $i$ and $j$ as seen at the receiver (the loss of code orthogonality), $\overline{I}(t)$ is the total time varying the macrocell interference, $\overline{N}(t)$ is the additive Gaussian noise.

In order to simplify the notation and the analysis of the occuring distribution, we can rewrite Eq. (1) in dB which operates in the logarithmic domain, namely

$$SIR(t) = S(t) - I(t) \tag{2}$$

where we have neglected the thermal noise and dropped the dependence on the $i$-th user. The signal power, $S(t)$ is given by

$$S(t) = 10 \log_{10} \overline{S}_i(t) \tag{3}$$

where $\overline{S}_i(t)$ is the signal power which is received by $i$-th receiver. In the logarithimc decibel domain, signal and interference power obey a Gaussian distribution with given mean and variance, $S(t) \sim N(\mu_s, \sigma_s^2)$, $I(t) \sim N(\mu_I, \sigma_I^2)$. Thus, the $SIR(t)$ is also Gaussian, i.e. $SIR(t) \sim N(\mu_s - \mu_I, \sigma_s^2 + \sigma_I^2)$.

In order to compute the throughput at the microcell we define the successful average duration of the packet transmission from the base station to the mobile user, $ADT_{suc}$ as the time duration that $I(t) < I_{Thr}$. It means that the SIR

stays above the given threshold level for at least $\tau_M$ seconds, where $\tau_M$ is the minimum time required for a packet transmission.

The probability density function of $\tau$ as well as the probability that $\tau$ is more than a given threshold, $\tau_M$, for the zero mean and unit variance normal process of $I_n = (I - \mu_I)/\sigma_I$ as

$$f_\tau(\tau) = \frac{\lambda_I I_n^2 \tau}{4} e^{-\frac{\lambda_t t_0^2 \tau^2}{8}} \tag{4}$$

and

$$P(\tau > \tau_M) = \int_\tau^\infty f_\tau d\tau = e^{-\frac{\lambda_t I_n^2 \tau_M^2}{8}} \tag{5}$$

The average duration of a successful event can now be obtained as

$$ADT_{suc} = e^{A\tau_M^2} \left[ \tau_M e^{-A\tau_M^2} + \sqrt{\frac{\pi}{A}} Q(\sqrt{2A}\tau_M) \right] \tag{6}$$

where $A = \lambda I_n^2/8$.

The probability of success, $P_{suc}$, is defined as the fraction of the time that the base station in a microcell can successfully transmit data packets (a successful event), i.e.

$$P_{suc} = \lim_{t \to \infty} (\text{Total 'success event' time in } [0, t])/t \tag{7}$$

Thus, the probability of success given by $P_{suc} = f_{suc} \cdot ADT_{suc}$, where $f_{suc}$ is given by [12]

$$P_{suc} = \frac{\sqrt{\lambda}}{2\pi} e^{-\frac{4A}{\lambda}} \left[ \tau_M e^{-A\tau_M^2} + \sqrt{\frac{\pi}{A}} Q(\sqrt{2A}\tau_M) \right] \tag{8}$$

The average throughput, $\Theta$, at the base station in the microcell is formulated by $P(I(t) \leq I_{Thr,i}) \cdot P(\tau > \tau_M)$. Using the Eq. (5) we obtain

$$\Theta = -\frac{c}{2} e^{-A\tau_M^2} e^{\frac{\sigma_I^2 - 2k\mu_I}{2\kappa^2}} \operatorname{erf}\left( \frac{\kappa^2 \operatorname{Ln}(\frac{c}{x}) - \kappa\mu_I + \sigma_I^2}{\sqrt{2}\kappa\sigma_I} \right) \Bigg|_{R_{min}}^{R^{max}} \tag{9}$$

where $R_{min}, R_{max}$ are the minimum and maximum data rates, $\kappa = \frac{10}{\operatorname{Ln}\ 10}$ $c = \frac{W}{\Gamma} g p_{max}$, $W$ is the bandwidth, $\Gamma$ is bit-energy-to-noise-spectral density ratio required to support the given service, $g$ is the path-loss, $p_{max}$ is the maximum transmission power at the microcell base station.

We assume that succesive burst and silence periods in a packet transmission form an alternating renewal process, and their durations are exponentially distributed with means $\frac{1}{\mu}$ and $\frac{1}{\lambda}$. If the microcell base station treated as a source is ON, it generates packets at the rate of one packet per unit time. When passing through a radio link, the packet series appears to be a Markovian ON-OFF process [2]. Then, the means of the ON and the OFF periods are as follows

$$\frac{1}{\mu} = \frac{\tau_M}{h} \tag{10}$$

$$\frac{1}{\lambda} = T - \frac{1}{\mu} \tag{11}$$

where $\tau_M$ is the mean time required for a packet transmission, $h$ is the radio link speed, $T$ is the sum of the ON and OFF periods.

Now, we find the traffic and cognitive network constraints imposed on multi-dimensional QoS requirements.

We assume that a mobile station dedicates bandwidth $C$ to a real-time service, and the deadline for packets of the service is $D$. Thus, we obtain a critical packet queue length, namely

$$q_D = C \cdot D \tag{12}$$

All packets beyond this point in the queue violate their deadlines and can be dropped. Then, the probability that the packet queue length in a mobile station is greater than $Q_D$ for Markovian traffic can be estimated as

$$\phi = (q > q_D) \approx e^{-q_D \delta} \tag{13}$$

$\delta$ is a constant satisfies the following condition when $t \to \infty$, namely

$$\delta = \max\{s : A(s) \leq C\} \tag{14}$$

A(s) is the so-called effective bandwidth of data transmission between micro-cell base station and mobile station, and is given by

$$A(s) = \lim_{t \to \infty} A(s,t) \tag{15}$$

According to the theory of effective bandwidth [7] the effective bandwidth for a Markovian traffic source with two states ON and OFF is given by

$$\hat{C}_{eff} = \frac{1}{2s}\left(hs - \mu + \lambda + \sqrt{(hs - \mu + \lambda)^2 + 4\lambda\mu}\right) \tag{16}$$

where $\mu$, $\lambda$ are the transition from the ON state to the OFF state and from the OFF state to the ON state, respectively, and $h$ is the traffic rate in the ON state corresponding to the speed of the radio link where the traffic is passing.

With the additive property of the effective bandwidth, the aggregate of homogeneous traffic input to the mobile station has the effective bandwidth, namely

$$A(s) = n \cdot \hat{C}_{eff} \tag{17}$$

where $n$ is the number of transmission channels.

In Appendix A, we show that the relation between the average throughput $\Theta$ in the CR network and the parameter $\delta$ is given by

$$\delta = \frac{h^2}{(h - \frac{C}{n})(h - \frac{C \cdot \Theta}{n})} \frac{1 - \Theta}{\tau_M} \tag{18}$$

Thus, in the high speed network ($h \to \infty$) the Eq. (18) is simplified as follows

$$\delta = \frac{1 - \Theta}{\tau_M} \tag{19}$$

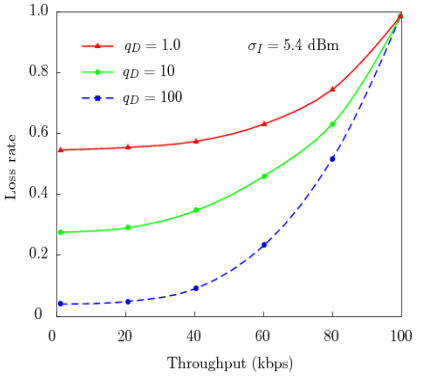



**Fig. 2.** Loss rate of the secondary user in the CR network as a function of throughput for various size of $q_D$

For the given value of $\tau_M$ the loss rate for the real-time service with deadline $D$ the traffic rate cannot be lower than the limit

$$\phi \approx e^{-q_D \cdot \delta} = e^{q_D(\frac{1-\Theta}{\tau_M})} \tag{20}$$

Thus, we obtain

$$\phi > e^{\frac{-q_D}{\tau_M}} \quad \text{for} \quad \forall\, \Theta > 0 \tag{21}$$

This inequality indicates a lower bound for the loss rate in the CR network. If $\tau_M$ is large enough, all packets will disappear ($\tau_M \to \infty, \phi \to 1$). On other hand, if $\tau_M$ is small enough ($\tau_M \to 0$, $\phi \to 0$), the packet loss rate in the CR network can be arbitrarily low for any throughput $\Theta$. So $\tau_M$ has a critical effect on the QoS behaviour of a mobile station.

## 3 Numerical Example

Let us assume a CR network with a service and a video service. We assume that this network supports the real-time services. In this sample, the bandwidth is shared by the voice and the video services by 10 and 50 Mbps, respectively. We assume that each radio link can collect up to 1 Mbps voice traffic and 5 Mbps video traffic.

Let the minimum time required for a packet transmission $\tau_M$ be equal to 2 s. The relation between loss rate $\phi$ and throughput $\theta$ for a different queue length $q_D$ in the mobile station is depicted in Fig. 2. Fig. 3 shows the relations between the loss rate and $\tau_M$ for a different throughput and a constant value of buffer size $q_D = 10$.

The impact of the $\tau_M$ on the average throughput $\Theta$ for Gaussian process with the given standard deviation $\sigma_I$ is presented in Fig. 4. The marked curves show that as the minimum required time interval of the packet transmission increases, the average throughput decreases. This means that the optimal packet lengths in the CR network are in dependency of the parameters of the fading channels.

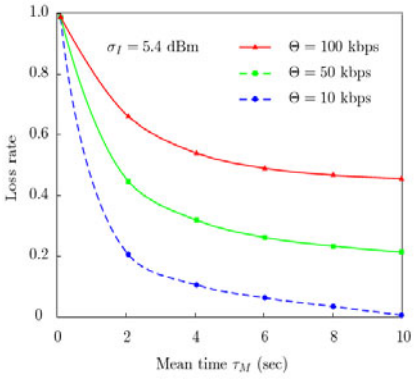



**Fig. 3.** Loss rate of the secondary user in the CR network as a function of the mean time $\tau_M$ for various value of throughput

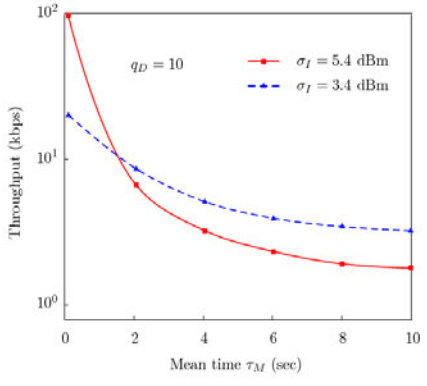



**Fig. 4.** Throughput as a function of the mean time $\tau_M$ for various values of the interference standard deviation $\sigma_I$.

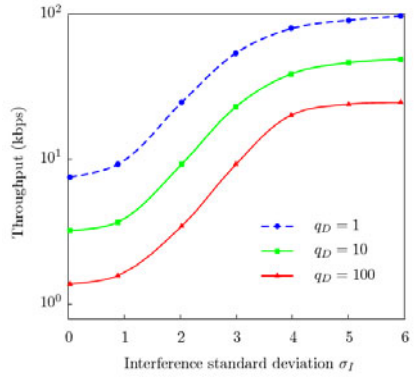



**Fig. 5.** Throughput depending on the interference standard deviation $\sigma_I$ for different values of the mean queue length

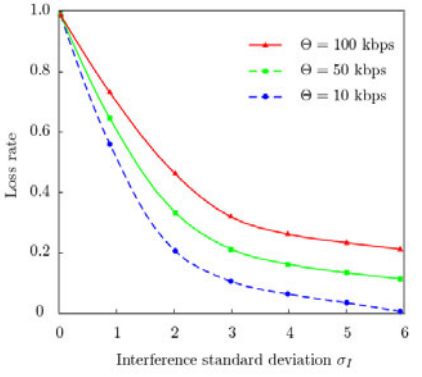



**Fig. 6.** Loss rate $\phi$ as a function of the macrocell inference standard deviation $\sigma_I$ for different values of the mean queue length

Figure 5 shows the average throughput depending on the macrocell interference standard deviation $\sigma_I$ for given mean queue length $Q_D$ at the secondary users and given mean interference level $\mu_I$. We can see that the throughput increases by increasing the interference standard deviation.

Figure 6 shows the loss rate $\phi$ as a function of macrocell interference standard deviation $\sigma_I$ for a given mean interference level $\mu_I$ and defined mean throughput $\Theta$. It can be seen that by increasing the mean throughput, the loss rate at the secondary users decreasing rapidly.

## 4 Conclusion

We have proposed a new approach for a cognitive radio organizing data with QoS prediction within CSMA-based hierarchical cell structures with the macrocell and microcell with the same frequency band. Based on the theory of effective bandwidths, we obtained a uniform formula for the throughput, the delay, and the loss rate for the Markovian traffic. Moreover, the important traffic parameters, such as the radio link speed and the minimum time required for the data packet were involved. The derived performance metrics were verified by means of the simulation experiment which incorporates the average throughput at the microcell. It allowed us to evaluate the QoS parameters at the given microcell or improve another parameters of the multidimensional traffic in the CR networks.

The proposed framework can be further used for studying the radio resource management or optimizing the QoS parameters with respect to the QoS operator policy.

## References

1. Akyildiz, I.F., Lee, W.-Y., Vuran, M.C., Mohanty, S.: NeXt Generation/Dynamic Spectrum Access/Cognitive Radio Wireless Networks: A Survey. Computer Networks Journal (Elsevier) 50, 2127–2159 (2006)
2. Anick, D., Mitra, D., Soudhi, M.M.: Stochastic Theory of Data-Handling System with Multiple Resources. The Bell System Tech. J. 61(8), 1871–1894 (1962)

3. Cao, L., Zheng, H.: Distributed Spectrum Allocation via Local Bargaining. In: Proc. IEEE SECON (September 2005)
4. Chen, T., Zhang, H., Maggin, G.M., Chlamtac, M.: CogMesh: A Cluster-Based Cognitive Radio Network. In: 2nd Int. Symp. on New Frontiers in Dynamic Spectrum Access Networks (DySPAN 2007), pp. 168–171 (2007)
5. Gorashi, S.A., Cheung, H.K., Said, F., Aghvami, A.M.: Performance of a CDMA-Based HCS Network with Hybrid Speed/Overflow Sensitive Handover Strategy. IEE Proc. Communications 150(4), 293–297 (2003)
6. Hou, Y.T., Shi, Y., Sherall, H.D.: Optimal Spectrum Sharing for Multi-Hop Software Defined Radio Networks. In: Proc. IEEE INFOCOM (May 2007)
7. Kelly, F.P.: Notes on Effective Bandwidths. In: Kelly, F.P., Zachary, S., Ziedins, I.B. (eds.) Stochastic Networks: Theory and Applications, pp. 141–168. Oxford University Press, Oxford (1996)
8. Mitola, J.: Software Radios: Wireless Architecture for the 21st Century. John Wiley and Sons Inc., Chichester (2000)
9. Peng, C., Zheng, H., Zhao, B.Y.: Utilization and Fairness in Spectrum Assignment for Opportunistic Spectrum Access. ACM/Springer MONET 11(4) (2006)
10. Shah, N., Kamakeris, T., Tureli, U., Buoldhikot, M.: Widebrand Spectrum Sensing Probe for Distributed Measurements in Cellular Band. In: ACM First Int. Workshop on Technology and Policy for Accessing Spectrum, vol. 222 (2006)
11. Su, H., Zhang, X.: Cross-Layered Based Opportunistic MAC Protocols for QoS Provisioning over Cognitive Radio Wireless Networks. IEEE Journal on Selected Areas in Communications 26(1), 118–129 (2008)
12. Vijayan, R., Holtzman, J.M.: Foundations for Level Crossing Analysis Handoff Algorithms. In: ICC 1993, vol. 2, pp. 935–939 (1993)
13. Zhao, J., Zheng, H., Yang, G.H.: Distributed Coordination in Dynamic Spectrum Allocation Networks. In: 1st Int. Symp. on New Frontiers in Dynamic Spectrum Access Networks (DYSPAN 2005), pp. 259–268 (2005)
14. Zhao, Q., Tong, L., Swami, A.: Decentralized Cognitive MAC for Dynamic Spectrum Access. In: 1st Int. Symp. on New Frontiers in Dynamic Spectrum Access Networks (DYSPAN 2005), pp. 224–232 (2005)
15. Zheng, H., Peng, C.: Collaboration and Fairness in Opportunistic Spectrum Access. In: Proc. of IEEE ICC 2005, pp. 3132–3136 (2005)

## Appendix A - The Relation between Throughput $\Theta$ and $\delta$

Here, we used the effective bandwidth in order to express the relation between the throughput in the CR network and parameter $\delta$. As can be remaindered, the parameter $\delta$ is the maximum value of $s$ satisfying the effective bandwidth condition. It is obvious that

$$A(s) = n \cdot \hat{C}_{eff} \tag{22}$$

where $n$ is the number of the transmission channels. From Eq. (14), $\delta$ is the maximum value of $s$ satisfying the condition of the effective bandwidth. From the Eqs. (15) and (16), we get

$$\frac{n}{2s}(hs - \mu - \lambda + \sqrt{(hs - \mu + \lambda)^2 + 4\lambda\mu} \leq C \tag{23}$$

where $C$ is dedicated bandwidth to real-time services. By means of rearranging terms, we obtain

$$\sqrt{(hs-\mu+\lambda)^2+4\lambda\mu} \leq \frac{2C}{n}s - hs + \mu + \lambda \tag{24}$$

After squaring both sides in the inequality, we get

$$(hs-\mu+\lambda)^2 + 4\lambda\mu \leq \left(\frac{2C}{n}s - hs + \mu + \lambda\right)^2 \tag{25}$$

A a result of developing the square on the right side and moving the terms to the left side of the inequality, we obtain

$$\left[\frac{C}{n}h - \left(\frac{C}{n}\right)\right] s^2 - \left[\frac{C}{n}(\mu+\lambda) - \lambda h\right] s \leq 0 \tag{26}$$

We know that $s > 0$. Thus, we get

$$\left[\frac{C}{n}h - \left(\frac{C}{n}\right)^2\right] s \leq \frac{C}{n}(\mu+\lambda) - \lambda h \tag{27}$$

When $h \geq \frac{C}{n}$, the left hand side of the inequality is non-negative. So

$$s \leq \frac{\mu+\lambda}{h-\frac{C}{n}} - \frac{\lambda h}{\left(\frac{C}{n}\right)h - \frac{C}{n}} \tag{28}$$

After rewriting it we obtain

$$s \leq \frac{\mu+\lambda}{h-\frac{C}{n}}\left(1 - \frac{1}{\frac{C}{n}}\frac{\frac{h}{M}}{\frac{1}{\lambda}+\frac{1}{\mu}}\right) \tag{29}$$

From the Eq. (11) we know the dependencies

$$\frac{1}{\mu} + \frac{1}{\lambda} = T = \frac{n \cdot \tau_M}{C \cdot \Theta} \tag{30}$$

$$\frac{h}{\mu} = \tau_M \tag{31}$$

As a result of substituting Eq. (30) and Eq. (31) into Eq. (29), we get

$$s \leq \frac{\mu+\lambda}{h-\frac{C}{\mu}}(1-\Theta) \tag{32}$$

From Eq. (30) and Eq. (31) we have

$$\mu + \lambda = \frac{nh^2}{(nh - C\cdot\Theta)\tau_M} \tag{33}$$

Using it into Eq. (32) we obtain

$$s \leq \frac{h^2}{(h-\frac{C}{n})(h-\frac{C\cdot\Theta}{n})} \cdot \frac{1-\Theta}{\tau_M} \tag{34}$$

Therefore

$$\delta = \frac{h^2}{(h-\frac{C}{n})(h-\frac{C\cdot\Theta}{n})} \cdot \frac{1-\Theta}{\tau_M} \tag{35}$$

This shows the relation between the throughput and parameter $\delta$.

# Hybrid Inter-Domain QoS Routing with Crankback Mechanisms

Ahmed Frikha, Samer Lahoud, and Bernard Cousin

University of Rennes 1, IRISA,
35042 Rennes Cedex, France
{ahmed.frikha,samer.lahoud,bernard.cousin}@irisa.fr

**Abstract.** In this paper we tackle the challenging problem of Quality of Service (QoS) routing in multiple domains. We propose a novel inter-domain QoS routing algorithm named HID-MCP. HID-MCP benefits from two major concepts that ensure high performance in terms of success rate and computational complexity. First, HID-MCP is a hybrid algorithm that combines the advantages of pre-computation and on-demand computation to obtain end-to-end QoS paths. Second, HID-MCP integrates crankback mechanisms for improving the path computation results in a single domain or in multiple domains. Extensive simulations confirm the efficiency of our algorithm on randomly generated topologies.

**Keywords:** QoS routing, inter-domain routing, crankback mechanisms, pre-computation, on-demand computation.

## 1 Introduction

Nowadays, diverse advanced applications are provided over IP-based networks (e.g. IPTV, video-on-demand, and VoIP). Guaranteeing the Quality of Service (QoS) to such applications is a difficult problem, especially when service delivery requires crossing heterogeneous domains under the responsibility of different operators. Inter-domain QoS routing, also known as Inter-Domain Multi-Constraint Path (ID-MCP) computation problem is one of the primary mechanisms for providing QoS. It consists of computing a path subject to multiple QoS constraints between a source and a destination node of a multi-domain network. Let us introduce some notations to formally define the ID-MCP problem. Let $G(N, E, D)$ denote a network of $D$ domains, $N$ is the set of nodes and $E$ the set of links. Let $m$ be the number of QoS constraints. In our study, we consider only additive metrics, such as cost and delay, without loss of generality [1]. An $m$-dimensional weight vector is associated with each link $e \in E$. This vector consists of $m$ non-negative QoS weights $w_i(e)$, $i = 1..m$. Let $p$ be a path in the graph $G(N, E, D)$ and $w_i(p)$ be the weight of $p$ corresponding to the metric $i$. As metrics are additive, $w_i(p)$ is given by the sum of the weights of the $i^{th}$ metric of the links of the path $p$: $w_i(p) = \sum_{e_j \in p}(w_i(e_j))$. Let $\overrightarrow{W}(p) = (w_1(p), w_2(p), ..., w_m(p))$ denote the weight vector of the path $p$.

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 450–462, 2011.

**Definition 1.** *Given a source node $s$, a destination node $d$ and a set of constraints given by the constraint vector $\vec{C}= (c_1, c_2, .., c_m)$, the Inter-Domain Multi-Constraint Path (ID-MCP) computation problem consists in finding a path $p$ which satisfies $w_i(p) \leq c_i$, $\forall i \in 1..m$. Such a path $p$ is called a feasible path.*

The ID-MCP problem is $\mathcal{NP}$-hard [2] and may have zero, one or multiple solutions (feasible paths). Computing such a path requires knowledge of the topology of each domain in the network, as well as the QoS metrics on network links. As the operators can be in competition, information about the internal topology or the available resources in the network is confidential. Hence, computing such a path using a centralized method is a hard task. Currently, the inter-domain routing protocol is BGP. This protocol cannot solve the ID-MCP problem since it does not take into account QoS constraints. Many extensions for BGP are proposed to support QoS routing [4]-[5]. However, the QoS capabilities of these propositions remain limited. Furthermore, solving the ID-MCP problem using a centralized method is a very complex problem. Therefore, the research community has recently been exploring the use of distributed architectures to solve this problem, such as the PCE (Path Computation Element) architecture [6]. Distributing the computation over domains preserves confidentiality of each domain and solves the scaling problem. To our knowledge, few works have been proposed to solve the ID-MCP problem using distributed methods. The algorithm proposed in [7] extends the exact algorithm SAMCRA [3] to an inter-domain level to solve the ID-MCP problem. The drawback of this algorithm is its high complexity. Work in [8] proposes also a promising distributed solution with crankback mechanisms for inter-domain routing. However this solution cannot take into account several QoS metrics.

In this paper, we propose a novel inter-domain QoS routing algorithm, named HID-MCP. HID-MCP is based on a hybrid computation scheme that combines path pre-computation and on-demand path computation. HID-MCP consists of two phases: An offline phase and an online phase. In the offline phase, HID-MCP pre-computes a set of QoS paths. In the online phase, HID-MCP combines the pre-computed paths to obtain an end-to-end path that fulfills the QoS constraints. Combining the pre-computed paths does not lead always to an end-to-end path. In such a case, a crankback mechanism is executed to perform on demand computations. Combining pre-computation and on-demand computation using crankback mechanisms improves the computation results and allows computational complexity to be reduced. Besides, our solution relies on a distributed architecture to overcome the limitations related to inter-domain routing. Extensive simulations confirm the efficiency of our algorithm in terms of success rate and computational complexity.

The rest of this paper is organized as follows. In Section 2, we present the concept of the HID-MCP algorithm and its operations. Simulation results are presented in detail in Section 3 and a conclusion is given in Section 4.

## 2 The HID-MCP Algorithm

In this paper, we propose a novel inter-domain QoS routing algorithm based on a hybrid computation scheme and named HID-MCP (Hybrid ID-MCP). The HID-MCP algorithm consists of two phases. In the first phase, named the offline path computation phase, the algorithm executes an intra-domain pre-computation algorithm and computes *look-ahead* information for each domain. The pre-computed intra-domain paths and the *look-ahead* information are stored in a database for later use. The second phase, named online path computation phase, is triggered upon the reception of a QoS request. In this phase, HID-MCP computes an end-to-end path that spans multiple domains and fulfills the QoS constraints. The end-to-end path computation benefits from the stored pre-computed intra-domain paths and the *look-ahead* information to speed up the computational time of the algorithm.

### 2.1 The Offline Path Computation Phase

The offline computation phase consists of computing in advance a set of intra-domain paths subject to multiple predetermined QoS constraints. It also computes *look-ahead* information at the level of each entry border node of the corresponding domain. In the following, we detail the operations involved in these two computations.

**The Path Segment Computation Procedure.** This procedure pre-computes a set of paths from each entry border node of the domain toward the other nodes of this domain as well as the entry border nodes of the neighbor domains. These paths satisfy a set of predetermined additive QoS constraints. In practice, some QoS metrics are more critical for certain applications, such as the delay for the VoIP-based applications. Therefore, our procedure pre-computes for each single QoS metric the path which minimizes the weight corresponding to this metric. For example, it pre-computes the path which minimizes the delay; this path can be useful for the VoIP-based applications.

Let $D_q$ be the considered domain, $n_1$ be a border node of $D_q$, $n_2$ be a node of $D_q$ or an entry border node of a neighbor domain, and $m$ be the number of the QoS metrics, our procedure computes $m$ shortest paths from $n_1$ to $n_2$. Each shortest path minimizes a single QoS metric. Hence, from each entry border node $n_1$ of $D_q$, this procedure computes $m$ shortest path trees. Each shortest path tree is computed using the Dijkstra algorithm and considering a single metric. Therefore, our procedure executes Dijkstra $m$ times per border node.

**Theorem 1.** *The complexity of the path segment computation procedure is in* $\mathcal{O}(B * m(N \log(N) + E))$*, where* $B$ *is the number of the entry border nodes of the domain.*

*Proof:* The complexity of this procedure depends on the number of constraints $m$. For one border node, this procedure is in $\mathcal{O}\left(m(N \log(N) + E)\right)$ corresponding to $m$ times the complexity of Dijkstra, which is $\mathcal{O}((N \log(N) + E))$. Considering the $B$ entry border nodes of the domain, the global complexity is then given by: $\mathcal{O}(B * m(N \log(N) + E))$.

**Look-Ahead Information Computation Procedure.** During the offline phase of HID-MCP, we propose the computation of *look-ahead* information in each domain. This information gives a measure of the best QoS performance that can be provided by the domain. Particularly, it allows the computation search space of a potential on-demand path computation procedure to be reduced. For instance, this information allows infeasible paths to be discarded from the search space of the procedure before exploring these paths. Therefore, *look-ahead* information reduces the computational complexity of the online phase and contributes to maintain a reasonable response time. *Look-ahead* information is inferred from the result of the pre-computation algorithm. Let $n_1$ be a border node of the domain, and $n_2$ be a node of the domain or an entry border node of a neighbor domain, and $p^*_{n_1 \mapsto n_2;i}$ denotes the pre-computed shortest path between node $n_1$ and node $n_2$ considering the metric $i$. The weight $w_i(p^*_{n_1 \mapsto n_2;i})$ is the lowest possible path weight between $n_1$ and $n_2$. Similarly, let us denote by $\overrightarrow{W}^*_{n_1 \mapsto n_2} = (w^*_1, .., w^*_m)$ the vector where $w^*_i = w_i(p^*_{n_1 \mapsto n_2;i})$. Then, $\overrightarrow{W}^*_{n_1 \mapsto n_2}$ represents the lowest weights to reach $n_2$ from $n_1$ for each single metric. We note that a path does not necessarily exist with this lowest weights for all the metrics simultaneously. However, this vector can be used in the online path computation phase to discard infeasible paths from the search space.

**Theorem 2.** *The complexity of the look-ahead information computation procedure is in* $\mathcal{O}(m * N * B)$.

*Proof: Look-ahead* information is inferred from the result of the path segment computation. At each entry border node of the domain, there are at most $m * N$ stored pre-computed paths. Hence, at the level of an entry border node $n$ the complexity of computing the $N$ vectors $\overrightarrow{W}^*_{n \mapsto n_j}$, where $n_j \in N$, is in $\mathcal{O}(m * N)$. Therefore, the complexity of computing the *look-ahead* information for all the entry border nodes of the domain is in $\mathcal{O}(m * N * B)$.

### 2.2 The Online Path Computation Phase

The online path computation consists in finding a feasible end-to-end path using the pre-computed paths and taking advantage of the *look-ahead* information. Upon the reception of a QoS request, the source and the destination domains are determined. According to the cooperation policy, the service provider computes the best domain sequence that links the source and the destination domain [6]. The path computation is triggered in the destination domain toward the source domain following the selected domain sequence. Note that, without loss of generality, we rely on backward computation according to the PCE architecture. Let $Seq = \{D_1, D_2, .., D_r\}$ denote the selected domain sequence, where $D_1$ is the destination domain and $D_r$ the source domain. Let $d$ be the destination node and $s$ be the source node. Algorithm 1 illustrates the operations performed in the online phase of HID-MCP. First, our algorithm attempts to compute an inter-domain path by combining the pre-computed paths in each domain $D_q$ in $Seq$ starting from the destination domain $D_1$: the path combination procedure

**Algorithm 1.** Online Phase of HID-MCP ($Seq$,$s$,$d$)

1: $q \leftarrow 1$; $H \leftarrow \phi$; $reject\_request \leftarrow false$;
2: **while** $(q \leq r)$ and not($reject\_request$) **do**
3: $H \leftarrow$ Path_combination_procedure($D_q, H, s, d$);
4: **if** $H \neq \phi$ **then**
5: $q \leftarrow q + 1$;
6: **else if** $intra\_domain\_crankback$ **then**
7: $H \leftarrow$ On_demand_computation($D_q, H, s, d$);
8: **if** $H \neq \phi$ **then**
9: $q \leftarrow q + 1$;
10: **else**
11: $reject\_request \leftarrow true$;
12: **end if**
13: **else**
14: $H \leftarrow \phi$; $q \leftarrow 1$;
15: **while** $(q \leq r)$ and not ($reject\_request$) **do**
16: $H \leftarrow$ On_demand_computation($D_q, H, s, d$);
17: **if** $H \neq \phi$ **then**
18: $q \leftarrow q + 1$;
19: **else**
20: $reject\_request \leftarrow true$;
21: **end if**
22: **end while**
23: **end if**
24: **end while**
25: Return $reject\_request == false$

is called (line 3). Operations performed by this procedure are detailed in section 2.2. The result of the combination procedure in each domain $D_q$ is a set of sub-paths linking the destination node to the entry border nodes of the up-stream domain $D_{q+1}$. These sub-paths are sent to domain $D_{q+1}$ to combine them with the pre-computed segments in domain $D_{q+1}$. To preserve domain confidentiality, sub-paths are communicated between domain under a novel compact structure named VSPH (Virtual Shortest Path Hierarchy[1]) [1]. This structure contains only the end nodes of the paths (the destination node and the entry border nodes of the up-stream domain) as well as the weight vector of each path. The VSPH is denoted by $H$ in algorithm 1. A virtual path $p_{d \to n}$ is represented in the VSPH by $\left[d, n, \overrightarrow{W}\left(p_{d \to n}\right)\right]$, where $d$ is the destination node, $n$ is an entry border node of the upstream domain $D_{q+1}$, and $\overrightarrow{W}\left(p_{d \to n}\right)$ is the weight vector of $p_{d \to n}$.

Combining the pre-computed paths in each domain can lead to an end-to-end path, as detailed in section 2.2. However in some cases, no feasible path is found, i.e. the returned VSPH is empty. We introduce in the following two novel

[1] The hierarchy is a structure which enables the storage of multiple paths between any two nodes [9].

approaches using crankback mechanisms in order to overcome this limitation. The first approach executes an intra-domain crankback while the second approach executes an inter-domain crankback. Both of these approaches perform an on-demand path computation. The aim of the on-demand path computation procedure is to provide better results than the pre-computed ones. Operations performed by this procedure are detailed in section 2.2.

The intra-domain crankback approach (lines 6-12) executes the on-demand path computation procedure in the current domain, i.e. where the combination has failed. Then, if a feasible path is found in the current domain, this path is sent to the up-stream domain which will resume the path combination procedure. Otherwise, if the algorithm does not find a solution in the current domain, i.e. $H = \phi$, the request is rejected.

The inter-domain crankback approach (lines 13-23) executes the on-demand path computation procedure starting from the destination node $d$. Each domain executes the on-demand path computation procedure and sends the computed VSPH to the up-stream domain. The computation stops when an end-to-end path is found or when the on-demand path computation procedure does not find a solution, i.e. the returned VSPH is empty. In the latter case, the request is rejected.

**Path Combination Procedure.** The aim of this procedure is to combine the paths in the received VSPH with the internally pre-computed one. Algorithm 2 illustrates the operations performed by the path combination procedure. First, the combination procedure selects the pre-computed paths linking nodes in the set $I$ to nodes in the set $E$, where $I$ is the ingress node set and $E$ the egress node set (lines 1-13). Then, these paths are combined with the aggregated paths received in the VSPH (line 17). Finally, feasible paths are aggregated and added to the new VSPH which will be sent to the upstream domain. Notes that at the level of the destination domain $D_1$ there is no received VSPH ($H = \phi$), the procedure selects the feasible pre-computed paths linking the destination $d$ to the entry border nodes of domain $D_2$, and aggregates them in a VSPH to be sent to domain $D_2$. Figure 1 illustrates an example of path combination with two constraints ($m = 2$) in an intermediate domain $D_q$.

**Theorem 3.** *The complexity of the pre-computed path combination procedure at the level of an intermediate domain* $D_q \in \{D_2, .., D_{r-1}\}$ *is in* $\mathcal{O}(m^q * B_{max}^2)$, *where* $B_{max}$ *denotes the maximum number of border nodes between two domains.*

*Proof:* There are at most $m^{q-1}$ paths from the destination to each entry border node of the domain $D_q$. In addition, at each entry border node, there are at most $m * B_{max}$ stored pre-computed paths to reach the upstream domain $D_{q+1}$. Hence, the complexity of combining the pre-computed paths and the received paths at the level of an entry border node is in $\mathcal{O}(m^q * B_{max})$. This operation is performed at each entry border node between the domain $D_q$ and the downstream domain $D_{q-1}$. Therefore, the global complexity of this procedure at each domain is in $\mathcal{O}(m^q * B_{max}^2)$.

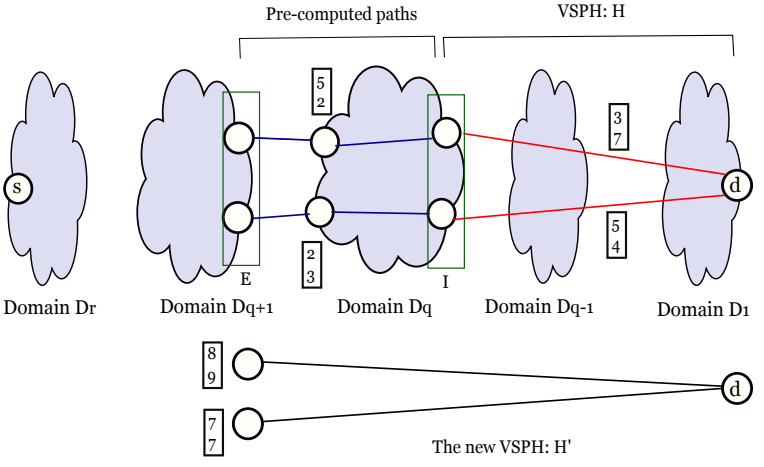



**Fig. 1.** Combining the pre-computed paths in domain $D_q$ with the VSPH

**The On-demand Path Computation Procedure.** When the pre-computed path combination procedure does not lead to a feasible path, the on-demand path computation procedure is called in the current domain or starting from the destination domain according to the two aforementioned approaches. We propose a modified version of the TAMCRA algorithm to perform the on-demand computation. Work in [10] shows that TAMCRA is an efficient tunable heuristic for the MCP problem. TAMCRA introduces a new parameter $k$ that limits the maximum number of stored paths at each intermediate node when searching for a feasible path. This parameter allows TAMCRA's performance to be tuned: the success rate can be improved by increasing $k$ at the expense of increased computational complexity. Algorithm 3 illustrates the operations performed by the on-demand path computation procedure. First of all, our proposed procedure computes a prediction for the lowest weight vector to reach domain $D_{q+1}$ through each path in the received VSPH (lines 11-19). We define for each aggregated path $\left[d, n, \overrightarrow{W}\left(p_{d\rightarrow n}\right)\right]$ in the VSPH and for each node $n_k$ in $E$, a weight vector $\overrightarrow{W^*}\left(d \mapsto n_j \mapsto n_k\right)$ that represents the sum of the weight vector of the computed segment $p_{d\mapsto n_j}$ and the lowest weight vector to reach $n_k$ from $n_j$ (line 13). Therefore, $\overrightarrow{W^*}\left(d \mapsto n_j \mapsto n_k\right) = \overrightarrow{W}\left(p_{d\mapsto n_j}\right) + \overrightarrow{W^*}_{n_j\mapsto n_k}$, where $\overrightarrow{W^*}_{n_j\mapsto n_k}$ is given by the *look-ahead* information. Note that the weight vector $\overrightarrow{W^*}\left(d \mapsto n_j \mapsto n_k\right)$ is not necessarily associated to an existing path. Next, we discard infeasible paths from the VSPH. For that, we define a new score for each path $p$ given by: $S(p_{d\mapsto n_j}) = \min_{n_k\in E}\left\{\max_{i\in 1..m}\left(\frac{w_i^*(d\mapsto n_j\mapsto n_k)}{c_i}\right)\right\}$. This score represents the lowest score to reach $d$ through $p_{d\mapsto n_j}$. A path $p$, that has a score $S(p) > 1$, is infeasible since it cannot lead to any node in $E$ while meeting the QoS constraints. Then, we classify the remaining paths in VSPH according to the score $S$. We select the $l$ shortest paths having the $l$ lowest scores, where $l$ is

**Algorithm 2.** Path combination procedure ($D_q$,$H$,$s$,$d$)

$P \leftarrow \{p/p$ pre-computed path in domain $D_q\}$;
$H' \leftarrow \phi$;
**if** $D_q == D_1$ **then**
  $I \leftarrow \{d\}$;
**else**
  $I \leftarrow \{n_j/n_j$ leaf node in $H\}$;
**end if**
**if** $D_q == D_r$ **then**
  $E \leftarrow \{s\}$;
**else**
  $E \leftarrow \{n_k/n_k$ entry border node of domain $D_{q+1}\}$;
**end if**
$Selected\_paths \leftarrow \left\{p_{n_j \rightarrow n_k}/p_{n_j \rightarrow n_k} \in P, n_j \in I, n_k \in E\right\}$;
**if** $D_q \neq D_1$ **then**
  **for** $p_{d \rightarrow n_j} \in H$ **do**
    **for** $p_{n_j \rightarrow n_k} \in Selected\_paths$ **do**
      $\vec{W}\left(p_{d \rightarrow n_k}\right) \leftarrow \vec{W}\left(p_{d \rightarrow n_j}\right) + \vec{W}\left(p_{n_j \rightarrow n_k}\right)$;
      **if** $p_{d \rightarrow n_k}$ is feasible **then**
        Add $\left[d, n_k, \vec{W}\left(p_{d \rightarrow n_k}\right)\right]$ to $H'$;
      **end if**
    **end for**
  **end for**
**else**
  **for** $p_{d \rightarrow n_k} \in Selected\_paths$ **do**
    **if** $p_{d \rightarrow n_k}$ is feasible **then**
      Add $\left[d, n_k, \vec{W}\left(p_{d \rightarrow n_k}\right)\right]$ to $H'$;
    **end if**
  **end for**
**end if**
Return $H'$;

a parameter of HID-MCP. The parameter $l$ of HID-MCP is very important to reduce the computational complexity of the on-demand computation procedure and to decrease the number of paths exchanged between domains. After that, for each selected shortest path $p_{d \mapsto n_j}$, we initialize the node $n_j$ by the corresponding weight vectors $\vec{W}\left(p_{d \mapsto n_j}\right)$ and we execute the TAMCRA algorithm starting from node $n_j$ to reach the nodes in $E$. We note that at the destination domain, *i.e.* where the computations start, there is no received VSPH. Hence, the on-demand procedure executes TAMCRA starting from the destination node. Finally, we aggregate the feasible paths computed by TAMCRA in a new VSPH.

**Theorem 4.** *The complexity of the on-demand path computation procedure at the level of an intermediate domain is in* $\mathcal{O}(l(k * N \log(k * N) + k^3 * m * E))$.

**Algorithm 3.** On demand computation procedure ($D_q$,$H$,$s$,$d$)

$temp\_paths \leftarrow \phi$; $feasible\_paths \leftarrow \phi$;
$\Psi \leftarrow \left\{ \vec{W^*} \,/\, \vec{W^*} \text{ look-ahead information in domain } D_q \right\}$;
**if** $D_q \neq D_1$ **then**
  $I \leftarrow \{n_j / n_j \text{ leaf node in } H\}$;
  **if** $D_q == D_r$ **then**
    $E \leftarrow \{s\}$;
  **else**
    $E \leftarrow \{n_k / n_k \text{ entry border node of domain } D_{q+1}\}$;
  **end if**
  $L \leftarrow \left\{ \vec{W^*}_{n_j \rightarrow n_k} \in \Psi, n_j \in I, n_k \in E \right\}$;
  **for** $\left[ d, n_j, \vec{W}\,(p_{d \rightarrow n_j}) \right] \in H$ **do**
    **for** $\vec{W^*}_{n_j \rightarrow n_k} \in L$ **do**
      $\vec{W^*}\,(d \rightarrow n_j \rightarrow n_k) \leftarrow \vec{W}\,(p_{d \rightarrow n_j}) + \vec{W^*}_{n_j \rightarrow n_k}$;
    **end for**
    $S(p_{d \rightarrow n_j}) \leftarrow \min_{n_k \in E} \left\{ \max_{i \in 1..m} \left( \frac{w_i^*(d \mapsto n_j \mapsto n_k)}{c_i} \right) \right\}$;
    **if** $S(p_{d \rightarrow n_j}) \leq 1$ **then**
      Add $[n_j, \vec{W}\,(p_{d \rightarrow n_j}), S(p_{d \rightarrow n_j})]$ to $temp\_paths$;
    **end if**
  **end for**
  **if** $temp\_paths \neq \phi$ **then**
    $Selected\_paths \leftarrow l$ shortest paths having the lowest S in $temp\_paths$
  **else**
    $H' \leftarrow \phi$;
    Return $H'$
  **end if**
  **for** $\vec{W}\,(p_{d \rightarrow n_j}) \in Selected\_paths$ **do**
    Initialize $n_j$ with the weight vector $\vec{W}\,(p_{d \rightarrow n_j})$
    Execute TAMCRA in $D_q$ starting from $n_j$ toward evry border node of domain $D_{q+1}$
    Add the obtained feasible paths to $feasible\_paths$
  **end for**
**else**
  Execute TAMCRA in $D_1$ starting from $d$
  Add the obtained feasible paths to $feasible\_paths$
**end if**
Extract $H'$ from $feasible\_paths$
Return $H'$

*Proof:* The most significant point that determines the complexity of the on-demand path computation procedure is the number of executions of the TAMCRA algorithm. Knowing that the number of initialized node is less or equal to $l$, the complexity of this operation is in $\mathcal{O}(l(k * N \log(k * N) + k^3 m * E))$, corresponding to $l$ times the complexity of TAMCRA.

## 3 Simulation and Analysis

In this section, we evaluate the performance of our novel algorithm HID-MCP by comparasion with the exact on-demand algorithm ID-MCP introduced in [7]. This algorithm has the best success rate, i.e. no other algorithm can have a success rate higher than that of ID-MCP, because ID-MCP finds a feasible path whenever a such path exists. However, the complexity of executing ID-MCP in each domain corresponds to the complexity of the SAMCRA algorithm given by: $\mathcal{O}((K_{max} * N \log(K_{max} * N) + K_{max}^3 m * E))$, where $K_{max} = \min(\exp(N-2)!, \frac{\prod_{i=1}^{m} c_i}{\max_j c_j})$ [10]. This complexity is very high comparing with that of our proposed algorithm. The simulations are performed using a network of three domains where each domain is built based on Waxman's model with 50 nodes in each domain. The probability that two nodes of the network are connected by an edge is expressed in [11]. We associate with each link two additive weights generated independently following a uniform distribution [10, 1023]. The QoS constraints are also randomly generated according to the following: Let $p_1$ and $p_2$ denote the two shortest paths which minimize the first and the second metric, respectively. Let $Z = [w_1(p_1), w_1(p_2)] \times [w_2(p_2), w_2(p_1)]$ be the constraint generation space. The problem is not $\mathcal{NP}$-Hard outside $Z$, i.e. either infeasible or trivial. As shown in figure 2, we divide this space into ten zones $Z_i$, $i = 1..10$ and we browse the space from the strictest constraint zone $Z_1$ to the loosest constraint zone $Z_{10}$. Then, we assess the performance of the algorithms according to these zones. We evaluate the algorithms based on the following performance criteria:

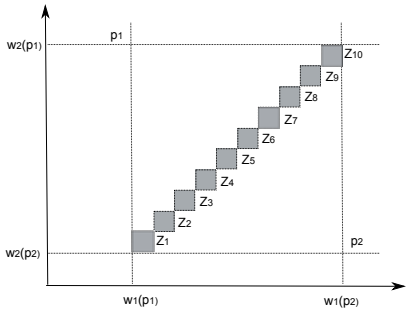



**Fig. 2.** Constraint generation zones for $m = 2$

- *GSR*: the global success rate given by the ratio of the number of the requests for which a feasible path is found and the total number of QoS requests.
- *CSR*: the efficiency of the combination procedure at a specific domain given by the ratio of the number of requests where the combination procedure is successful (e.g. leading to at least one feasible segment between the destination and the border nodes of the upstream domain) and the total number of received requests.

In the following, each figure measures the variation of one performance metric according to the constraint generation zones $Z_i$, $i \in 1..10$, with a 95% confidence interval. We focus on the success rate of the combination procedure ($CSR$) in a

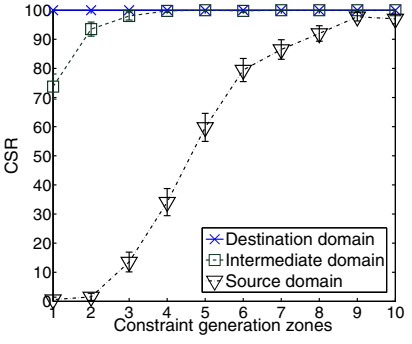



**Fig. 3.** Success rate of the combination procedure in each domain

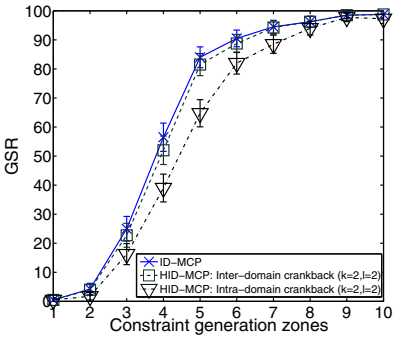



**Fig. 4.** Comparison of the global success rate of the algorithms

given domain. The complement of $CSR$ corresponds to the percentage of executions of the on-demand computation procedure. Figure 3 illustrates the variation of the $CSR$ in each domain according to strictness of the QoS constraints. In the destination domain, the combination procedure is always successful. In the intermediate domain the success rate of this procedure is high and equals 100% when constraints are not very strict. However, we note that the $CSR$ in the source domain is low, specifically when the constraints are strict. Nonetheless, this procedure performs well when the constraints are less strict. From this figure, we deduce that the probability of executing the on-demand computation is high only in the source domain when the constraints are very strict. This proves that the global empirical complexity of HID-MCP remains reasonable.

Figure 4 illustrates the variation of the global success rate ($GSR$) of HID-MCP with intra-domain crankback mechanism, HID-MCP with inter-domain crankback mechanism, and the exact algorithm ID-MCP, according to the strictness of the QoS constraints. We remark that the success rate of HID-MCP with inter-domain crankback when $l = 2$ and $k = 2$ is very close to the success rate of ID-MCP. As explained in section 2, $k$ is a parameter of TAMCRA and $l$ is the maximum number of paths selected from the VSPH. As expected, the success rate of HID-MCP with intra-domain crankback when $l = 2$ and $k = 2$ is lower than that of HID-MCP with inter-domain crankback with the same parameters, especially in the middle of the constraint generation space. In fact, when the combination procedure fails, HID-MCP with inter-domain crankback executes the on-demand computation procedure starting from the destination domain, while HID-MCP with intra-domain crankback executes it only in the current domain.

Consequently, the quality of the paths computed by the inter-domain crankback approach in each domain is better than the ones computed by the intra-domain crankback approach. Thus, the probability that a feasible path is found using HID-MCP with inter-domain crankback is higher. However, its computational complexity is high compared to HID-MCP with intra-domain crankback, but remains acceptable compared to ID-MCP. The fundamental result deduced from this figure is that HID-MCP has a slightly lower global success rate compared to the exact algorithm, while having a very low complexity.

## 4 Conclusion

In this paper, we studied the inter-domain QoS routing problem. We proposed a novel inter-domain QoS routing algorithm based on a hybrid computation scheme, named HID-MCP. We introduced two different mechanisms for improving the success rate. The first mechanism performs local improvement using an intra-domain crankback, while the second mechanism executes a global improvement using an inter-domain crankback. Extensive simulations showed that HID-MCP with both of the aforementioned approaches provides a high success rate and maintains a low complexity time. In particular, the inter-domain crankback improvement-based HID-MCP has a success rate very close to that of the exact solution, while the intra-domain crankback improvement-based HID-MCP provides lower computational complexity. This gives operators the choice to execute either of the approaches, depending on the computation policy and the priority of the request.

## References

1. Frikha, A., Lahoud, S.: Hybrid Inter-Domain QoS Routing based on Look-Ahead Information. IRISA, Tech. Rep. 1946 (2010)
2. Wang, Z., Crowcroft, J.: Quality-of-Service Routing for Supporting Multimedia Applications. IEEE Journal on Selected Areas in Communications 14(7), 1228–1234 (1996)
3. Van Mieghem, P., Kuipers, F.A.: Concepts of exact QoS routing algorithms. IEEE/ACM Transactions on Networking 12(5), 851–864 (2004)
4. Knoll, T.: BGP Extended Community Attribute for QoS Marking, draft-knoll-idr-qos-attribute-02. IETF (2009) (work in progress)
5. Griffin, D., Spencer, J., Griem, J., Boucadair, M., Morand, P., Howarth, M., Wang, N., Pavlou, G., Asgari, A., Georgatsos, P.: Interdomain routing through QoS-class planes. IEEE Commun. Mag. 45(2), 88–95 (2007)
6. Farrel, A., Vasseur, J.P., Ash, J.A.: Path Computation Element (PCE)-Based Architecture. IETF RFC 4655 (August 2006)
7. Bertrand, G., Lahoud, S., Texier, G., Molnar, M.: A Distributed Exact Solution to Compute Inter-domain Multi-constrained Paths. In: Oliver, M., Sallent, S. (eds.) EUNICE 2009. LNCS, vol. 5733, pp. 21–30. Springer, Heidelberg (2009)

8. Esmaeili, M., Xu, F., Peng, M., Ghani, N., Gumaste, A., Finochietto, J.: Enhanced Crankback Signaling for Multi-Domain IP/MPLS Networks. Computer Communications 33(18), 2215–2223 (2010)
9. Molnar, M.: Hierarchies for Constrained Partial Spanning Problems in Graphs. IRISA, Tech. Rep. 1900 (2008)
10. Van Mieghem, P., De Neve, H., Kuipers, F.A.: Hop-by-hop quality of service routing. Computer Networks 37, 407–423 (2001)
11. Calvert, K.I., Doar, M.B., Zegura, E.W.: Modelling Internet Topology. IEEE Communications Magazine 35(6), 160–163 (1997)

# Limited Values of Network Performance and Network Productivity Estimation Approach for Services with Required QoS. Service Benchmarking

Denis Andreev, Konstantin Savin, Victor Shalaginov,
Viya Zharikova, and Sergey Ilin

Central Science Research Telecommunication Institute (ZNIIS),
111141, Russia, Moscow, 1-st Proezd perova polya, 8
andreevd@zniis.ru

**Abstract.** The existing QoS research is concentrated on mathematical models using MOS/R-factor estimation. The Research Department Technopark of the Central Science Research Telecommunication Institute (ZNIIS) has applied the service degradation model on a private model network and tested it under various payloads using different network technologies. The obtained results have shown irregular distribution of MOS/R for different services (service profiles – service modifications), which radically contradicts to the requirements of various traffic network performance (NP) defined by the international standards (ITU-T). The present paper describes the limited values of network performance and network productivity estimation approach for services with required QoS. Besides, the statistics of QoS behavior (MOS/R for etalon service) at various payloads and segment of network (access and transport stratum) are given. In conclusion, the typical recommendations for network operators are given including parameters for network configuration at different segments.

**Keywords:** QoS, testing, MOS, R-factor, benchmarking, network performance, SLA, de-jitter buffer, service

## 1 Introduction

Large-scale NGN technology implementation in the first decade of the 21st century has set up new tasks in the field of quality of service. NGN concept implies a heterogeneous network construction when voice, data and video services are rendered to subscribers on the basis of a packet switching technology. The NGN heterogeneous character has essentially complicated a problem of interoperability. For NGN the term of global interoperability has been entered [1] which includes interoperability of equipment (technical means), services, classes and parameters of quality of service.

Interoperability of classes and parameters of QoS should be provided in interaction of fixed network, NGN, wireless networks of standards IEEE 802.11 and IEEE 802.16, GSM and LTE networks, as well as in further ubiquitous sensor networks [2].

The classes and parameters of QoS interoperability should conform on the NGN domain organization principle [3], based on user-network provider SLA, and on

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 463–474, 2011.

network-network SLA. Under exist network architecture the important task on QoS support consists in distribution of shares of QoS parameters on network segment (access, transport, service control) with the purpose of maintenance of the guaranteed degree of quality of service on a whole end-to-end network. This approach can further be used as initial model of resource admission control function subsystem of NGN (RACF/RACS) [10].

For determination the limited values of networks characteristics the stress testing under high level conditions have to be used. This method was called – benchmarking.

Benchmarking term was entered in [4] for equipment testing under high load (DUT – Device Under Test), in [5] benchmarking is used for system testing (SUT – System Under Test). Further, under ITU-T research the new approach of testing was proposed where two scenarios is applied — the network testing as a whole and the separate network segment testing (NUT – Network Under Test) [6]. This methodology was described in ITU-T Recommendation Q.3900 [7] and used for global interoperability testing on the model networks. The research results provided in this paper are based on the Model network of the Central Research Telecommunication Institute (ZNIIS) (Russian Federation) built in accordance with ITU-T Recommendation Q.3900 [7].

The authors investigate QoS support problems under benchmarking conditions on the model networks for VoIP service as a most difficult service in the view of real time QoS support.

The metrics of quality assessment for fixed [8] and wireless networks [9] remain MOS and R-factor.

For the purpose of rational distribution of QoS shares on the network segment the authors of this paper offer the set of service profile which is defined by the following common factors: codec, echo-cancellation, voice active detection (VAD) etc. For the given results practical application the new complex index iSA is offered.

## 2 Description of Etalon Service Model for QoS Support

The principle of definition of the maximum productivity of a network and its elements is assumed as a basis of benchmarking testing.

The throughput expressed in volumes of the passed traffic and the quantity of processed calls (SAPS) can be regarded as a criterion of network productivity.

In view of a tendency of independence of services from a network infrastructure in terms of productivity we understand the maximum quantity of services with certain properties (a service profile), that can be granted on a network segment and in a network as a whole without QoS loss.

The technology of QoS measurement has to be based on the quality of estimation of transferred information and quality of working of signaling protocols which both give for customer quality of experience (QoE).

The typical VoIP/FoIP services profiles include the following key parameters:

- Codec;
- Voice active detection (VAD);
- Echo compensation (Rec. ITU-T G.168);
- Timer values of signaling protocols;

- Call duration;
- Packetization time;
- Protocols FoIP (release of T.38).

For the purpose of definition of requirements to network segment for granting the service profile with the set QoS parameters it is necessary to define limit values of network performance parameters and network productivity parameters on various segments of a network.

The typical set of network performance parameters for IP network include: IPTD, IPDV, IPLR, IPER.

IPTD – the packet transfer delay during session (IP Packet Time Delay);
IPDV – deviation of packet transfer delay (IP Packet Delay Variation);
IPER – ratio of errors packet transfer (IP Packet Error Ratio);
IPLR – ratio of packet transfer loss (IP Packet Loss Ratio).

QoS classes and correspondence network performance parameters are shown in Rec. ITU-T Y.1541 [11]. The specified indicators and values apply just for different kinds of traffic, however, a number of services when traffic is closed to the specified types, demands more exact values when specified values of quality of these services in MOS/R-factor estimation do not correspond to user expectations and are at the lowest rate of QoE.

As productivity network parameters we use the following characteristics:

- Bandwidth;
- Access technologies;
- Routing algorithms including traffic prioritization and analyze on transport segment;
- De-jitter buffer.

As a reference of etalon service model which is used for network performance testing the following conditions are offered: quality of media information MOS is not worse than 3.5, duration of session equals 180 seconds and message signaling timer (signaling protocols) delays are not exceeding threshold values [12].

Parameters influencing QoE include the following set:

- Network effectiveness ratio;
- Service latency;
- Quality of transferred media-information.

**Network Effectiveness Ratio (NER)** [18] is defined as percent of all sessions with a positive result — call begin and release with the reason of a normal class (answer, occupied, does not answer, not accessible) from total number of sessions.

**Service latency** is defined by the time throughout the given service provided to the subscriber.

Issues of signalling messages exchange quality are not less important, than MOS estimation. For instance, when delay appears in messages of signaling protocols answer from Control Subsystem (Softswitch – SSW, IP Multimedia subsystem – IMS) the subscriber perceives deterioration of expected service as subscriber's expectation is based on the previous experience.

In accordance with ITU-T Rec. E.431 [13] the average values of post dialing delay (PDD) for different signaling protocols can lie within the limit of 3 to 5 seconds. Also this parameter could be extended on the following additional parameters:

- Lost calls;
- Initial call time;
- Release call time.

Each of these parameters, as a rule, corresponds with reception and transfer of one or several signaling messages.

In the Rec. ITU-T Q.543 [13] limit values of signaling timers SS 7 for various signaling messages are defined so as to ensure an accessible service.

ETSI has begun its work on preparation of detailed specifications with limit values of timers for different signaling protocols.

**Quality of transferred media-information**

At present, there are three methods of estimation of media-information transfer quality that are widely used on real networks. (Rec. ITU-T G.1011 [14]):

- intrusive (PESQ, POLQA);
- non-intrusive (Rec. ITU-T P.563 [15]);
- E-model.

For the given research work the E-model has been chosen as the most suitable approach for real operator network (MOS/R-factor estimation).

ETSI specification ETSI TR 102 775 V1.4.1 [16] and Appendix 7 of ITU-T Recommendation Y.1541 [11] determine correspondence of R-factor values to MOS values and expectation rate from subscriber's point of view which can be used as a QoE estimation.

**Integral Ratio of Service Availability (Quality of Media-Information Transfer)**

For a complex service quality estimation which includes estimation quality of media-information transfer and quality of signaling message handling the integral coefficient of service availability is offered:

$$iSA = \frac{Succesfull\ sessions}{Total\ sessions} \tag{1}$$

The successful sessions are defined as sessions (calls) responsible etalon service model.

This coefficient permits to estimate quality of service in technical part and could be used for calculating and planning networks segment. Also iSA could be used for balancing payload on the whole network and between some network elements in Busy Hour Call attempts (BHCA) and for network segment restriction on payload processing.

Each of the shown parameters of the etalon service model corresponds with some network performance parameters. As a result, for maintenance of the given service quality parameters it is necessary to define limit values of network performance parameters on various network segments for different telecommunication services.

Correlation etalon service model, network performance parameters and parameters of network productivity are given in Table 1.

**Table 1.** Relation of Service to Network Behaviour

| Content of service profile | Etalon service model | Network performance, limit values | Network productivity, limit values |
|---|---|---|---|
| – Codec<br>– Packetization time<br>– VAD<br>– G.168<br>– T.38 | – MOS > 3.5<br>– Call duration = 180 s<br>– Signaling timers < ETSI specifications requirements | – IPTD<br>– IPDV<br>– IPLR<br>– IPER | – Bandwidth;<br>– Access technologies;<br>– Routing algorithms including traffic prioritization and analysis on transport segment;<br>– De-jitter buffer. |

## 3 Determination of Limit Values of Network Performance Parameters

International standardization documents (recommendations or specifications), for instance, Rec. ITU-T Y.1542 [17], define various approaches for «end-to-end» QoS support on various NGN network segments.

However, network operators require the exact border values for specified network segments which provide appropriate indexes of QoS and QoE estimation.

For estimation the limit values of network performance and network productivity parameters the Model Network could be used. Model Network simulates the capabilities similar to those available in the present telecommunication networks, have a similar architecture and functionality and use the same telecommunication equipment (technical means) [7].

For the given tasks the Model Networks have to imitate additionally service realization and traffic transfer.

The unified model network was created on the Model Networks of Technopark test zone. The functionality and technical capabilities were defined after preliminary technical audit.

The Model network structure is shown in Fig. 1

The QoS/NP test specifications include the following checks:

- to determine amount of service with profile (profile N) which could be provided on scenarios «end-to-end» with the given characteristics of network productivity and in conditions of etalon service model (without QoS loss);
- to determine the limit values of network performance for different network segments (access and transport) for each service and in conditions of etalon service model (without QoS loss).

During testing the VoIP service profiles were generated with following characteristics: codec G.711 or G.729, VAD switch off, packetization time 20 s and Echo canceller is switched on.

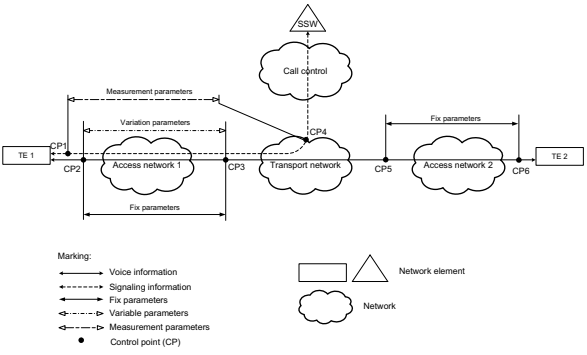



**Fig. 1.** Model Network for Network Performance, QoS and Network Productivity Testing

## 4 Key Results of Network Performance and QoS Parameters Testing on Model Network

In accordance with technical audit of existing operator networks the set of profiles has been determined (Table 2). For all determined service profiles the relevant parameters were assigned:

- Voice active detection (VAD) — off;
- Echo compensation (Rec. ITU-T G.168) — on;
- Timers values of signaling protocols — average values of timers = 300 ms;
- Call duration time — 180 s;
- Packetization time — 20 ms.

**Table 2.** Service Profiles for QoS/NP Testing on Model Network

| Service profile | Codec | De-jitter buffer, msec |
|---|---|---|
| S.Profile 1 | G.711 | 0 |
| S.Profile 2 | G.711 | 50 |
| S.Profile 3 | G.711 | 100 |
| S.Profile 4 | G.711 | 150 |
| S.Profile 5 | G.729 | 0 |
| S.Profile 6 | G.729 | 50 |
| S.Profile 7 | G.729 | 100 |
| S.Profile 8 | G.729 | 150 |

In the course of carrying out of tests the interesting results have been obtained. Fig.2 shows QoS dependence of MOS/R-factor versus session duration (sec.) for a bandwidth «end-to-end» 384 kbps with a simultaneous rendering in the channel two services with the S.Profile 1 (pointed by different color).

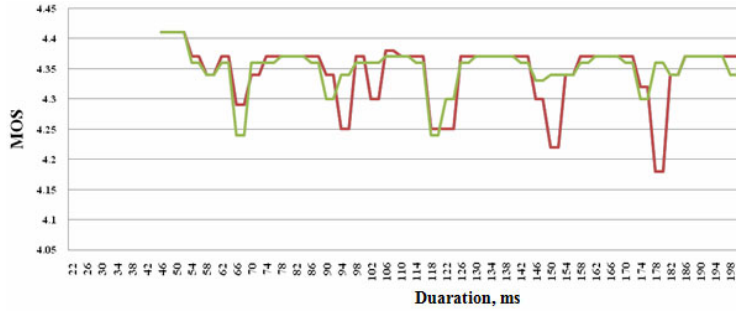



**Fig. 2.** MOS Variation Using Dependence of Set of Services with Given Profile (BW 384 kbps)

Fig.3 shows the average of MOS index for each session which appear in the given channel simultaneously.

As a result of measurements the relation of iSA and a bandwidth for different service profile has been constructed (Fig.4).

The network performance parameters for each case (MOS/IPTD/IPDV/IPLR) are also shown in Fig.4.

Carrying out the analysis of results the dependence of MOS/IPTD for each type of service has been constructed and the limiting delay factor «end-to-end» was defined. The relation of MOS and IPTD for one of service profiles is shown in Fig.5. The «end-to-end» $IPTD_{lim}$ for this case was determined as 370 ms.

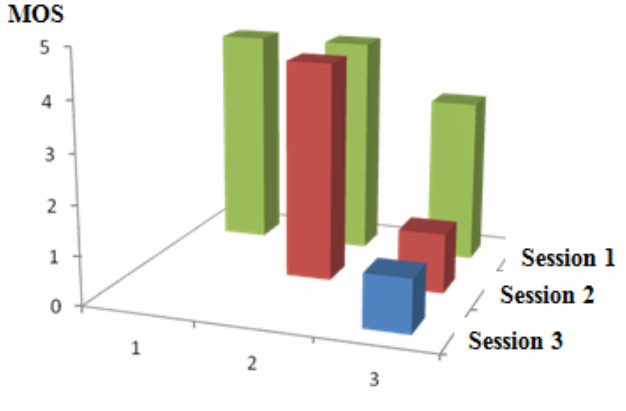



**Fig. 3.** Average MOS Index for S.Profile 1 on Fixed Bandwidth 384 kbps

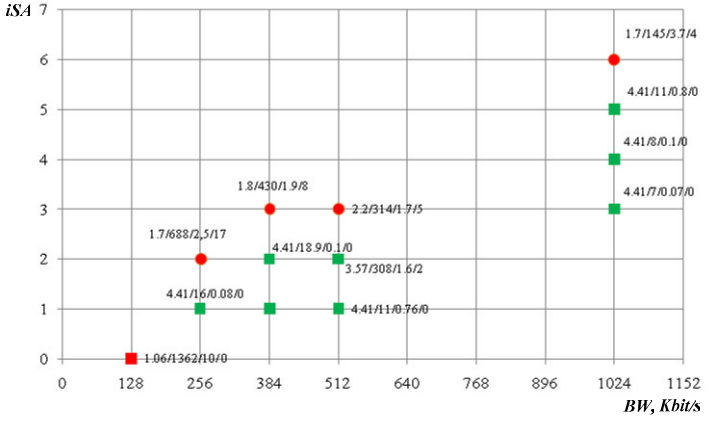



**Fig. 4.** Relation iSA and BW for s.profile 1

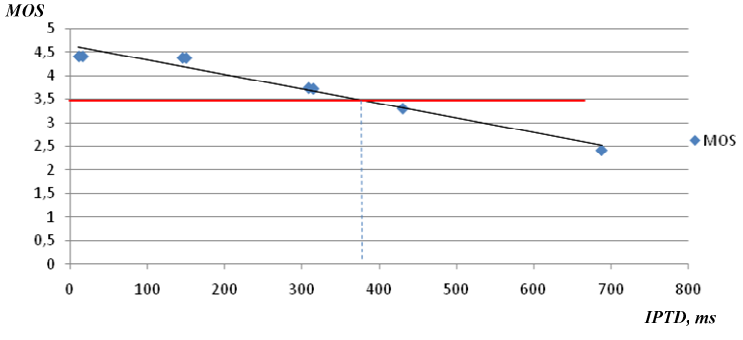



**Fig. 5.** Instance of MOS/IPTD relation for the S.Profile 1

## 5 Typical Recommendations for Network Operators for Support Requirement Values of QoS

The limited values of network performance and productivity parameters which is obtained during QoS service estimation distributed for whole network — end-to-end.

However, for distributing services as a rule is used different telecom players (service providers, infrastructure operators and etc.). Each of them to control just only part of segments which is used for service distribution.

Thereupon the major aspect consists in a formation of requirements to all network segments. For this purpose, during the given work the following approach has been applied (on an instance of calculation IPTD).

The hypothetic network model which is proposed in ITU-T Rec. Y.1541 [11] assumes to share network on five common segments. ETSI in TR 102717 [20] specify each segment etalon network model for QoS assessment.

ITU-T Rec. Y.1542 [17] determines various approaches for QoS support for different NGN segments. However, for the network segment as determined in ITU-T Rec. Y.1542, the exact values of limit network performance/productivity and correspondence of QoS and QoE values have to be found.

It should be noted that the existing latency appears globally on aggregation layers (access and switching) and the etalon network scheme under research work specifies to simplify the model (four segments): two access segments, transit and switching.

The cumulative «end-to-end» delay summarized in delays of two segments — access and transport.

Thus, for determination of limiting IPTD value on an access segment we consider that the signaling traffic is more sensitive to delays and delays deviation.

On the Model network the SIP network solution was used.

For establishment of voice session based on SIP, as a rule, 12 signaling messages are used. According to the international specifications, the time response to one message takes around 300 ms. Hence, the general time (max) under SIP establishes a session of 3.6 s.

In accordance with ITU-T Rec. E.431 [19] the maximum value of post dialing delay (PDD) is 5 s. Hence, the signaling message delays increase should not exceed 28%.

It should be taken into account that signaling protocols are more sensitive to the delays rather than the payload and neglecting delays for transfer of signaling messages on a transit network (signaling messages are transferred in separate VPN) is taken such that deviation of signaling messages delay (28%) extends only on access segment of existing network.

In case of similar technologies application on access network segment the obtained value of delay deviation is fifty-fifty.

Exact IPTD values for transport and access segment have been obtained. The instance of IPTD requirements to transport and access based on different technologies segments is given in Tables 3-4.

**Table 3.** IPTD Values on Transport Segment and Access Segment with Ethernet technology

| Service profile | IPTD | | |
|---|---|---|---|
| | End-to-end | Transport One domain | Transport Two domains |
| S.Profile 1 | 370 | 310 | 155 |
| S.Profile 2 | 100 | 40 | 20 |
| S.Profile 3 | 100 | 40 | 20 |
| S.Profile 4 | 100 | 40 | 20 |
| S.Profile 5 | 120 | 60 | 30 |

**Table 4.** IPTD Values on Transport Segment and Access Segment with xDSL Technology

| Service profile | IPTD | | |
|---|---|---|---|
| | End-to-end | Transport One domain | Transport Two domains |
| S.Profile 6 | 230 | 110 | 55 |
| S.Profile 7 | 200 | 80 | 40 |
| S.Profile 8 | 170 | 50 | 25 |

Also, the relation of a network productivity parameter (de-jitter buffer) and network performance parameter (IPTD) has been obtained. The instance of dependence is shown in Tables 5-6.

**Table 5.** Instance of Dependance of De-jitter Buffer and IPTD for Different Network Segment with Ethernet Technology on Access Stratum

| Service profile | De-jitter buffer values, ms | IPTD, ms | |
|---|---|---|---|
| | | Access segment | Transport segment |
| S.Profile 1 | 0 | 30 | 310 |
| S.Profile 2 | 50 | | 40 |
| S.Profile 3 | 100 | | 40 |
| S.Profile 4 | 150 | | 40 |
| S.Profile 5 | 0 | | 60 |

**Table 6.** Instance of Dependance of De-jitter Buffer and IPTD for Different Network Segment with xDSL Technology on Access Stratum

| Service profile | De-jitter buffer values, ms | IPTD, ms | |
|---|---|---|---|
| | | Access segment | Transport segment |
| S.Profile 6 | 0 | 60 | 110 |
| S.Profile 7 | 150 | | 80 |
| S.Profile 8 | 0 | | 50 |

## 6 Conclusions

This paper describes the results of measurement of limit values of network performance on the separate network segment for various sets of services.

Tests were conducted on the specialized Model network constructed as a result of the technical audit of exist telecommunication networks.

Under given work the proposal of a new service profile determination approach was achieved and 8 typical service profiles have been determined.

This paper provides the new method of complex assessment and base on the integral index of service availability (iSA). For the given method the approach on the

NP distribution on the network segment was proposed and the NP typical values and the network productivity for service profiles under etalon service model were determined.

The values of the obtained network productivity parameters should be used for configuring equipment on an existing operator network with the purpose of restricting situations of excess of quantity of services rendered on a network segment with the given service profile.

Also, the obtained values can be used for intrusive and non-intrusive monitoring as well as for control of SLA technical requirements.

The obtained results presented in this paper provide objective information to the effect that the service behavior depends on multitude factors which can be further used in future ITU and ETSI research. The attention should be concentrated mainly on the following mandatory aspects: rules and order of traffic/service processing on the terminal equipment (static and dynamic de-jitter buffer, packetization time depending on terminal equipment (TE) and etc.), an order of traffic routing (especially voice) in transport IP networks (the difference in traffic processing on the equipment of various vendors is experimentally defined) and etc.

In our further research work we are going to develop additional models of NGN service profiles and to propose an innovation approach for monitoring quality of services on a real network.

## References

1. Koucheryavy, A.: Networks Interoperability. In: Proceedings of 11th International Conference on Advanced Telecommunication Technologies, ICACT 2009, Phoenix Park, Korea, February 15-18 (2009)
2. Lee, J., Lee, J., Choi, J.K.: Additional Text for QoS Mapping in Y.VC-REQ (Multi-connection Requirements). NGN-GSI-C696-E, ITU-T, Geneva (August 2010)
3. Cheng, Y., Jiang, H., Zhuang, W., Niu, Z., Lin, C.: Efficient Resource Allocation for Chines 3G/4G Wireless Networks. IEEE Communication Magazine (January 2005)
4. Bradner, S., McQuaid, J.: Benchmarking Methodology for Interconnection Devices. RFC 2455 (March 1999)
5. Menasce, D.A.: Load Testing, Benchmarking, and Application Performance for the WEB. In: Computer Management Group (CMG) Conference, Reno, NV, USA (December 2002)
6. Koucheryavy, A., Andreev, D.: Draft Recommendation Q.tt1 Methods of Testing and Model Network Architecture for NGN Technical Means Testing as Applied to Public Telecommunication Networks. In: COM-11-D88-E, ITU-T, Geneva, January 23-27 (2006)
7. Recommendation ITU-T Q.3900 Methods of Testing and Model Network Architecture for NGN Technical Means Testing as Applied to Public Telecommunication Networks (September 2006)
8. Sun, L., Ifeachor, E.: New Methods for Voice Quality Evaluation for IP Networks. In: Proceedings International Teletraffic Congress (ITC-18), Berlin, Germany, 31 August-5 September (2003)
9. van den Berg, H., Bohnert, T.M., Cabral, O., Moltchanov, D., Staehle, D., Velez, F.: Rerformance Evaluation and Traffic Modelling. In: Koucheryavy, Y., Gambene, G., Staehle, D., Barcelo-Arroyo, F., Braun, T., Siris, V. (eds.) Traffic and QoS Management in Wireless Multimedia Networks. COST 290 Final Report. LNEE, vol. 31. Springer, Heidelberg (2009)

10. Andreev, D.V., Shalaginov, V.A.: Network Resource Control System for Delivering QoS of Infocommunication Services in NGN. Electrosvyaz, pp. 49–53 (2009) ISSN: 0013-5771
11. Recommendation ITU-T Y.1541 Network Performance Objectives for IP-based Services (February 2006)
12. ETSI TS 186 025-4 V2.0.4 Telecommunications and Internet Converged Services and Protocols for Advanced Networking (TISPAN); IMS/PES Performance Benchmark Part 4: Reference Load network quality parameters (March 2011)
13. Recommendation ITU-T Q.543 Digital Exchange Performance Design Objectives (March 1993)
14. Recommendation ITU-T G.1011 Reference Guide to Quality of Experience Assessment Methodologies (June 2010)
15. Recommendation ITU-T P.563 Single-ended Method for Objective Speech Quality Assessment in Narrow-Band Telephony Applications (May 2004)
16. ETSI Specification ETSI TR 102 775 V1.4.1 Speech and Multimedia Transmission Quality (STQ); Guidance on Objectives for Quality Related Parameters at VoIP Segment-Connection Points; A Support to NGN Transmission Planners (September 2010)
17. Recommendation ITU-T Y.1542 Framework for Achieving End-to-End IP performance Objectives (July 2006)
18. Recommendation ITU-T E.425 Internal Automatic Observations (March 2002)
19. Recommendation ITU-T E.431 Service Quality Assessment for Connection Set-up and Release Delays (1992)
20. ETSI Specification ETSI TR 102 717 V1.1.1 Speech and Multimedia Transmission Quality (STQ), Quality of Service Implications of NGN Architectures (October 2009)

# A Cooperative Network Monitoring Overlay

Vasco Castro, Paulo Carvalho, and Solange Rito Lima

University of Minho, Department of Informatics, 4710-057 Braga, Portugal
pmc@di.uminho.pt

**Abstract.** This paper proposes a flexible network monitoring overlay which resorts to cooperative interaction among measurement points to monitor the quality of network services. The proposed overlay model, which relies on the definition of representative measurement points, the avoidance of measurement redundancy and a simple measurement methodology as main design goals, is able to articulate intra- and inter-area measurements efficiently. The distributed nature of measurement control and data confers to the model the required autonomy, robustness and adaptiveness to accommodate network topology evolution, routing changes or nodes failure. In addition to these characteristics, the avoidance of explicit addressing and routing at the overlay level, and the low-overhead associated with the measurement process constitute a step forward for deploying large scale monitoring solutions. A JAVA prototype was also implemented to test the conceptual model design.

**Keywords:** Network Monitoring, Quality of Service, Overlay Networks.

## 1 Introduction

Monitoring of large networks raises multiple challenges regarding scalability, robustness and reliability of measurements. A monitoring model that captures the real network behaviour but that only works on small topologies is of limited applicability in today's networks. Therefore, in a monitoring system, it is necessary to find a compromise among all design goals contributing to a globally scalable and representative monitoring solution. It is known that monitoring systems where a single point is responsible for gathering and processing measurements obtained throughout the network suffer from severe scalability and robustness limitations. To address this problem, distributed solutions where monitoring data is collected and processed at each measurement point (MP) have been proposed. For instance, solutions based on active edge-to-edge measurements provide a straightforward way of measuring service quality, however, the potential interference of cross probing among boundary nodes on network behaviour needs to be carefully considered.

To reduce network overhead and improve spatial coverage, it is important to identify the most representative and critical network points in order to obtain an overall view of the network status involving only a subset of MPs. Resorting to composition of metrics between these MPs, i.e. through concatenation of partial metrics, the interference on network operation can be reduced, avoiding redundant measurements in overlapping links. The composition of metrics also allows observing trends, being more informative as a result of the underlying metric partitioning scheme.

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 475–486, 2011.

In this context, this paper proposes a collaborative network monitoring overlay which resorts to the cooperation between representative MPs strategically located in the network to compute performance and quality metrics both intra-area and end-to-end. The aim is to pursue a flexible, scalable and accurate monitoring overlay solution that simplifies and systematises the cumulative computation of metrics by involving only a subset of network nodes.

This paper is organised as follows: related work is discussed in Section 2, the proposed monitoring model and its components are described in Section 3, the model prototype is presented in Section 4, the main key points and open issues of the solution are highlighted in Section 5 and the conclusions are summarised in Section 6.

## 2 Related Work

Active monitoring carried out on an edge-to-edge basis, i.e., between network boundaries, is particularly suitable for monitoring network performance and quality of service (QoS) [1]. This approach improves scalability as only edge nodes are involved in the monitoring process, removing the complexity of monitoring tasks from the network core. The use of synthetic traffic injected in the network for measurement purposes simplifies the estimation of metrics such as delay, loss, available bandwidth [2,3,4,1]. Nevertheless, intrusive traffic may be significant in network domains involving a large number of boundary nodes. Active hop-by-hop monitoring aims to reduce the amount of synthetic traffic of active edge-to-edge measurements. Considering that in edge-to-edge probing, probes from distinct pair of edges may cross the same links, hop-to-hop monitoring strategies try to avoid repeating probes in those links. However, capturing network behaviour combining hop-by-hop measures is not an efficient and easy solution as it involves : (i) a high-degree of metrics' concatenation; (ii) monitoring agents in all network nodes; and (iii) additional traffic in the network for reporting metrics to management stations. To reduce the amount of data exchanged between management stations and MPs, several solutions have been pointed out, namely the use of flow aggregation [5], statistical summarisation [6] and network thresholds crossing alerts [7].

Inferring the traffic load of each topological link resorting only to measures of traffic entering and leaving the network, in addition to routing information, has been matter of study within the network tomography research area [8,9]. Tomography concepts continue to deserve significant attention for estimating distinct aspects of network behaviour, including QoS and fault diagnosis [10,11,12,13]. In [14], network tomography is applied to the definition of a monitoring overlay, which resorts to a subset of the topology links (overlay links) to infer packet loss ratio in all network nodes.

Taking in consideration the mentioned strategies, this study proposes a network monitoring overlay solution which resorts to representative MPs to compute performance and quality metrics both intra- and inter-area. Performing network monitoring through representative and collaborative MPs allows to define a virtual monitoring topology based on a cumulative approach for multi-metric computation with reduced overhead.

## 3 A Cooperative Monitoring Overlay

The proposed model relies on a collaborative participation of representative MPs acting as peers, each one contributing with a disjoint measure component to the evaluation of a global measure. Achieving a measurement between any two points of the network in distinct administrative entities implies the cooperation between different areas regarding evaluating a final metric. Thus, end-to-end measurements are obtained through the aggregation of metrics calculated in each of the network areas involved.

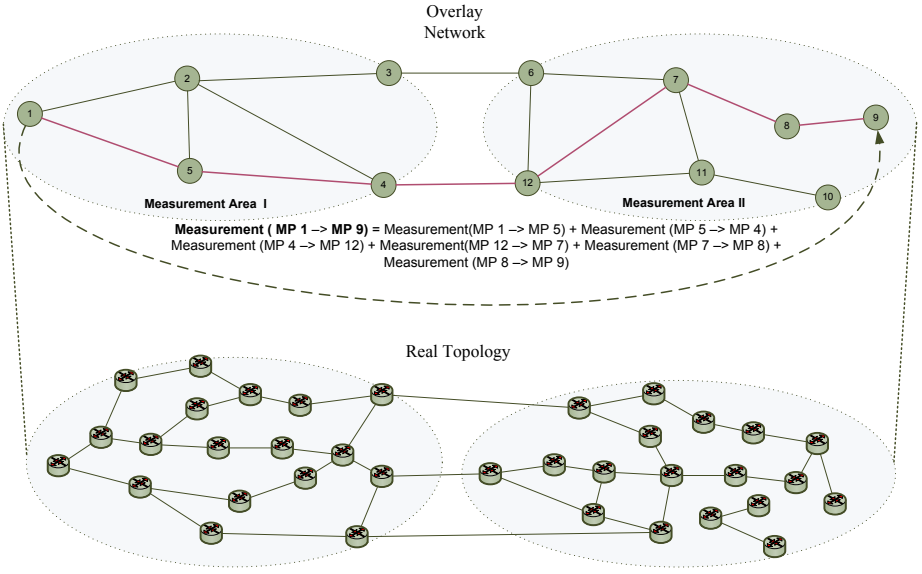



**Fig. 1.** Example of measurement between different administrative areas

Figure 1 illustrates the monitoring overlay network and the underlying physical topology. The overlay network consists of representative MPs and these are the only players taking part in the measurement process. Each MP in the overlay is expected to store the measurements to its neighbouring MPs. Thus, measurement data is distributed and stored throughout the overlay network. Based on a monitoring request, each MP in the measurement path provides the required measures for aggregation in order to calculate a set of metrics between any specified MPs. This distributed approach also has the advantage of avoiding the existence of a single point of failure. Distributing measurement data over several MPs also enables a rapid recovery of the measurement process by bringing alternative MPs in the process of rebuilding the measurement path in case of routing or network topology changes. Note that these changes do not necessarily imply a change in the overlay topology.

The proposed model allows measurements at two levels: Intra-area and Inter-area. Intra-area measurements are carried out on a regular time basis to ensure that MPs in

the same area have a clear view of network status and quality of service. An MP may, at anytime, send or exchange measurement data between itself and any other MP within its area. Thus, by retrieving data from multiple MPs in the area and using composition of metrics, it is possible to calculate the value of a metric for a given measurement path. Thus, each MP stores information on the level of quality of service to its neighbouring MPs. Inter-area measurements are performed through the composition of the metrics resulting from intra-area measurements. Conversely to intra-area operation, this type of measurement does not need to be performed continuously, but on request. This process can be triggered, for example, by an application signalling process to assess the communication path before establishing an end-to-end session crossing different network areas.

### 3.1 Model Operation

As mentioned before, measurement of multiple metrics can be carried out between any two MPs in the overlay network. This section presents a description of the phases involved in the measurement process, assuming that no optimisation tasks (e.g. caching or metrics' composition) are performed. The process, being sender-oriented, is rather simple and effective: an entity requiring measurement information issues a *Measurement Request* and, on success, will receive a *Measurement Report*.

*Measurement Request* - Initially, a monitoring entity sends a message to the initial MP indicating that it needs to obtain a set of metrics between a pair of MPs. For the topology in Figure 1, the measurement process takes place between MP1 and MP9. Upon receiving the request, MP1 sends a specific packet request for measurement purpose across the overlay network. Each MP in the overlay path will intercept this packet and attach measurement data between itself and the upstream MP, before sending it to the downstream MP. Figure 2 illustrates MP5 receiving the request and forwarding it after adding the measurement data between MP1 and MP5. This process is repeated until the destination is reached, i.e., each MP will successively attach its measurement data along the overlay, as shown in Figure 3. The final MP or the destination, upon receiving the packet measurement request, will add measurement data corresponding to the last segment of the path.

*Measurement Report*

Once the measurement in the last MP is obtained, the resulting report message is sent back to the initial MP with the collected measurement data (see Figure 4). At this point, the initial MP is able to compose the required metrics in order to obtain the end-to-end (MP-to-MP) measurement view. This operation can assume distinct cumulative functions (additive, multiplicative, max-min, etc.) depending on the nature of the metric being evaluated.

In practice, this measurement operation can be considerably simplified as area border MPs (e.g. MP 12 in Figure 1) may already have up-to-date measurements from the remaining measurement path. This allows an immediate reply from that MP to the measurement requester, reducing measuring latency significantly. This process can be further improved through proper pro-active metrics dissemination among inter-area MPs.

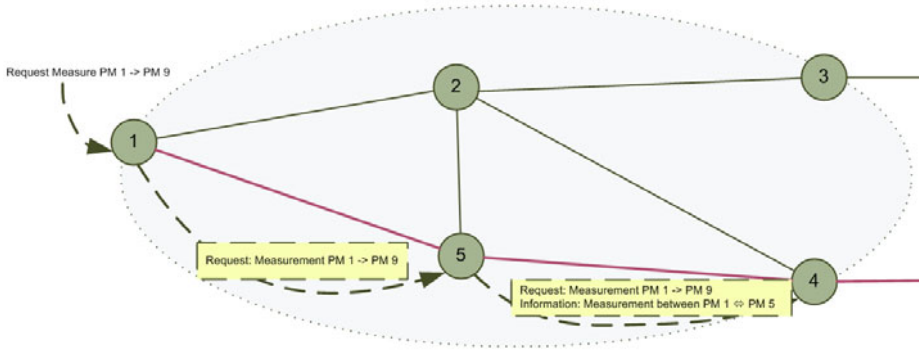



**Fig. 2.** Example of MP5 handling a measurement request

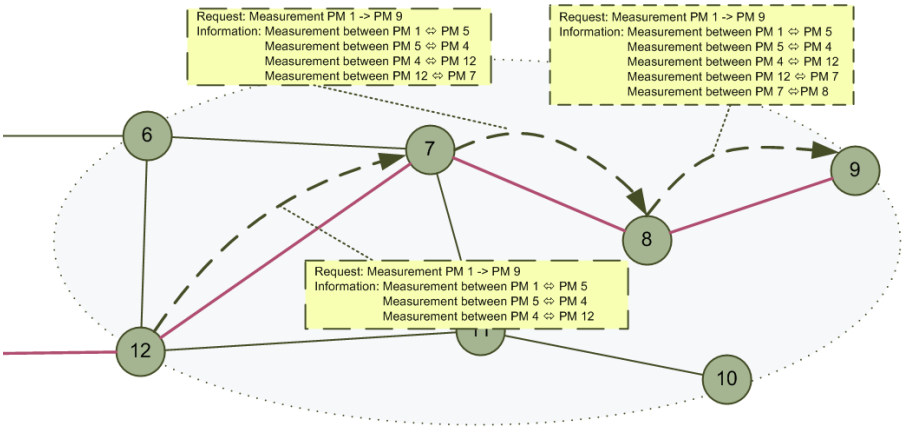



**Fig. 3.** Measurement process across multiple MPs

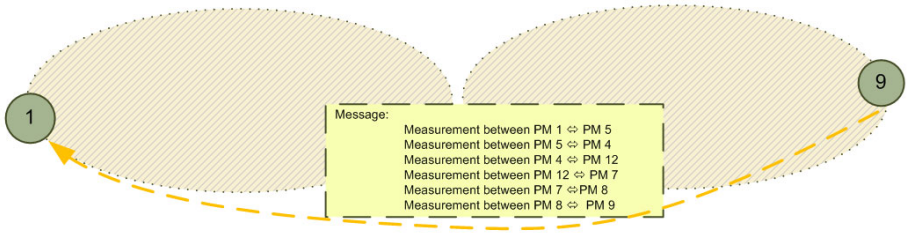



**Fig. 4.** Example of a Measurement Report

One challenge of the present model is to identify the representative MPs. Although several works target this topic [15,14,16], this aspect requires further study. These issues will be revisited in Section 5.

## 4 The Implemented Prototype

### 4.1 Model Components

To test the conceptual model design goals, a model prototype was implementated in Java and MySQL for databases support. The prototype includes four main components: (i) the “Measure Requestor”; (ii) the “Packet Interceptor”; (iii) the “Measure Processor”; and (iv) the “Measure Receiver". Figure 5 illustrates the main interactions among these components.

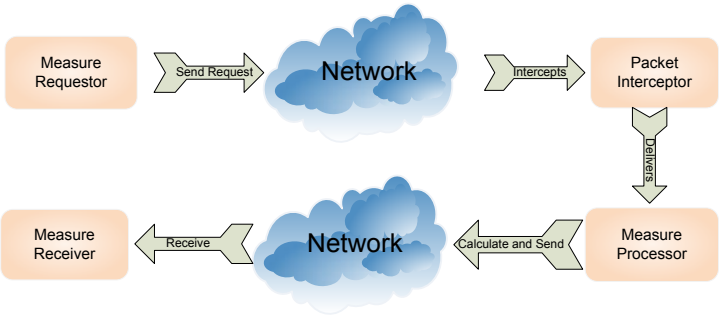



**Fig. 5.** Interaction of model components

**Measure Requestor** - This component is responsible for initiating the measurement process between two MPs. In the developed prototype, this is a command line application that receives as parameters, the source and destination MPs, and the set of metrics to measure.

**Packet Interceptor** - This component is responsible for capturing measurement packets. These packets are differentiated in the network through the use of `router alert` option within IPv4 header, avoiding packet processing at upper protocol layers. In a Linux router, this can be accomplished resorting to `iptables` and proper rules to verify the option `router alert` (it requires the extension `xtables-addons`), intercepting, in this way, the measurement packets. Captured packets are taken from kernel to user space (through `libnetfilter_queue`) for processing at MPs. The use of `router alert` option avoids the use of explicit MP addressing, allowing for a more flexible overlay topology definition.

**Measure Processor** - This component is responsible for processing and concatenating measurement data, playing a relevant role in the model prototype due to its functionality. Once a packet request is intercepted at an MP, this component detects the new request,

validates it and appends the required metrics to the measurement packet. This process involves identifying the latter upstream MP before adding its measurement contribution. Then, the component builds an IP packet setting the `router alert` option, updates the data payload accordingly and sends the packet to the downstream MP. Once the last MP is reached, the "Measure Processor" opens a TCP connection to the initial MP for sending the aggregate measurement outcome. Figure 6 depicts the modules within "Measure Processor" and how they interact to provide this component functionality.

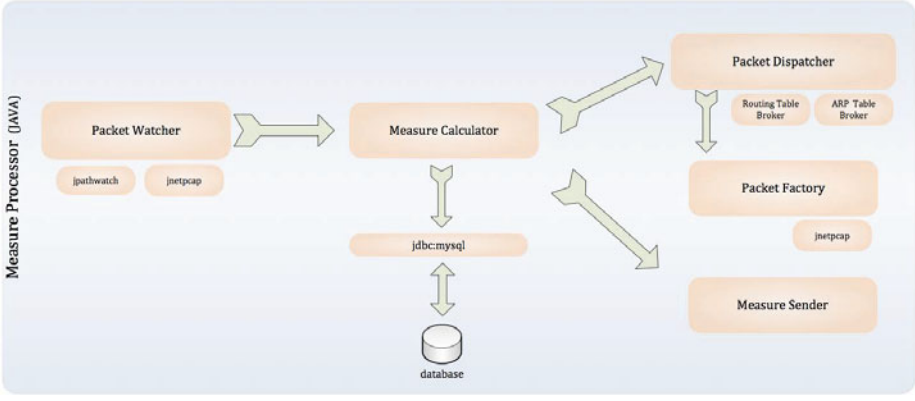



**Fig. 6.** The component Measure Processor

**Measure Receiver** - When the measurement process starts, a measurement packet request is issued and, simultaneously, the request is stored in a database, remaining in listening mode on an UDP port. Upon receiving the corresponding measurement result, this component updates the database for the corresponding request. As mentioned, the measurement reply aggregates all the metrics collected along the overlay measurement path.

### 4.2 Model Primitives

In the proposed model, the measurement primitives are structured in XML (Extensible Markup Language). Although XML structuring tends to be verbose, characteristics such as its universal format, self-descriptive nature, simplicity and extensibility, and the numerous available APIs for manipulating it, are a clear advantage.

A measurement message, following a simple format, comprises two parts or nodes: "Measure Request" ("mr") and "Measure Response" ("mrp"), as illustrated in Figure 7. The first part, generated by the starting MP, defines a header specifying the initial request for measurement. Thus, the node "mr" is composed of the following sub-nodes:

(i) hs (host source) - network address identifying the source MP;
(ii) hd (host destination) - network address identifying the destination MP;
(iii) id (identification) - key identifier associated with the request for measurement;
(iv) ms (measures) - set of metrics.

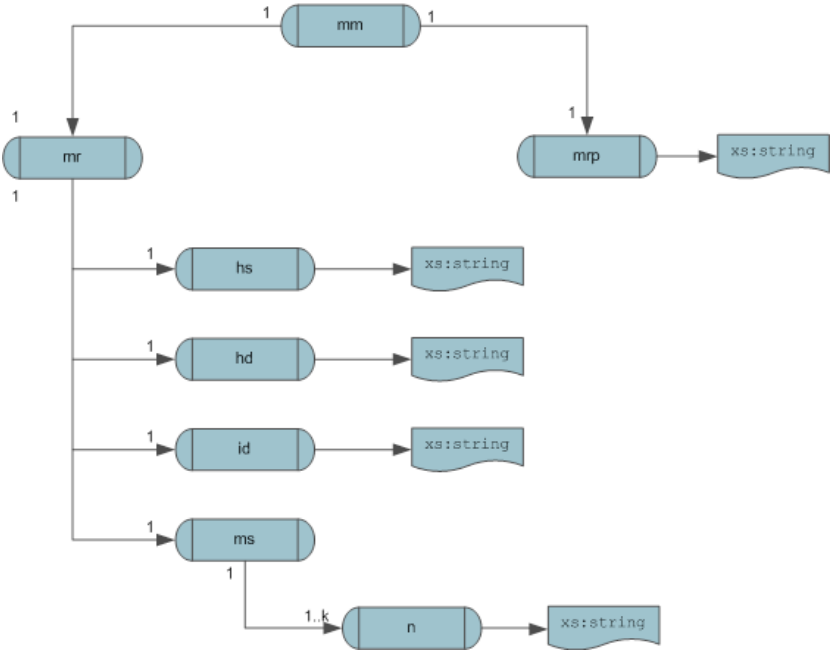



**Fig. 7.** Measurement message in XML

```
node1;node2;timestamp;metric1;metric2;...metric(k)|
node2;node3; timestamp;metric1;metric2;...metric(k)|
...|...|node(n-1);node(n);timestamp;metric1;metric2;...metric(k)
```

**Fig. 8.** Structure of node "mrp"

```
<?xml version="1.0" encoding="UTF-8"?>
<mm xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" >
    <mr>
        <hs>192.168.99.100</hs>
        <hd>192.168.117.101</hd>
        <id>25892e17-80f6-415f-9c65-7395632f0223</id>
        <ms>
            <n>loss</n>
            <n>delay</n>
        </ms>
    </mr>
    <mrp>192.168.99.100;192.168.200.1;20101012101132312;0;6</mrp>
</mm>
```

**Fig. 9.** Example of a Measurement Request between neighbouring MPs

The second part of the message, consisting of node "mrp", allows MPs in the measurement path for appending measurement data after intersecting the measurement request. Each MP provides information regarding the upstream MP, the current MP, a timestamp, and the values for the metrics defined in node "ms". The structure of node "mpr" is as shown in Figure 8.

For the sake of clarity, a simple example of an XML measurement request for packet loss and delay between MPs 192.168.99.100 and 192.168.117.101 is provided in Figure 9.

### 4.3 Testing the Prototype

As proof-of-concept of the present model, a virtualised network topology (using VMware) was considered for testing the proposed solution. Figure 10 illustrates a simple network monitoring overlay including two distinct monitoring areas and three representative MPs (MP1, MP2 and MP3). As expected only these nodes detect measurement requests and act accordingly. The virtual machines 2 and 4 run IMUNES (Integrated Network Topology Emulator / Simulator) to emulate common IPv4 backbones, running OSPF as routing protocol. This virtualised testbed allowed to carry out preliminary tests to validate the full life-cycle of an application measurement request, from its occurrence to the final response, reporting the corresponding measurement data in XML.

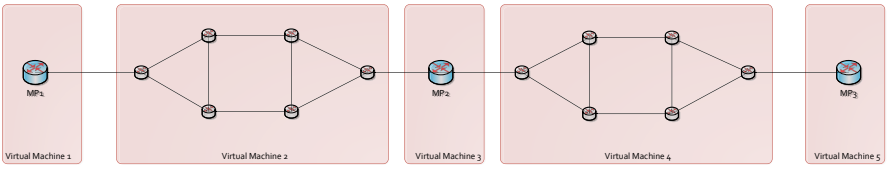



**Fig. 10.** Virtualised network for test purposes

## 5 Model Key Points and Open Issues

This section highlights the proposed model key points regarding its design and functionality and discusses open issues that may contribute positively to ongoing and future developments.

### 5.1 Key Points

The present model proposal for a cooperative network monitoring overlay, taking advantage of decentralising the control and data plane, exhibits several key properties, namely:

**Autonomy** - Each MP is responsible for maintaining its own measurements, providing them on request. Therefore, its location does not need to be pre-determined, conferring a high-degree of decentralisation to the model. The decentralisation inherent to the proposed model allows for a high-degree of autonomy as all MPs only rely on themselves upon receiving a measurement request. All information required to satisfy a measurement request is contained in each of the corresponding MPs. The autonomy degree can be improved, if each MP is aware of representative MPs in the same measurement area. This would allow to take more advantage of metrics' composition, providing also a better response in case of MP failure, for instance, through auto-configuration.

**Robustness to failure** - As mentioned above, measurement data are not centralised on a single network point being disseminated throughout MPs in the overlay, thereby ensuring that if an MP fails: (i) it does not represent the loss of all measurement information, only monitoring between that MP and its neighbouring MPs is affected; (ii) there is no need for reconfiguring the overlay network as the inclusion or exclusion of MPs is transparent to the network entities that wish to obtain an MP-to-MP (or end-to-end) measurement.

**Adaptability** - Topology changes do not require the reconfiguration of the entire overlay network, or intervention in all MPs. In fact, upon a topology change, the only need is to reconfigure neighbouring adjacencies so that the existing MPs take into account the new MPs.

**Scalability** - Attending to the nature of the model, expanding the overlay topology does not imply a direct increase in monitoring traffic. Topology growth only leads to large payloads of measurement request packets, as consequence of an eventual increase in the number of MPs, i.e. for a monitoring request traversing a longer measurement path.

**Low overhead** - The solution resorts to special-purpose probing packets requiring low processing from the network equipment, therefore, the interference of monitoring with the normal network operation is minimised. The overhead of reporting measurements to a central management or monitoring entity is also avoided, as measurements occur on demand. For large networks, fragmentation of measurement requests may however occur, as discussed in the following section.

**End-to-end capability** - The implemented prototype demonstrated that it is possible to build up an end-to-end or any other MP-to-MP combination based on local measurements.

### 5.2 Open Issues

**Location of representative MPs** - In the proposed model, as in real network operation, there is clear added-value for having MPs on (or near to) area border routers. This results from their strategic location both from technical and administrative perspectives. However, the selection of representative MPs inside a measurement area requires a deep analysis of aspects such as the centrality of MPs, (overlapping) routing paths and aggregate traffic behaviour in order to devise a suitable set of representative MPs. The challenge lays on finding the minimal set of MPs able to provide the most representative

and accurate monitoring view of the area. An equivalent study can also be carried out in the inter-area context.

**Metrics composition and dissemination** - The process of metrics' dissemination and composition deserves further development. In particular, combining a pro-active approach of disseminating metrics inside a measurement area with the possibility of avoiding overlapping measurement paths, the measurement latency and overhead may be considerably reduced.

**Fragmentation** - IP fragmentation of a measurement request may occur when the number of MPs in a measurement path increases. This problem can be avoided if fragmentation is handled within the measurement layer. This can be easily achieved fragmenting a measurement request, e.g. per metric under evaluation, if required. Alternatively, the use of measurement payload compression may also remove the need for fragmentation.

## 6 Conclusions

This paper has presented innovative research work regarding the definition of a network monitoring overlay which resorts to a cooperative interaction among representative MPs to monitor the quality of network services. In the proposed model, measurement overhead and redundancy are reduced through the composition of metrics from non-overlapping measurement paths, both intra- and inter-area. This aspect along with the ability of accommodating network topology and routing changes aim to contribute to a scalable and flexible end-to-end monitoring solution. A JAVA prototype has been implemented to test the conceptual design goals of the model. Future work will be focused on tackling the open issues identified above and performing large scale monitoring tests.

## References

1. Habib, A., Khan, M., Bhargava, B.: Edge-to-edge measurement-based distributed network monitoring. Computer Networks 44, 211–233 (2004)
2. Blefari-Melazzi, N., Femminella, M.: Measuring the edge-to-edge available bandwidth in a DiffServ domain. Int. J. Netw. Manag. 18, 409–426 (2008)
3. Duffield, N., Lo Presti, F., Paxson, V., Towsley, D.: Inferring link loss using striped unicast probes. In: Proceedings Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM 2001, vol. 2, pp. 915–923. IEEE, Los Alamitos (2001)
4. Jain, M., Dovrolis, C.: End-to-end available bandwidth: measurement methodology, dynamics, and relation with TCP throughput. IEEE/ACM Trans. Netw. 11, 537–549 (2003)
5. Lin, Y.-J., Chan, M.C.: A scalable monitoring approach based on aggregation and refinement. IEEE JSAC 20 (2002)
6. Asgari, A(H.), Egan, R., Trimintzios, P., Pavlou, G.: Scalable monitoring support for resource management and service assurance. IEEE Network 18(6), 6–18 (2004)
7. Wuhib, F., Stadler, R., Clemm, A.: Decentralized service-level monitoring using network threshold crossing alerts. IEEE Communications Magazine 44(10), 70–76 (2006)
8. Vardi, Y.: Network Tomography: Estimating Source-Destination Traffic Intensities from Link Data. Journal of the American Statistical Association 91(433), 365–377 (1996)

9. Medina, A., Taft, N., Salamatian, K., Bhattacharyya, S., Diot, C.: Traffic Matrix Estimation: Existing Techniques and New Directions. In: ACM SIGCOMM (2002)
10. Gu, Y., Jiang, G., Singh, V., Zhang, Y.: Optimal probing for unicast network delay tomography. In: INFOCOM 2010, pp. 1244–1252. IEEE Press, Piscataway (2010)
11. Burch, H., Chase, C.: Monitoring link delays with one measurement host. SIGMETRICS Perform. Eval. Rev. 33, 10–17 (2005)
12. Arya, V., Duffield, N., Veitch, D.: Temporal Delay Tomography. In: INFOCOM, pp. 276–280 (2008)
13. Huang, Y., Feamster, N., Teixeira, R.: Practical issues with using network tomography for fault diagnosis. SIGCOMM Comput. Commun. Rev. 38, 53–58 (2008)
14. Chen, Y., Bindel, D., Katz, Y.H.: Tomography-based Overlay Network Monitoring. In: ACM SIGCOMM Internet Measurement Conference (IMC), pp. 216–231. ACM Press, New York (2003)
15. Ratnasamy, S., Handley, M., Karp, R.M., Shenker, S.: Topologically-Aware Overlay Construction and Server Selection. In: INFOCOM (2002)
16. Ni, J., 0002, H.X., Tatikonda, S., Yang, Y.R.: Network Routing Topology Inference from End-to-End Measurements. In: INFOCOM, pp. 36–40 (2008)

# Decision of Transmit Relays and Transmit Power of Double Opportunistic Transmit Cooperative Relaying System in Rayleigh Fading Channels

Nam-Soo Kim[1] and Ye Hoon Lee[2,*]

[1] Department of Computer and Communication Engineering, Cheongju University, Cheongju, 360-764, South Korea
nskim@cju.ac.kr
[2] Department of Electronic and Information Engineering, Seoul National University of Science and Technology, Seoul 139-743, South Korea
y.lee@snut.ac.kr

**Abstract.** Recently, cooperative diversity has been introduced to wireless ad-hoc networks for power saving and for diversity gain over fading channel. The space diversity gain is obtained from the multiple copies of signals from the several cooperating relays. Consequently this diversity gain saves the power of the networks. However, a receiver generally has fixed number of branches at a destination. Therefore the excess number of transmit relays does not contribute to the performance improvement, and only waste the network power and radio resources. Therefore, we propose a method which is adjusting the average number of transmit relays, and controlling the transmit power of the relays locally to satisfy the required performance in double opportunistic transmit (DOT) relaying system. Numerical analysis shows that average number of transmit relays decreases with the increase of threshold and vice versa. And we noticed that the transmit power of relays for the required system performance can be controlled locally.

**Keywords:** Cooperative diversity, Relay, Rayleigh fading.

## 1 Introduction

Ad-hoc networks have focused on key technology for next generation wireless systems. However, the power consumption of wireless ad-hoc networks is critical to maintain network lifetime and communication reliability [1]. Recently, cooperative diversity has been applied to wireless ad-hoc networks to reduce power consumption and to improve system performance by mitigating the fading effects of wireless channels [2]-[5].

Conventional opportunistic transmit (COT) relaying system, when the received signal-to-noise ratio (SNR) exceeds the threshold, then the relay transmits [6]-[7]. The COT system, however, only considers the SNR of the source-relay(S-R) path regardless of the SNR of the relay- destination(R-D) path as a transmit condition. For

* Corresponding author.

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 487–495, 2011.

that reason, it can't guarantee that all the transmitted signals from the opportunistic relays satisfy the target threshold of a destination. When the received SNR at a destination is less than the threshold, the outage is occurred. It means that the useless signal is transmitted. To avoid this situation, double opportunistic transmit (DOT) relaying system which include the SNRs of the destination-relay(D-R) path as well as that of the S-R path as a transmit condition is proposed [8], [9].

Generally, there are two methods for satisfying the required system performance in a cooperative relay system with spatial diversity: (a) when the receiving signal strength is equal at each diversity branch, adjust the number of the diversity branches, (b) when the number of the diversity branches is limited, control the power of a transmitter [10], [11].

Cooperative relaying system utilizes spatial diversity. [8] and [9] assume that each node has equal transmit power. In this case, we have to adjust the number of the diversity branches to guarantee the required performance at a destination. However, it is generally accepted that the utilization of the communication resources (i.e. time slots of time division multiplexing, frequencies of frequency division multiplexing, codes of code division multiplexing) is not efficient by changing the number of the diversity branches frequently [7]. Moreover when the number of branches (i.e. fingers in a Rake receiver) in a receiver is limited, the excess number of signals is useless and wastes the power of a system.

Therefore in this paper, we consider the case that the number of branches is fixed at a destination. We propose the adjusting method that the average number the transmit relays equals to the number of the branches of a destination. Also we propose the control method of the transmit power of a relay to satisfy the required system performance. To adjust the transmit power of a relay, COT relaying system adapts a separate central controller that monitors the received signal power of all relays. And based on this monitored results, it gives the power control information to each relay. Unfortunately this controller increase control traffics and imposes system complexity. Hence this paper propose the control method of the average number and the transmit power of the transmit relays locally without a central controller in DOT relaying system.

This paper is organized as follows. Section II describes the model of the proposed DOT relaying system. Decision of the transmit relays, control of the average number of the transmit relays, outage probability, and power control of the transmit relay are described in Section III. Section IV consider some numerical examples and review the results. Finally, in Section V, the summary and conclusions of this paper are presented.

## 2 System Model

Fig. 1 shows the DOT system model [8], [9]. $S$, $R$, and $D$ denote the source, relay and destination, respectively. $R_k$ ($k = 1, 2, 3, ..., K$) means $k$ th relay of the system. DOT system need three time slots to convey the information from S to D: 1st time slot; S transmits the information to R, 2nd time slot; D transmits a pilot tone to R, 3rd time slot; when both of the received SNRs from S and D exceed respective threshold, the relays transmit [8], [9].

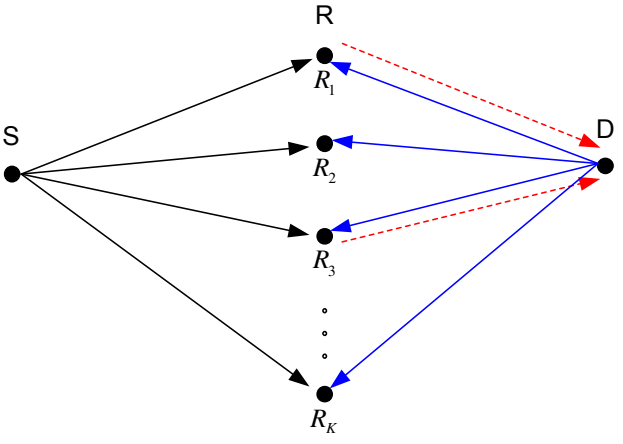



**Fig. 1.** DOT cooperative relaying system model

In Fig. 1, the solid arrows show a source of S-R path and a destination of D-R path transmit to relays. The slotted arrows denote the relays which satisfy the transmit conditions transmit to the destination. We assume Decode-and-forward (DF) relays which transmit the decoded information to D. And the multipath channels are assumed independent identically distributed (i.i.d) Rayleigh fading channel. Each received signal from $R_k$ is independently faded and combined at D for the diversity gain [10].

## 3 Power Control of the Proposed Transmit Relay

### 3.1 Decision of the Transmit Relay

If we assume the receiving branches in a destination are fixed, the number of transmit relays must be limited. In DOT system, the relays of which both of the received SNRs from S and D exceed respective threshold transmit to the destination. The transmit set can be written by

$$C = \{\gamma_{Sk} > \Gamma_{SR},\ \gamma_{Dk} > \Gamma_{DR},\ k = 1,\ 2,...,\ K\} \tag{1}$$

where $\gamma_{Sk}$ and $\gamma_{Dk}$ denotes SNR of $S - R_k$ path and that of $D - R_k$ path, respectively. If the channel is reciprocal and the transmit power is equal, the SNR of $D - R_k$ path is identical to that of $R_k - D$ path (i.e. $\gamma_{Dk} = \gamma_{kD}$ ). $\Gamma_{SR}$ and $\Gamma_{DR}$ is the threshold of S-R path and D-R path, respectively. We assume $\Gamma_{SR}$ and $\Gamma_{DR}$ is identical at each relay $R_k (k = 1,\ 2,\ ...,\ K)$, respectively. $|C|$ is the size of the transmit relay set. $\Pr(|C| = i)$ is the probability that $i$ relays transmit among $K$ relays, and can be written by

$$\Pr(|C| = i) = \binom{K}{i} \left[\Pr\left(\gamma_{Si} > \Gamma_{SR},\ \gamma_{Di} > \Gamma_{DR}\right)\right]^i \times \left[1 - \Pr\left(\gamma_{Si} > \Gamma_{SR},\ \gamma_{Di} > \Gamma_{DR}\right)\right]^{K-i} . \tag{2}$$

where $\gamma_{Si}$ and $\gamma_{Di}$ is the SNR of $S-R_i$ and that of $D-R_i$. From (2) transmit $i$ relays, that the SNR exceed the thresholds, among $K$ relays are equivalent that selection of the strongest $i$ paths (highest SNR) among $K$ available ones. This is the concept of the generalized selection combining (GSC). Especially GSC has two steps to combine the received signal: (a) Selection - select $i$ paths (highest SNR) among $K$ available ones, (b) Combine - selected signals are combined [11]-[13]. In DOT system, the selection step is performed at R and the combining step is executed at D.

The transmit probability of a DF relay in i.i.d (independent and identically distributed) Rayleigh fading channel is [14]

$$\begin{aligned}\Pr\left(\gamma_{Si}>\Gamma_{SR},\ \gamma_{Di}>\Gamma_{DR}\right) &= \Pr\left(\gamma_{Si}>\Gamma_{SR}\right)\Pr\left(\gamma_{Di}>\Gamma_{DR}\right)\\ &= \exp\left[-\left(\frac{\Gamma_{SR}}{\overline{\gamma}_{Si}}+\frac{\Gamma_{DR}}{\overline{\gamma}_{Di}}\right)\right] \quad .\end{aligned} \tag{3}$$

where $\overline{\gamma}_{si}$ and $\overline{\gamma}_{Di}$ denotes the average SNR of $S-R_i$ path and that of $D-R_i$ path, respectively. We assume the average SNR of each relay is identical, and denote $\overline{\gamma}_{Si}=\overline{\gamma}_{SR}$ and $\overline{\gamma}_{Di}=\overline{\gamma}_{DR}$ for all $i$. Notice that COT system considers the first term in (3), and consequently the transmit probability becomes $\mathbf{Pr}(\gamma_{Si}>\Gamma_{SR})$ - the transmit condition of a relay in COT system is that whether the SNR of S-R path is greater than the threshold $\Gamma_{SR}$ or not [6], [7].

### 3.2 Average Number of Transmit Relays

The average number of transmit relays which satisfy the SNRs exceed the thresholds (i.e. $\Gamma_{SR}$ and $\Gamma_{RD}$) can be written by [14]

$$\begin{aligned}&M_{DOT}\\ &=\sum_{i=0}^{K} i\binom{K}{i}\left[\Pr\left(\gamma_{Si}>\Gamma_{SR},\ \gamma_{Di}>\Gamma_{DR}\right)\right]^{i}\times\\ &\quad\left[1-\Pr\left(\gamma_{Si}>\Gamma_{SR},\ \gamma_{Di}>\Gamma_{DR}\right)\right]^{K-i} \quad .\\ &=\sum_{i=0}^{K} i\binom{K}{i}\left[\exp\left\{-\left(\frac{\Gamma_{SR}}{\overline{\gamma}_{SR}}+\frac{\Gamma_{DR}}{\overline{\gamma}_{DR}}\right)\right\}\right]^{i}\left[1-\exp\left\{-\left(\frac{\Gamma_{SR}}{\overline{\gamma}_{SR}}+\frac{\Gamma_{DR}}{\overline{\gamma}_{DR}}\right)\right\}\right]^{K-i}\\ &=K\exp\left\{-\left(\frac{\Gamma_{SR}}{\overline{\gamma}_{SR}}+\frac{\Gamma_{DR}}{\overline{\gamma}_{DR}}\right)\right\}\end{aligned} \tag{4}$$

It is noticed that the average number of transmit relays are function of two thresholds, $\Gamma_{SR}$ and $\Gamma_{DR}$.

When the number of branches are fixed at D, the threshold $\Gamma_{DR}$ to control the average number of transmit relays can be obtained from (4)

$$\Gamma_{DR}=-\overline{\gamma}_{DR}\left[\frac{\Gamma_{SR}}{\overline{\gamma}_{SR}}+\ln\left(\frac{M_{DOT}}{K}\right)\right] \quad . \tag{5}$$

The average SNR of S-R path $\overline{\gamma}_{SR}$ and that of D-R path $\overline{\gamma}_{DR}$ can be obtained from measurement at R, $K$ and $M_{DOT}$ is given parameters. Therefore the threshold $\Gamma_{DR}$ can be controlled locally at R, consequently the average number of transmit relays can be limited form (5) without the central system controller. Usually a central controller which monitors the received SNR of all relays is adapted for power control of relays. In this case, the control traffic and the system complex are increased.

### 3.3 Outage Probability of DOT System

The end-to-end probability of a DOT system can be written by

$$P_{out} = \sum_{i=0}^{K} \Pr\left(\gamma_c < \Gamma_{RD} \middle\| C \middle| = i\right) \Pr(|C| = i) \tag{6}$$

where $\gamma_c$ and $\Gamma_{RD}$ are the combined SNR at D and the threshold of R-D path, respectively.

Since $i$ paths (highest SNR) among $K$ available ones are selected, the conditional probability of GSC combined signal in (6) can be given by [11]

$$\begin{aligned}\Pr\left(\gamma_c < \Gamma_{RD} \middle\| C \middle| = i\right) = &\binom{K}{i}\{1-\exp(-\Gamma_{RD}/\overline{\gamma}_{RD})\times \\ &\sum_{l=0}^{i-1}\frac{(\Gamma_{RD}/\overline{\gamma}_{RD})^l}{l!} + \sum_{l=1}^{K-i}(-1)^{i+l-1}\binom{K-i}{l}\left(\frac{i}{l}\right)^{i-1}\times \\ &\left[\frac{1-e^{-(1+l/i)(\Gamma_{RD}/\overline{\gamma}_{RD})}}{1+l/i} - \sum_{m=0}^{i-2}(-\frac{l}{i})^m\left(1-e^{-\Gamma_{RD}/\overline{\gamma}_{RD}}\sum_{k=0}^{m}\frac{(\Gamma_{RD}/\overline{\gamma}_{RD})^k}{k!}\right)\right]\}\end{aligned} \tag{7}$$

where $\Gamma_{RD}$ and $\overline{\gamma}_{RD}$ are the threshold at D and the average SNR of R-D path, respectively. Consequently, substituting (2) and (7) into (6), we get the end-to-end outage probability.

### 3.4 Power Control of a Transmit Relay

When the required outage probability is given, the satisfying $(\overline{\gamma}_{RD}/\Gamma_{RD})_{req}$ can be obtained from (6). And $(\overline{\gamma}_{DR}/\Gamma_{DR})_M$ to keep the average number of transmit relays is given from (5). Define $F$ ,

$$F = \frac{(\overline{\gamma}_{RD}/\Gamma_{RD})_{req}}{(\overline{\gamma}_{DR}/\Gamma_{DR})_M} \quad . \tag{8}$$

Assume the average receiving power is inversely proportional to the power $\alpha$ of the distance between the transmitter and the receiver. And assume the noise power of each node is identical. The average receiving power is written by

$$\overline{\gamma}_{RD} = P_R d^{-\alpha}/N, \quad \overline{\gamma}_{DR} = P_D d^{-\alpha}/N \tag{9}$$

where $P_R$ and $P_D$ are transmit power of relay and destination, respectively. $d$ is the distance between the transmitter and the receiver, and $N$ is the noise power of a receiver. The corresponding transmit power of a relay, obtained by substituting (9) into (8), is

$$P_{R,req} = \left( \frac{1}{\Gamma_{DR}} \right)_M \times F \times \Gamma_{RD} \times P_D \quad . \tag{10}$$

## 4 Numerical Results

For numerical analysis, we assume the independent and identically distributed Rayleigh fading channel. Fig. 2 shows the threshold versus the average number of relays for different $M_{DOT}$ with $K = 10$, $\overline{\gamma}_{DR} = 10dB$. When $\overline{\gamma}_{SR} / \Gamma_{SR}$ equals 10 dB, the average number of the transmit relays are 8, 6, and 4 for the threshold $\Gamma_{DR}$ = 1.23, 4.11, and 8.16, respectively. It is noticed that the average number of the transmit relays decreases as the threshold increases. Also when the average number of the transmit relays are fixed, the threshold increases as the average SNR of S-R path increases.

The outage probability and average SNR of R-D path is shown for different $K$ with $\overline{\gamma}_{SR} / \Gamma_{SR} = 10\ dB$ in Fig. 3. As we expected, this figure shows the improvement in outage probability as the number of the relays increases. At the given outage probability of $P_{out} = 1 \times 10^{-3}$, the normalized average SNR $\overline{\gamma}_{RD} / \Gamma_{RD}$ is 2.7, 2.25, and 1.99 dB for $K$ = 8, 10, 12, respectively.

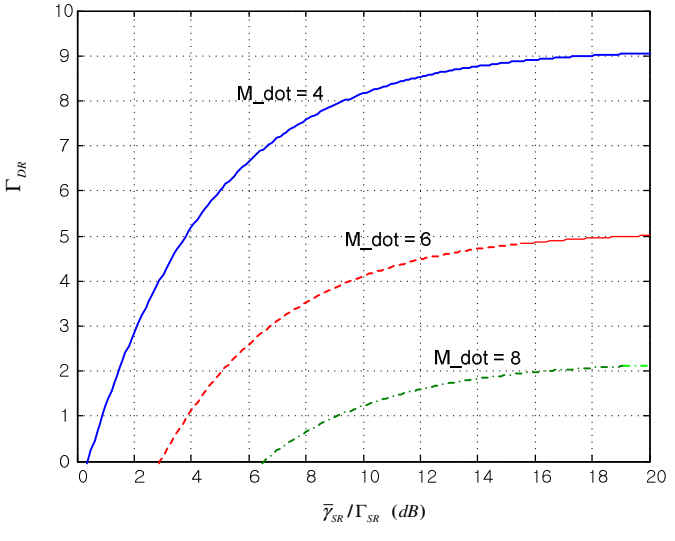



**Fig. 2.** Threshold $\Gamma_{DR}$ vs. average number of transmission relays ( $K = 10$, $\overline{\gamma}_{DR} = 10dB$ )

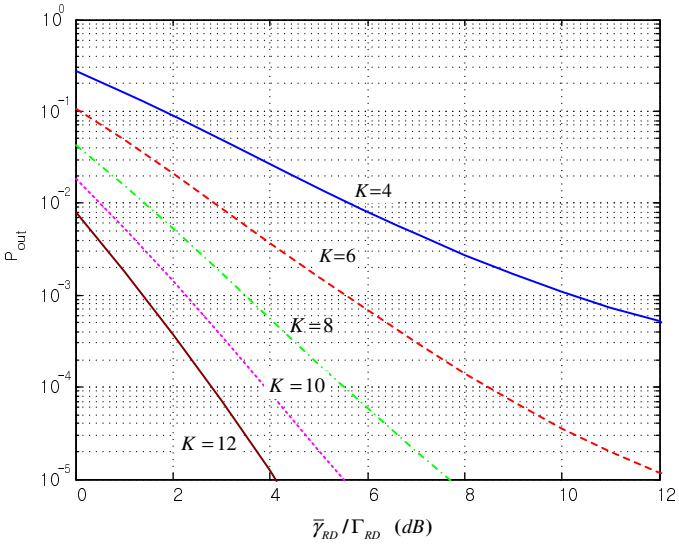



**Fig. 3.** $\bar{\gamma}_{RD}$ vs. outage probability ( $\bar{\gamma}_{SR}/\Gamma_{SR}=10dB$ )

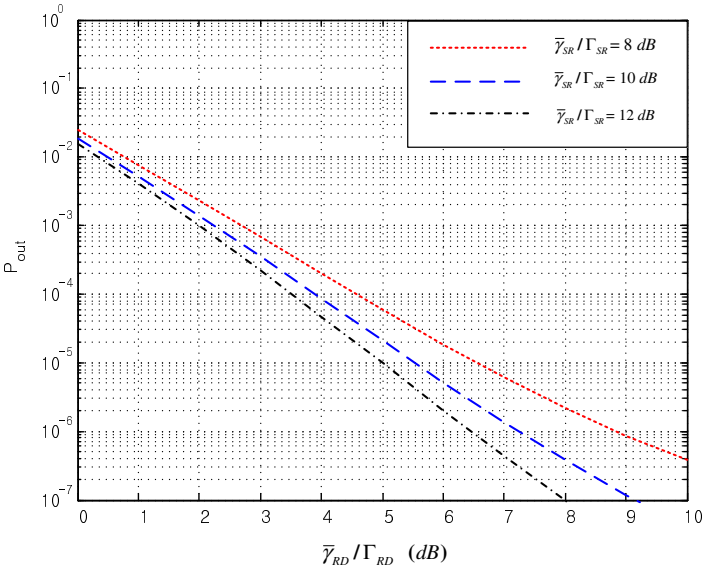



**Fig. 4.** $\bar{\gamma}_{RD}/\Gamma_{RD}$ vs. outage probability ( $K=10$ )

Fig. 4 shows $\bar{\gamma}_{RD}/\Gamma_{RD}$ versus outage probability for different $\bar{\gamma}_{SR}/\Gamma_{SR}$ with $K=10$. The figure shows the improvement in outage probability as $\bar{\gamma}_{SR}/\Gamma_{SR}$ increases. And we noticed that $(\bar{\gamma}_{RD}/\Gamma_{RD})_{req}$ decreases to keep the same outage probability as $\bar{\gamma}_{SR}/\Gamma_{SR}$ increases.

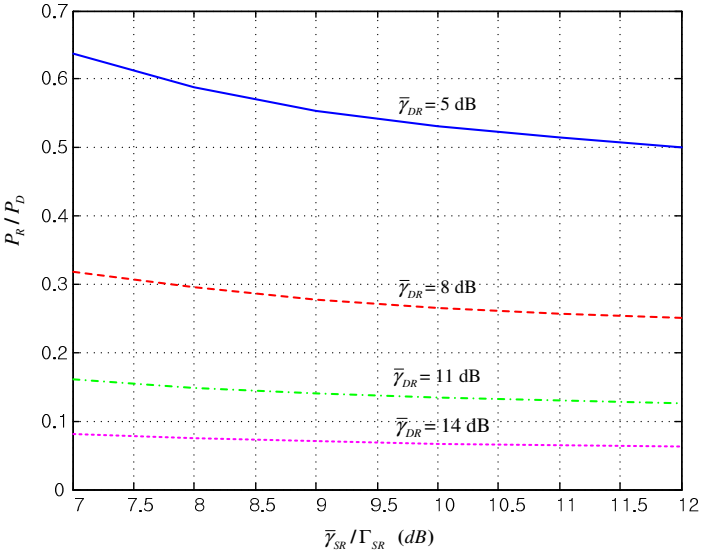



**Fig. 5.** Transmit power of the power controlled relay node ( $K = 10$, $\Gamma_{RD} = 1$ )

The ratio of the transmit power of a relay to that of the destination is shown in Fig. 5 (K=10, $P_{out} = 1 \times 10^{-3}$ ). It is noticed that the transmit power of relay node decreases as $\bar{\gamma}_{SR} / \Gamma_{SR}$ increases. To satisfy outage probability of $P_{out} = 1 \times 10^{-3}$, the required ratio of the transmit power of relay to that of destination decreases as the transmit power of destination(i.e. $\bar{\gamma}_{DR}$ ) increases.

## 5 Conclusions

A receiver generally has the fixed number of branches at a destination. Under this condition, the excess numbers of transmit relays do not contribute to the performance improvement, and only waste the network power and radio resources. In this paper, we propose a method which is adjusting the average number of transmit relays, and controlling the transmit power of the relays locally to satisfy the required performance in Double opportunistic transmit (DOT) relaying system.

Firstly, the average number of transmit relays can be controlled by adjusting the threshold. It is shown that the average number of transmit relays decreases as the increase of threshold and vice versa. Secondly, the ratio of the transmit power of a relay to that of the destination to satisfy the required outage probability decreases as the transmit power of the destination increases. This proposed method can be applied to the DOT system with the fixed number of branches of a receiver at destination. With this method, the average number of transmit relays can be controlled and the transmit power of a relay to satisfy the required system performance can be adjusted locally.

**Acknowledgement.** This work was supported in part by the project of development of local tactical technology which was conducted by the Ministry of Knowledge Economy of the Korean Government (project no. 70007243) and in part by the Korea Research Foundation (KRF) grant funded by the Korea government (MEST) (No. 2011-0003512).

## References

1. Barbeau, M., Kranakis, E.: Principles of ad hoc networking. John Wiley & Sons Ltd., Chichester (2007)
2. Hasna, M., Alouini, M.-S.: Performance of Two-hop Relayed Transmissions over Rayleigh Fading Channels. In: Vehicular Technology Conference 2002, vol. 4, pp. 1992–1996 (2002)
3. Lanememan, J.N., Wornell, G.W.: Energy Efficient Antenna Sharing and Relaying for Wireless Networks. In: WCNC 2000, vol. 1, pp. 7–12 (2000)
4. Emamian, V., Anghel, P., Mostafa Kaveh, M.: Multi-user Spatial Diversity in a Shadow-fading Environment. In: Vehicular Technology Conference 2002-Fall, vol. 1, pp. 573–576 (2002)
5. Anghel, P.A., Kaveh, M.: Exact Symbol Error Probability of a Cooperative Network in Rayleigh-fading Environment. IEEE Trans. on Wireless Communications 1(5), 1416–1421 (2004)
6. Beaulieu, N.C., Hu, J.: A Closed Form Expression for the Outage Probability of Decode-and-forward Relaying in Dissimilar Rayleigh Fading Channels. IEEE Communications Letters 10(12), 813–815 (2006)
7. Zhio, Y., Adve, R., Lim, T.: Outage Probability at Arbitrary SNR with Cooperative Diversity. IEEE Communications Letters 9(8), 700–702 (2005)
8. Kim, N.-S., Lee, Y.H.: Double Opportunistic Transmit Relaying System with GSC for Power Saving in Rayleigh Fading Channel. In: COMM 2010, Romania, pp. 368–370 (2010)
9. Kim, N.-S., Lee, Y.H.: Double Opportunistic Transmit Cooperative Relaying System with GSC in Rayleigh Fading Channels. Journal of the Korean Institute of Electromagnetic Engineering and Science 10(4), 270–275 (2010)
10. Goldsmith, A.: Wireless Communications. Cambridge University Press, Cambridge (2005)
11. Simon, M.K., Alouini, M.-S.: Digital Communications over Fading Channels. John Wiley & Sons, Chichester (2000)
12. Yang, H., Alouini, M.-S.: MRC and GSC Diversity Combining with Outage Threshold. IEEE Trans. on Vehicular Technol. 54(3), 1081–1090 (2005)
13. Yang, H.-C.: New Result on Ordered Statistics and Analysis of Minimum-selection Generalized Selection Combining (GSC). IEEE Trans. on Wireless Communications 5(7), 1876–1885 (2006)
14. Gesbert, D., Alouini, M.-S.: How Much Feedback is Multi-user Diversity Really Worth? In: IEEE International Conference on Communications (ICC), pp. 234–238 (2004)

# GROUP: A Gossip Based Building Community Protocol*

Ranieri Baraglia[1], Patrizio Dazzi[1], Matteo Mordacchini[2], Laura Ricci[3], and Luca Alessi[3]

[1] HPC Lab, ISTI-CNR, Italy
{ranieri.baraglia,patrizio.dazzi}@isti.cnr.it
[2] Ubiquitous Internet Lab, IIT-CNR, Italy
matteo.mordacchini@iit.cnr.it
[3] Dept. Computer Science, University of Pisa, Italy
ricci@di.unipi.it, alessi.it@gmail.com

**Abstract.** The detection of communities of peers characterized by similar interests is currently a challenging research area. To ease the diffusion of relevant data to interested peers, similarity based overlays define links between similar peers by exploiting a similarity function. However, existing solutions neither give a clear definition of peer communities nor define a clear strategy to partition the peers into communities. As a consequence, the spread of the information cannot be confined within a well defined region of an overlay. This paper proposes a distributed protocol for the detection of communities in a P2P network. Our approach is based on the definition of a distributed voting algorithm where each peer chooses the more similar peers among those in a limited neighbourhood range. The identifier of the most representative peer is exploited to identify a community. The paper shows the effectiveness of our approach by presenting a set of experimental results.

## 1 Introduction

The exploitation of social information filtering solutions is becoming more and more relevant in a world where there is a growing need to rapidly access and be aware of many types of distributed resources like Internet pages, shared files, on line products and news. The definition of these solutions implies to face several challenges such as scalability and the management of highly dynamic information.

Several information filtering algorithms have been investigated for centralized architectures, while their definition is still a challenging issue for highly distributed systems, like the P2P ones. The basic block required for defining these algorithms is a protocol for spreading information over the P2P overlay. Among different proposals, the one exploiting the gossip ([6,7]) paradigm looks

* The research leading to these results has received funding from CONTRAIL (EU-FP7-257438),RECOGNITION (EU-FP7-257756) and SCUBE (EU-FP7-215483) EU projects.

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 496–507, 2011.

very promising. These proposals stem from their ability to easy cast information among a large set of interconnected nodes, even if the nodes frequently join and leave the network. These protocols present good reliability, fault tolerance and scalability. Each node maintains a partial view of the other peers in the network, the protocol is highly adaptable and the exchange of gossip messages allows the natural removal of old information in favour of new one.

The gossip based approach has been exploited to build similarity based overlays [12,13], where a link between similar peers is defined by exploiting a point-to-point similarity function. Each peer exploits a random sampling layer to discover new nodes in the P2P network so that similar peers may get in touch and new links can be defined between them.

In the proposal we have presented in [8], each peer builds its own neighbourhood which is then exploited for exchanging information related to common interests. The neighbourhood of each peer is computed by considering a similarity function based on users' profiles, so that each peer may identify those characterized by similar interests. [8] exploits Cyclon [12] and Vicinity [13] protocols, where the former allows the random sampling of the network, while the latter is exploited to define the similarity overlay. However, [8] neither investigates the concept of community in a P2P network nor defines a strategy to partition the peers into a set of communities.

The task of detecting communities in a distributed environment is a complex issue because while each peer has a local vision of the overlay, the detection of communities requires the aggregation of peers which are not directly connected in the overlay. A community detection algorithm has to identify the boundaries of each community in the similarity overlay so that the spreading of the information may be confined within a well defined region of the overlay, i.e. the portion of the network including peers interested in that information. The algorithms proposed in the literature [6,7] do not enable the discovering and the characterization of similarity groups, i.e. large groups of peers with an high degree of similarity, they only links similar peers. In order to let these group to emerge, the collective knowledge of the peers in a network has to be exploited.

This paper proposes GROUP (**G**ossip-based peer-to-pee**R** c**O**mm**U**nity building **P**rotocol), an asynchronous protocol for building communities in P2P networks. GROUP exploits Vicinity to define the similarity based overlay and a distributed election algorithm to detect the communities over the overlay returned by Vicinity. The algorithm finds out the most suitable peer able to characterize a set of peers with similar interests. Each community is paired with a representative peer and its profile is exploited to represent the community. Peers can choose the community to adhere by comparing their profile against that of the representative. At the end of the protocol, each peer belongs to the community characterized by the highest profile similarity. Community profiles allow to have a more extensive representation of the more deep characteristics that link together nodes in the network and could be exploited for a more efficient and effective information diffusion among peers.

The paper is organized as follows. Section 2 describes solutions already present in the literature. Section 3 includes a more precise definition of the problem to be faced and describes the proposed solution. Section 4 discusses the experiments conducted to evaluate the proposed approach. Finally, Section 5 presents the conclusions.

## 2 Related Work

Several solutions have been proposed in the past for defining P2P overlays suitable for spreading and retrieving information in an efficient way. Such solutions exploit different techniques such as Gossip [4,8,6,7,2] or Semantic Overlay Network (SON) [1].

The GosSkip[4] system is a self organizing and fully distributed overlay that provides a support to data storage and retrieval in peer-to-peer environments. It is built using a gossip protocol that organizes peers so that they form an ordered double-linked list. Each a peer is associated with a single item of data and it has a name that describes the semantics of the object to which is associated. These names follow a total and deterministic order. So, the position of an element is fully determined by its name. For the information distribution, its gossip protocol maintains $O(log(N))$ peer states, and has a routing cost of $O(log(N))$. The association of links to published object can lead to a very large number of connections. This is especially true in networks where the number of objects shared by each user is large. Furthermore, the use of GosSkip could be difficult when searching without knowing the exact name which identifies the item you are looking for.

The aim of the study conducted in [1] is to reduce the search time of queries executed in peer-to-peer networks based on a random overlay, and at the same time to maintain a high degree of autonomy of the nodes. The authors propose to define node connections based on the content shared by the peers. Thus, each subset of semantically related nodes form a Semantic Overlay Network (SON). Queries are routed to the proper SON, increasing the chances to find matching files, and reducing the search load on nodes that have unrelated content. The main disadvantages of this solution is the rigid predefined structure of the SON-based overlay network.

The authors of [9] propose a solution for Peer Data Management Systems (PDMS). A PDMS consists of peers, viewed as autonomous and heterogeneous data sources, having their own content modelled through schemas. The authors propose a strategy for clustering related peers through the maintenance of a multilayer network organization. Each layer represents different semantic concepts. Each peer takes part to its most semantically similar layers. Within each layer, it gathers with its most similar nodes using a fine-grained neighbour selection mechanism. A critical aspect of this solution is the evolution of the interests of a peer. If this leads to changes in the peer's semantic concepts, it may triggers a distributed mechanism to reorganize the overlay network, involving all the neighbouring nodes belonging to the related SONs.

The protocol described in [10] presents a distributed approach for discovering connectivity-based clusters in P2P networks. It discovers clusters based on the network graph connections around a given set of nodes using local peer knowledge. The main drawback is that the quality of the clusters highly depends on the initial choice of those peers. Moreover, this protocol needs an explicit management of joining and leaving peers.

## 3 GROUP: Protocol Definition

We consider a scenario where each peer is associated with a network's user which is characterized by a profile including $k$ terms. The $k$ terms characterize the content to which the user is interested, for instance the documents that he/she has previously accessed. This set of terms can be built incrementally, for instance according to predefined thresholds that fix when a term has to enter/leave the profile of the user.

In this scenario the goal is to achieve a partition $\mathscr{P}_I = \{P_1, \ldots, P_s\}$ of the peers such that each $P_i$ includes a subset of the peers in $\mathscr{P}_I$ which are similar in term of their profiles. Therefore each $P_i$ represents a different community, which can be exploited to define more focused dissemination of the information in the P2P network.

Previous approaches (e.g. Vicinity) try to gather similar nodes by putting them in direct contact when they share similar interests. On the other way round, the local knowledge of the peers typically does not allow a good identification of the features which may characterize the communities of peers. Thus, a protocol based on the collaboration between peers has to be exploited in order to reach a collective agreement over the definition of the communities and of the communities representatives.

Our aim is to detect communities and to identify them so that each peer may characterize itself as member of a community. We have chosen to exploit the profile of a peer inside each community as the community identifier. Such peer is the one that is chosen by the community peers as their *best representative*.

In order to allow each peer to be aware of those peers belonging to the network, the Cyclon and the Vicinity protocols are exploited. While the former is exploited to define a random sampling protocol, the latter is used to put in contact each peer with the most similar ones. The paper does not present these gossip protocols as they are well defined in the literature [12,13].

Instead we focus on the definition of the community detection algorithm. Our approach for building communities is based on a distributed voting algorithm which is executed over the similarity based overlay returned by Vicinity. The algorithm leads to the election of a set of communities representatives peers. Each elected peer, together with the peers that have contributed to its election constitutes a community. The community is represented by the profile of the peer which have been elected.

The detection of the representative of each community is structured in the following stages: Similar Neighbours Detection, Potential Candidates Selection, Representative Election.

**Algorithm 1.** Similar Neighbours Detection

```
1: Order NEIGHBORS by similarity;
2: Let BESTN = Top-k(NEIGHBORS);
3: for all b ∈ BESTN do
4:     if d(p, b) ≤ (1 − neighbor_threshold) then
5:         Send Vote to b;
6:     else
7:         Break;
8:     end if
9: end for
```

**Algorithm 2.** Potential Candidates Selection

```
1: Let CANDIDATES = ∅;
2: for all n ∈ {NEIGHBORS ∪ {p}} do
3:     if VotesRcvd(n) ≥ REPRTHR then
4:         Let CANDIDATES = CANDIDATES ∪ {n}
5:     end if
6: end for
7: Let break=false;
8: if CANDIDATES = ∅ then
9:     Let CANDIDATES = NGHREPRS
10:    if CANDIDATES = ∅ then
11:        Let CANDIDATES = p ad interim
12:        Let break=true;
13:    end if
14: end if
15: if not(break) then
16:    Let R = min_{c ∈ CANDIDATES} d(p, c)
17:    Send ReprVote to R
18: end if
```

**Algorithm 3.** ActiveThread

```
1: while true do
2:
3:     if Timer t expired then
4:         Reset t;
5:         SimilarNeighbourDetection();
6:         wait(t′)
7:         PotentialCandidateSelection();
8:         wait(t″)
9:         RepresentativeElection();
10:    end if
11:
12: end while
```

**Algorithm 4.** PassiveThread

```
1: while true do
2:     Receive MSG as MESSAGE
3:     M-TYPE = MSG.type;
4:     M-TSTAMP = MSG.t-stamp;
5:     M-CONT = MSG.content;
6:     M-SENDER = MSG.sender;
7:     if M-TYPE = Vote then
8:         Add M-CONT to VQUEUE with M-TSTAMP
9:         Send VQUEUE.size to M-SENDER
10:    else if MTYPE = ReprVote then
11:        Add M-CONT to REPVQUEUE
12:                with M-TSTAMP
13:        Send REPVQUEUE.size to M-SENDER
14:    end if
15: end while
```

The three phases of the voting protocol are periodically executed by an active thread created by each peer. Since each phase implies the exchange of some messages with the neighbours, a corresponding passive thread whose main task is the reception of the messages is activated by each peer. Note that this is the typical behaviour of a gossip protocol.

Alg.3 and Alg.4 describe the active and the passive thread, respectively. In the active thread the different voting phases are separated by exploiting a set of timers. The goal of the first voting phase is to detect the most similar peers in the neighbourhood of a peer. This phase, detailed in Alg.1, consists in a preliminary voting procedure in which each peer votes its most similar peers, i.e. its best neighbours in term of profiles similarity. Each peer arranges its neighbours in decreasing order with respect to their similarity value, and gives a vote to, at most, the top-k neighbours, without considering the peers whose similarity is lower than a *neighbour threshold*. k defines the maximum number of neighbours that can be voted by a peer, while the neighbour threshold represents the peer

similarity value under which two peers are not considered similar, and therefore should not be selected to be voted. $d(p, n)$ denotes the distance in the similarity space between the peer $p$ and its neighbour $n$. Each vote sent by a peer $p$ to its neighbour $n$ is received by the passive thread of $n$ and it is inserted into a queue which is shared with the active thread of the same peer. Furthermore, the passive thread of $n$ sends a reply message to $p$ including the number of votes received by $n$ in the current voting phase. This information is in turn received by a further passive thread of $p$ which is not presented to make the presentation simple. Different similarity metrics can be exploited in this phase, for instance Jaccard or cosine similarity [6].

After this phase, each peer has notified to all its neighbours the number of votes it has received. At this point the second phase starts where each peer chooses, among its neighbours, its potential candidate to represent its community. The potential candidate is chosen among the neighbours which have received in the previous phase a number of votes (VotesRecvd(n)) higher than the representative threshold $ReprThr$, i.e. the minimum number of votes obtained by a peer to be voted as a potential candidate. Among the potential candidates, the neighbour characterized by the higher degree of similarity is chosen. If no neighbour has received a sufficient number of votes, the peer votes itself. By tuning the representative threshold parameter, it is possible to influence the total number of representatives, and, assuming to keep fixed the other parameters, the cardinality of each community of peers. Alg.2 shows this voting phase. Again, the potential candidate vote is received by the passive thread which notifies to the sender the number of potential candidate votes received.

Finally, in the *Representative Election Phase*, each peer elects its actual representative by considering both the potential candidate votes which it has been received by itself and those received by its neighbours. We do not show the pseudo code of this phase because of its complexity.

If a peer $p$ has collected the highest number of potential candidate votes with respect to its neighbourhood, it considers itself as its own representative. Otherwise, a potential representative R become actual representative if, among the peer's neighbour, R is the one that has received the highest number of potential candidates votes. If there are two potential representatives receiving the same number of votes, p chooses its most similar one. Anyway, there are further scenarios to be addressed. If $p$ discovers that its voted representative $R$ has not voted for itself, but it has chosen in turn as its own actual representative another peer $R_0$ that is within a distance $\epsilon$ from R, then $p$ changes it choice by selecting $R_0$ as its actual representative. If $R$ has chosen $R_0$ as its representative, but they are sufficiently separated ($d(R, R_0) > \epsilon$), $p$ maintains its choice, i.e. it chooses $R$ as its representative. The link between $R$ and $R_0$ represents a connection among their respective communities, augmenting the spreading of information inside the network. A further special case occurs when none of the potential representatives of $p$ can be considered its actual representative, because it has received a few votes. In this case $p$ asks to its neighbours for their actual representatives and it selects the one characterized by the highest similarity value. This process

leads the peers in the network to spontaneously gather into communities, each community including all the peers that have chosen the same representative. As mentioned before, each vote has a limited life time, exactly as it happens in a democracy where a mandate expires after a certain amount of time. Thus, periodically, at predefined intervals of time, the election procedure is repeated. The continuous refresh of information is ensured by the activities performed by peers for building their interest-proximity networks that put in contact similar nodes. These activities are supported by an epidemic diffusion of the information, both as far as concerns the user profiles and the representative votes. As it happens to representatives, communities are dynamic entities subjected to changes. Beyond the joins and leaves due to peers churn, the community may be split or merged. It is worth noticing that all the operations described so far are performed by each peer individually, without any form of synchronization with other ones. The only interaction is due to the exchange of information, for exchanging both gossip updates that includes peers' profiles and votes. When a proper defined time interval has elapsed, each peer independently starts a new voting turn and at the end of this phase it is able to cope with new situations, like the arrival or the departure of other peers. The underlying gossip mechanism allows a peer $p$ to obtain an up to date situation of similar peers in its neighbourhood. Hence, when a new voting phase starts, new peers will be considered, and old or disconnected ones will be taken into account no more. If a high number of updates in p's neighbourhood occurs, $p$ will choose a new community, possibly joining previously created communities or splitting its old community and defining a new group. Thus, no explicit mechanism to handle joins or splits of communities is required. The experimental section includes some highlights related to these mechanisms.

## 4 Evaluation

This section shows the evaluation of the our approach through a set of experiments which show its effectiveness. The implementation has been realized by the OverlayWeaver Peer-to-Peer framework [11]. In order to test the "real" effectiveness of our approach, we exploit a real dataset instead of using a synthetic one. More in detail, we have used a bio-* subset of the dataset released by Mendeley [5,3], a company that produces a publication management tool that run on the client side but takes advantages of a web storage where each user can store the most interesting publications he/she owns. The dataset released contains a set of anonymous users, each one with a set of references indicating the papers owned by it. The subset we choose is made of 2800 users, each of them with at least 20 papers. For each user we retrieved the content of the papers in his/her profile, we filtered out the stop-words and we extracted the most frequent terms that we used as user profile. The similarity metric used in the experiments is the Jaccard similarity measure [6]. The timer $t$ value was set to a time equivalent to 2 cycles, as justified later in this section. According to what we stated in the problem description section (Section 3), our goal consists in the definition of

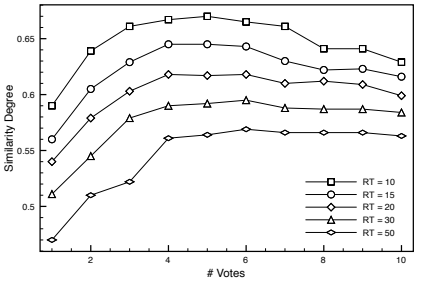



**Fig. 1.** Number of votes impact: Similarity Degree

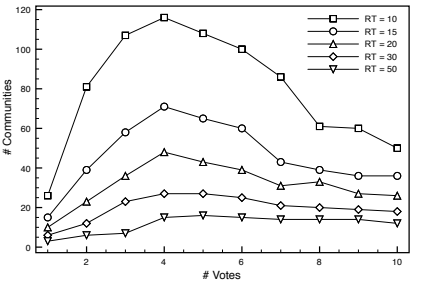



**Fig. 2.** Number of votes impact: Number of Communities

an algorithm for building up explicit communities composed by peers sharing a common interest, whose size should not be too small as well as too big and, possibly, independent from the network size. In order to evaluate the ability of our approach to address this goal we have measured the effectiveness of the results by considering the internal community similarity, as well as the total number of communities and, as a consequence, their mean size. To measure the mean internal community similarity we use the following Similarity Degree measure:

$$\frac{1}{N} \times \sum_{n \in N} d(n, R(n))$$

where $N$ is the whole set of peers belonging to the network, $R(n)$ is the representative of peer $n$ and $d$ is the distance function defined in the previous section. Since each peer autonomously chooses the community to join, this measure is useful to see whether its choice gives it a good representative. Good representatives ensure a sufficiently high grade of similarity with their community members, thus enabling an effective communication on the network. This factor has to be coupled with proper community sizes, in order to check that the system does not create communities made of too few members, thus breaking the network into small groups and vanishing the effect of peer gathering. We analyse both these aspects in the following experiments.

*Number of Votes and Representative Threshold impact:* Figure 1 and Figure 2 show the effects of varying the number of votes a peer can give, with respect to the Similarity Degree and the number of communities. This is shown when different *representative_threshold* (RT in the figures) are used. The network size is fixed to 2800. Figure 1 shows that the highest values of similarity degree can be achieved when the *representative_threshold* is low. This is a quite expected behaviour. Indeed, a low threshold brings to the creation of a greater number of communities (as it is shown also in Figure 2). A higher number of communities means, in turn, a smaller number of peers per community, hence a (potential) greater internal homogeneity. Anyway, communities with too few members can be less useful when used to exchange information. It is interesting to note that

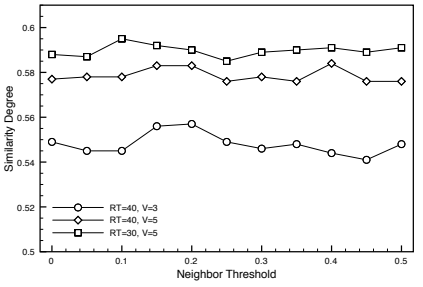



**Fig. 3.** Neighbor Threshold impact: Similarity Degree

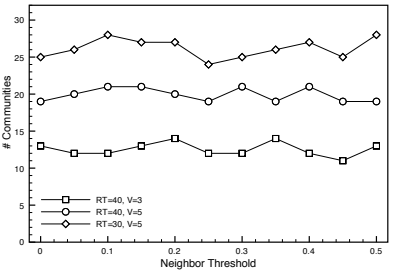



**Fig. 4.** Neighbor Threshold impact: Number of Communities

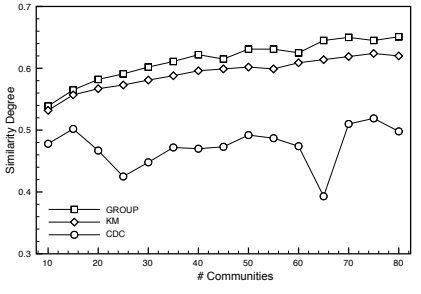



**Fig. 5.** Internal Community Cohesion: Similarity Degree

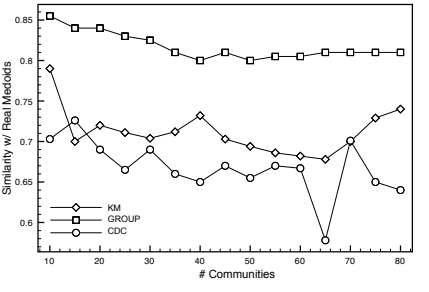



**Fig. 6.** Internal Community Cohesion: Similarity against the optimal medoid

both the number of communities and consequently the highest value of Similarity Degree are achieved when the number of votes a peer can express is 4 or 5, almost independently from the *representative_threshold* value.

*Neighbor Threshold impact:* Figure 3 and Figure 4 show the changes on the Similarity Degree and the number of communities obtained by using different values of the *neighbor_threshold* parameter. Even in this case the network size is fixed to 2800 nodes. Different *representative_threshold* are used as well as with different number votes (V in the figures) that can be expressed by a peer. As described above, it is easy to see how different representative thresholds brings up to the creation of very a different number of communities and a quite different result in terms of Similarity Degree, however both figures show how the different *neighbor_threshold* value has almost no relevant effects both in terms of communities number and in terms of Similarity Degree.

*Internal Community Cohesion:* Figure 5 and Figure 6 show a comparison among three different algorithms for grouping nodes, namely GROUP, the CDC algorithm [10] and K-Means, used as a centralized clustering algorithm. The network size is 2800 peers, the representative threshold is 30 and the number of votes is 4. The last two parameters are derived from the previous experiments.

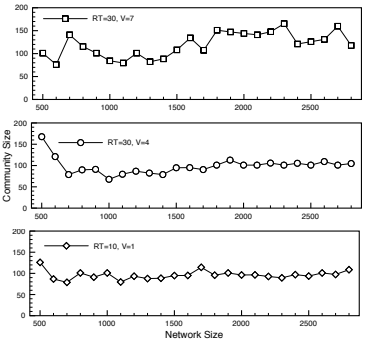



**Fig. 7.** Communities Mean Size

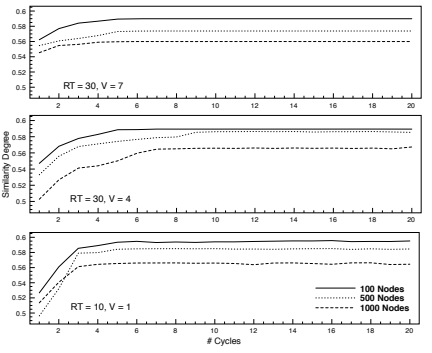



**Fig. 8.** Impact of Network Instability

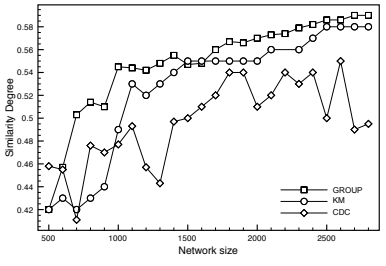



**Fig. 9.** Approach Scalability

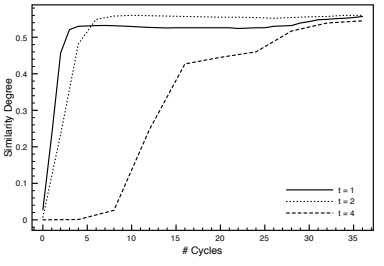



**Fig. 10.** Impact of different timers

The comparison has been conducted measuring both the Similarity Degree of communities and computing the distance between the community representative and the real medoid of the communities formed by each algorithm. It is easy to see that how our approach is able both to build communities characterized by a higher internal similarity degree and to elect better representatives, i.e. representatives that have near the "real" centers (in term of similarity) of the communities formed by GROUP.

*Communities Mean size:* Figure 7 shows the effect of the network size to the mean size of the communities. We used three different settings of thresholds and votes. This figure shows that the network size has basically little or no influence on the size of the communities. Indeed, the mean community size is about 100 elements. This is due to self-organization mechanism, that spontaneously avoids the creation of too big communities, while ensuring a sufficient number of members. This result is in line with what we stated in Section 3.

*Network Instability:* Figure 8 shows the impact of instability in the network due to the presence of failing peers and the ability of our approach to re-build communities characterized by an high Similarity Degree. Namely, this figure shows the results achieved on a initial network of 2800 nodes by suddenly

(i.e. without any graceful departure mechanism) removing 100, 500 and 1000 peers respectively. We used three different settings of votes and thresholds pairs. Cycles can be seen as time units. The figure shows that our approach is able to recover from these situations and to adapt almost immediately the communities. Clearly, when removing more nodes, the final community has a lower Similarity Degree, due to the fact that, probably, some of the most similar peers have left the network. Moreover, the effects of failing peers are more relevant with the lowest values of *representative_threshold* and votes. As noted in the first set of experiments, those settings lead to smaller communities. Thus, removing elements from them can easily disrupt their internal cohesion. Larger communities (highest values of *representative_threshold* and votes) suffer less in this scenario. All the settings converge in a very limited number of cycles (less than 10), thus showing the robustness of the proposed approach.

*Scalability:* Figure 9 shows the scalability of our approach. The figure shows the comparison among GROUP, CDC and K-Means regarding their ability to achieve a high Similarity Degree value when the network size changes. Our solution has been tested using a threshold of 30 votes and allowing 4 votes per peer. K-Means has run using the same number of communities that came out from GROUP. It is easy to note that our approach performs better than the other algorithms with almost all tested network sizes. In particular, it shows a clear trend to improve its performance when the network size increases.

*Timer Impact:* Figure 10 depicts the impact of choosing different values of the timer $t$ that separates vote waves. We used three different values: 1, 2 and 4 cycles respectively. Results show that a longer time interval implies a slower convergence of the algorithm, as one may expect. Anyway, too frequent elections may be influenced by the partial convergence of the underlying similarity layer. It causes a fast achievement of suboptimal values, that tend to be optimal only with an extra amount of cycles. A value that let the similarity layer to converge but leave a still small time from one wave to another, seems to achieve the best results.

## 5 Conclusions

This paper presents GROUP, a distributed P2P community building protocol whose goal is to identify a set of representatives peers characterizing user communities. A representative is a peer elected by a set of peers on the basis of their similarity with it. The approach is based on a set of P2P epidemic protocols. Several tests conducted using the Mendeley dataset have shown the effectiveness of our approach in comparison to a set of other ones. We are currently integrating *GROUP* with a locality sensitive DHT based on min wise independent permutations which indexes the communities. In this way a new peer joining group is able to efficiently retrieve and choose the more similar communities.

## References


1. Aberer, K., Cudré-Mauroux, P.: Semantic overlay networks. In: Proc. of the Int. Conference on Very Large Data Bases, vol. 31, p. 1367. Citeseer (2005)
2. Bertier, Frey, Guerraoui, Kermarrec, Leroy: The Gossple Anonymous Social Network. In: Proc. of the ACM/IFIP/USENIX 11th Middleware Conference (Middleware 2010), pp. 814–829 (2010)
3. Jack, K., et al.: Mendeley's reply to the DataTEL challenge. In: Proc. of the 1st Workshop on Recommender Systems for Technology Enhanced Learning (2010)
4. Guerraoui, R., Sidath, B.H., Kermarrec, A.M., Le Fessant, F., Huguenin, K., Rivière, E.: Gosskip, an efficient, fault-tolerant and self organizing overlay using gossip-based construction and skip-lists principles. In: Sixth IEEE International Conference on Peer-to-Peer Computing, 2006 Ratnasamy (2001)
5. Henning, V., Reichelt, J.: Mendeley-A Last. fm For Research? In: IEEE Fourth Int. Conference on eScience, 2008, pp. 327–328. IEEE, Los Alamitos (2009)
6. Jelasity, M., Guerraoui, R., Kermarrec, A.M., van Steen, M.: The peer sampling service: experimental evaluation of unstructured gossip-based implementations. In: Jacobsen, H.-A. (ed.) Middleware 2004. LNCS, vol. 3231, pp. 79–98. Springer, Heidelberg (2004)
7. Jelasity, M., Voulgaris, S., Guerraoui, R., Kermarrec, A.M., van Steen, M.: Gossip-based peer sampling. ACM Trans. Comput. Syst. 25(3), 8 (2007)
8. Mordacchini, M., Baraglia, R., Dazzi, P., Ricci, L.: A p2p recommender system based on gossip overlays (prego). In: CIT, pp. 83–90 (2010)
9. Penzo, W., Lodi, S., Mandreoli, F., Martoglia, R., Sassatelli, S.: Semantic peer, here are the neighbors you want! In: Proc. of the 11th Int. Conf. on Extending Database Technology: Advances in Database Technology, pp. 26–37. ACM, New York (2008)
10. Ramaswamy, L., Gedik, B., Liu, L.: A distributed approach to node clustering in decentralized peer-to-peer networks. IEEE Transactions on Parallel and Distributed Systems, 814–829 (2005)
11. Shudo, K., Tanaka, Y., Sekiguchi, S.: Overlay weaver: An overlay construction toolkit. Computer Communications 31(2), 402–412 (2008)
12. Voulgaris, S., Gavidia, D., Van Steen, M.: Cyclon: Inexpensive membership management for unstructured p2p overlays. Journal of Network and Systems Management 13(2), 197–217 (2005)
13. Voulgaris, S., van Steen, M.: Epidemic-style management of semantic overlays for content-based searching. In: Cunha, J.C., Medeiros, P.D. (eds.) Euro-Par 2005. LNCS, vol. 3648, pp. 1143–1152. Springer, Heidelberg (2005)

# A Study on Upload Capacity Utilization with Minimum Delay in Peer-to-Peer Streaming

Geun-Hyung Kim

Department of Visual Information Engineering, Dong-Eui University
995 Eomgwangno, BusanJin-Gu, Busan 614-714, Korea
geunkim@deu.ac.kr

**Abstract.** With the widespread deployment of broadband residential access networks and the development of media compression technologies, media streaming over the Internet has spread rapidly and attracted a large number of users and researchers. Currently, peer-to-peer (P2P) overlay networking is considered as a promising way to distribute large scale media streams over the Internet, due to the scalability, robustness, cost-effectiveness, and ease of deployment. However, there are still several fundamental questions: What is the maximum upload capacity utilization with minimum delay in the multi tree-based P2P streaming? How to maximize the peer's upload capacity utilization? What is the effect of peer out-degree on the upload capacity utilization? To address these problems, we investigate the effect of overlay topology and peer out-degree on the streaming delay and upload capacity utilization. Finally, we show that a carefully-designed overlay topology and peer placement strategy can achieve close-to-optimal delay bounds and upload capacity utilization on P2P streaming systems.

## 1 Introduction

As Internet protocol TV(IPTV) services and web-based video distribution services(e.g., YouTube, Joost, etc.) accelerate video distribution over the Internet, the demand for large-scale streaming architecture has increased. Conventional client-server architecture is not as scalable as the number of users continues to increase, because the server should supplement output links. However, the P2P architecture retrenches the investment in the server and the network by sharing the upload capacity of peers to distribute a part of the same content to other peers. Due to this advantage, lots of P2P streaming systems [1] [2] [3] [4] for live and on-demand streaming services have been deployed. However, all existing systems give long playback startup delays and high variable playback delays among peers [5].

Current P2P streaming systems can be mainly classified into two types: tree-based and mesh-based. In the tree-based P2P streaming systems, multi tree-based P2P streaming systems have been popular, since single tree-based P2P streaming systems have a fatal weakness that all leaf peers do not contribute their upload capacities. In the mesh-based P2P streaming systems, peers establish the

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 508–519, 2011.

peering relationship dynamically to download and upload a chunk-a small piece of the stream data. The main differences between both systems are in the overlay topology construction scheme and the chunk scheduling scheme. In spite of these differences, the trajectory of each chunk in both systems forms a spanning tree. Therefore, we model the P2P streaming systems as multiple layered overlay trees for analyzing delay and utilization.

The major issues in building the P2P streaming systems are to minimize the difference of presentation point among peers and to maximize the resource sharing of peers, e.g., upload capacity. Though there may be several factors that affect the performance bounds in the P2P streaming systems, the peer out-degree bound and the sharing upload capacity influence the performance significantly. Peer out-degree is defined as the total number of distinct neighbors to which the peer delivers chunks. In real environments, the peer out-degree is bounded, as users usually limit the sharing upload capacity. In addition, there is a significant constraint in P2P streaming systems, that a chunk should be delivered to all peers in time to play it back smoothly.

In this paper, we investigate the upload capacity utilization and the initial playback delay in multi tree-based P2P streaming systems under peer out-degree bound. In addition, we analyze the effect of an overlay topology on the streaming delay when the upload capacity and peer out-degree are given. We define the *pure leaf* concept and the *upload capacity utilization function* to analyze the upload capacity utilization of P2P streaming systems. Consequently, we investigate the maximum upload capacity utilization with minimum streaming delay under peer out-degree bound and describe the multiple overlay networks construction mechanism.

To the best of our knowledge, the research on what is the maximum upload capacity with guaranteeing minimum streaming delay bound and how to achieve it has not been done too much. Several studies have been performed on theoretical analysis to understand the streaming delay. The aim of this paper is to investigate theoretically the upload capacity utilization in P2P streaming system with minimum streaming delay and the multi-tree overlay construction algorithm.

The rest of this paper is organized as follows. In Section 2, we briefly discuss the related work. In Section 3, we formulate the maximum utilization problem with minimum delay in P2P streaming system and give numerical analysis result. Finally, conclusions are given in Section 4.

## 2 Related Work

The overlay architecture and chunk scheduling strategy are important parts, when building P2P streaming systems which provide the satisfied performance and efficient resource sharing. In terms of the overlay architecture, P2P streaming systems can be classified into two categories: tree-based and mesh-based. The tree-based P2P streaming systems [6] [7] [8] have explicitly constructed multiple spanning trees in which a peer pushes chunks actively to its children peers.

Initial tree-based P2P streaming systems[6] proposed application level multicast scheme to distribute a media stream. Splitstream[7] proposed multiple trees. It divides a media stream into multiple sub-streams and distributes each sub-stream over a separate tree. CoopNet[8] applied centralized multiple trees with MDC coding technique.

Recently, mesh-based P2P streaming systems[9] [10] have been widely adopted. In the mesh-based P2P streaming systems, the peering relationships are established/terminated dynamically based on the chunk availability and upload capacity on peers. Each peer selects a subset of peers in the P2P streaming systems as neighbors from which the peer can retrieve chunks. It periodically exchanges information about its chunk availability and upload capacity with neighbors. Tang *et al.*[11] proposed a push-pull mechanism, combining the strengths of tree-based mechanism and mesh-based mechanism, to achieve lower chunk delivery delay.

In order to play the media stream smoothly in all peers, peers should retrieve chunks of the media stream in time. That is, every chunk should traverse all peers in the P2P streaming systems in time, regardless of the overlay architecture. Therefore, its trajectory forms a directed spanning tree. The multi tree-based P2P streaming systems have several directed spanning trees. The construction scheme of these spanning trees may lead to the performance gap between the fundamental performance limits and the actual performance.

Despite of the deployment of several P2P streaming systems, theoretical analysis of performance limits of the P2P streaming systems was poorly understood. Since 2007, several works have been performed to analyze the fundamental limits of P2P streaming performance, such as streamingcapacities[12], streaming delay[13] [14] [15] [16] [17], and sustainable streaming rate[18] [19] [20].

Y. Liu[13] derived the minimum average delay bound for realtime P2P streaming systems without peer out-degree bound and proposed a simple snow-ball streaming algorithm to approach the derived minimum average delay bound. S.Liu *et al.*[18] studied several performance limits on streaming rate, minimum server load, and minimum tree depth of the P2P streaming systems with peer out-degree bound. C. Feng *et al.*[14] generalized the minimum delay bound in [13] by relaxing the assumption that the upload capacity of each peer should be divisible by the streaming rate. In addition, they proposed a graph labeling segment scheduling algorithm to achieve the maximum sustainable streaming rate and characterized the gap between the fundamental limits and the actual system performance in mesh-based P2P streaming systems. However, they did not consider peer out-degree bound and their delay bound is too rough to estimate exactly the number of participating peers. G. Bianchi *et al.*[15] studied the theoretical performance limits of streaming delay in chunk-based multi tree-based P2P streaming systems under peer out-degree bound and upload capacity constraint. They derived $n$-step Fibonacci sum representing the number of peers that may receive each chunk in given a time interval. F. Huang *et al.*[16] defined the minimum-delay P2P streaming problem(MDPS) and showed that it is NP-complete. They focused on building minimum delay mesh overlay streaming and proposed a centralized and distributed approximation algorithms based on

clustering and filtering. J. W. Jiang *et al.*[17] investigated the minimum worst user delay and maximum streaming rate in the region of homogeneity based on [15]. However, they did not consider the upload capacity utilization and the situation with heterogeneous peer upload capacities. T. Bonald *et al.*[20] investigated the rate/delay performance trade-off of several push-based protocols and proposed epidemic-style algorithms to achieve joint rate and delay optimality in static P2P streaming systems. However, previous works did not consider the peer upload capacity utilization with minimum delay guarantee under out-degree bound.

In this paper, we define the upload capacity utilization as well as the pure leaf peer to measure the upload capacity sharing. In addition, we describe the overlay construction algorithm which guarantees the minimum streaming delay and maximizes the upload capacity utilization based on the result of G. Bianchi *et al.*[15].

## 3 System Model and Problem Formulation

In this section, we present a mathematical model and formulations of the maximum upload capacity utilization problem with minimum streaming delay.

### 3.1 System Model

We assume that a P2P streaming system consists of a single streaming server, which generates chunks of a continuous stream at constant rate $r_s$, and $N$ peers. The system is composed of $L$ overlay networks logically and an overlay network has $N$ peers. In the multi tree-based P2P streaming systems, the streaming server divides whole stream data evenly into $L$ sub-streams and distributes each sub-stream through one of $L$ overlay networks with transmission rate $r_s/L$. In order for peers to playback the stream smoothly, they should participate in all overlay networks. We suppose that the peer out-degree of all peers is bounded, since the resources of peer are limited. The peer out-degree of all peers is bounded into $M$ equally.

We model a P2P streaming overlay network as a directed graph $G = (V, E)$, where $V$ is the set of participating peers and $E$ is the set of overlay edges $e = (i, j)$ which means that peer $i$ forwards the received chunk to peer $j$. We define the upload capacity of a server and a peer $i$ as $u_s$ and $u_i (i \in \{1, .., N\})$ respectively. We assume the peer upload capacity is only bottlenecks in the overlay networks according to measurement of studies of existing P2P streaming systems [14]. In the chunk-based P2P streaming systems, a peer can not distribute a chunk until it completes the chunk reception. Hence, we assume P2P streaming system runs in discrete time slots and the playback of a chunk in this system requires a time slot. The upload capacities of a peer $i$ and a server are chunks per time slot respectively, i.e., $u_s$ and $u_i$, are greater than or equal to $r_s$ a chunk per time slot. We suppose that $u_i$ is $k{\cdot}r_s, (k = 1, 2, ..)$. The key notations, used in this paper, are shown in Table 1.

**Table 1.** Notations in the System model

| Notation | Description |
|---|---|
| $N$ | number of peers in the P2P streaming system |
| $L$ | number of overlay networks in the P2P streaming system |
| $M$ | peer out-degree in the P2P streaming system |
| $r_s$ | streaming rate ($r_s$ bytes per single time slot) |
| $u_s, u_i$ | upload capacity of server and peer $i$ (per single time slot) |
| $p_i$ | binary value which denotes if peer $i$ is a *pure leaf* |
| $S_M(n)$ | maximum number of peers that receive a chunk newly at time slot $n$ with peer out-degree $M$ |
| $N_M(n)$ | maximum number of peers which may receive a chunk in a time interval $n$ with peer out-degree $M$ |
| $I_M(n)$ | minimum number of interior peers in a time interval $n$ with peer out-degree $M$ |
| $D_M(k)$ | minimum number of time slots it takes for at least $k$ peers to receive a single chunk from its generation with peer out-degree $M$ |
| $U_M(N)$ | upload capacity utilization function for peer out-degree $M$ and $N$ peers |

To formulate upload capacity utilization problem, we define *pure leaf*. The *pure leaf* is the leaf peer on all overlay networks. We introduce the binary value $p_i$ which denotes if peer $i$ is a *pure leaf* or not.

$$p_i = \begin{cases} 0, \text{ if peer } i \text{ is an interior peer in any overlay network.} \\ 1, \text{ if peer } i \text{ is a } \textit{pure leaf}. \end{cases}$$

To measure upload capacity utilization, we define upload capacity utilization function $U_M(N)$ for the $M$ peer out-degree and $N$ peers as follows:

$$U_M(N) = \sum_i (1 - p_i)/N, \quad i \in V. \tag{1}$$

In this paper, we formulate the maximum upload capacity utilization problem in the following way. This problem is to construct multiple overlay networks which maximize upload capacity utilization with minimum streaming delay.

$$\max \{U_M(N)\} \tag{2}$$

subject to

$$\begin{aligned} &u_s = r_s, \quad u_i = k{\cdot}r_s, k = 1, 2, .. \ , i \in V. \\ &D_M(N) = min\{t | S_M(t) = N\}. \end{aligned} \tag{3}$$

In order to maximize the upload capacity utilization in the multi tree-based P2P streaming systems, a leaf peer in any overlay network should be placed to have different role in other overlay networks. At first, we investigate the number of leaf peers in a $M$-ary complete tree without considering the upload capacity. Lemma 1 gives the number of leaf peers in a $M$-ary complete tree with $N$ peers.

**Lemma 1.** *Consider a $M$-ary complete tree with $N$ peers. The number of leaf peers $N_l$ is given by* $N - \frac{M^h-1}{M-1} + \lfloor \frac{M^{h+1}-N(M-1)-1)}{M(M-1)} \rfloor$
*where* $h = \lceil \log_M (N(M-1)+1) - 1 \rceil$.

*Proof.* Let the height of tree be $h$. Then the number of peers, $N$, satisfies these inequalities, $\sum_{i=0}^{h-1} M^i < N \leq \sum_{i=0}^{h} M^i$. Therefore, the number of leaf peers at the height $h$ is the difference between the number of all peers in the tree and the number of peers at the height $h-1$. That is, the number of leaf peers at height $h$ is $N - \frac{M^h-1}{M-1}$. Since the tree is not a saturated tree but a complete tree, there may be leaf peers at the height $h-1$ and $h$. The number of leaf peers at the height $h-1$ can be derived from the number of leaf peers at the height $h$ and the maximum number of peers at the height $h$. The number of leaf peers at the height $h-1$ is $\lfloor (M^h - (N - \frac{M^h-1}{M-1}))/M \rfloor = \lfloor \frac{M^{h+1}-N(M-1)-1)}{M(M-1)} \rfloor$. Consequently, the number of leaf peers $N_l$ is $N - \frac{M^h-1}{M-1} + \lfloor \frac{M^{h+1}-N(M-1)-1)}{M(M-1)} \rfloor$.

From Lemma 1, we derive the number of interior peers in the $M$-ary complete tree with $N$ peers in an overlay network. The number of interior peers in an overlay network is $\frac{M^h-1}{M-1} - \lfloor \frac{M^{h+1}-N(M-1)-1)}{M(M-1)} \rfloor$. The number of interior peers in all overlay networks is $M \cdot \{ \frac{M^h-1}{M-1} - \lfloor \frac{M^{h+1}-N(M-1)-1)}{M(M-1)} \rfloor \}$.

### 3.2 Streaming Delay

To analyze the streaming delay, we consider the delay required to deliver a single chunk to all peers in a overlay network. We assume that the chunk size is $r_s$ and the upload capacity of peer $i$, $u_i$, is $r_s$ to investigate the minimum delay. Therefore, all peers can push only a chunk to neighbors in a time slot. When a peer receives more than one chunk per time slot, the fan-out delay of the peer is longer than one time slot, since the peer can upload only a chunk per time slot. We define $N_M(n)$ as the maximum number of peers which may receive a chunk in a time interval $n$ after its generation at the source with peer out-degree $M$ and $S_M(n)$ as the maximum number of peers which receive a chunk newly at time slot $n$. As discussed in [15], $S_M(n)$ is expressed as the $M$-step Fibonacci sequence as follows:

$$S_M(n) = \begin{cases} 0 & \text{if } n \leq 0 \\ 1 & \text{if } n = 1 \\ \sum_{i=1}^{M} S_M(n-i) & \text{if } n > 1. \end{cases} \tag{4}$$

Fig. 1 shows the maximum number of leaf peers disseminated within given time slot $n$ and we can infer that the larger $M$ is, the shorter dissemination delay is obtained. However, the required buffer on each peer increases when the $M$ increases. There are few differences in the required time slot to disseminate a chunk to all peers, when the $M$ is greater than 7.

We derive the maximum number of peers which may receive a chunk in a time interval $n$ with peer out-degree $M$, $N_M(n)$, as depicted in Eq. 5.

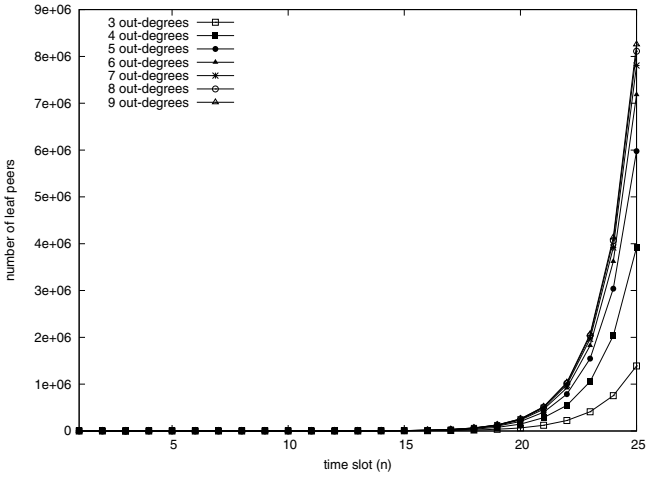



**Fig. 1.** maximum number of leaf peers under out-degree bound

$$N_M(n) = \begin{cases} 0 & \text{if } n \leq 0 \\ \sum_{i=1}^{n} S_M(n) & \text{if } n > 0. \end{cases} \tag{5}$$

From Eq. 4 and Eq. 5, the minimum number of interior peers within given time slot $n$, $I_M(n)$, is obtained as shown in Eq. 6.

$$I_M(n) = \begin{cases} 0 & \text{if } n \leq 1 \\ N_M(n-1) & \text{if } n > 1. \end{cases} \tag{6}$$

In order to analyze the maximum number of interior peers in Lemma 2, we adopt the delay function $D_M(k)$, discussed in [14,15], as minimum number of time slots it takes for at least $k$ peers to receive a specific chunk. Therefore, the $D_M(k)$ is derived from $N_M(n)$ as follows:

$$D_M(k) = \min\{n | N_M(n) = k\}, \quad \text{for } 1 \leq k \leq N.$$

**Lemma 2.** *Consider a P2P streaming system under peer out-degree $M$, consisting of $N$ peers, in which the upload capacity of all peer is $r_s$ per time slot. The maximum number of interior peers $N_i$ with minimum streaming delay $D_M(N)$ are as follows:*

$$I_M(N) = \begin{cases} N_M(D_M(N)-1) - S_M(D_M(N)-1) \\ \quad \textit{if } N - N_M(D_M(N)-1) \leq S_M(D_M(N)) - S_M(D_M(N)-1). \\ N - S_M(D_M(N)) \quad \textit{otherwise.} \end{cases}$$

*Proof.* When $N$ is $N_M(D_M(N))$, the tree has maximum number of peers in a time interval $D_M(N)$. Therefore, the number of leaf peers in this tree is $S_M(D_M(N))$ and the number of interior peers in this tree is $N - S_M(D_M(N))$.

When $N_M(D_M(N)-1) < N < N_M(D_M(N))$, $S_M(D_M(N)) - S_M(D_M(N)-1)$ is the number of interior peers which can have child peers as leaf peers within

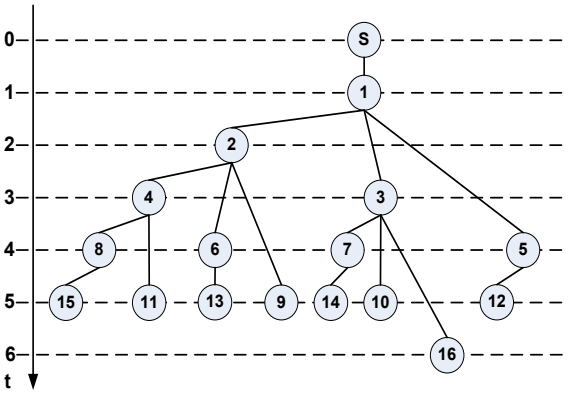



**Fig. 2.** The chunk distribution over an overlay network ($M$=3, $N$=16)

time slot $D_M(N)$. If $(N-N_M(D_M(N)-1)) \leq (S_M(D_M(N))-S_M(D_M(N)-1))$, the number of leaf peers newly added is $S_M(D_M(N)-1)+N-N_M(D_M(N)-1)$. Consequently, the number of interior peers is $N_M(D_M(N)-1)-S_M(D_M(N)-1)$.

When $S_M(D_M(N)) - S_M(D_M(N)-1) < N - N_M(D_M(N)-1)$, the peers more than $S_M(D_M(N)-1) + N - N_M(D_M(N)-1)$ are leaf peers added at time slot $D_M(N)-1$. Therefore, when $S_M(D_M(N)) - S_M(D_M(N)-1) < N - N_M(D_M(N)-1)$, the number of leaf peers is $S_M(D_M(N))$ and the number of interior peers is $N - S_M(D_M(N))$.

In order to investigate the overlay topology which gives minimum streaming delay, when $u_i = r_s$, we derive the overlay network as shown in Fig. 2, when $M$=3 and $N$=16.

As shown in Fig. 2, the server uploads a chunk to peer 1 within the time slot 1. The peer 1 uploads the received chunk to peer 2 within the time slot 2. Both peer 1 and 2 upload the same chunk to peer 3 and 4 in the time slot 3, respectively. And then, peer 1, 2, 3, and 4 upload the same chunk to peer 5, 6, 7 and 8 respectively. At the time slot 4, the number peers which receive the chunk is 8. At the next time slot, seven peers can distribute the chunk to other peers, since peer 1 has already supported three peers. After then, peer 2 can not support new peer because of the same reason. In this case, the streaming delay $D_3(16)$ is 6. In addition, the ratio of interior peers to all peers for various peer out-degrees is depicted as follows: Though a minimum number of interior peers in an overlay network is more than a maximum number of leaf peers, the upload capacity utilization in P2P streaming system is less than 1, since the peer may support only a peer in a time slot.

| $M$ | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|
| $ratio(\%)$ | 54.36 | 51.87 | 50.86 | 50.41 | 50.20 | 50.10 | 50.05 |

### 3.3 Continuous Streaming

In this section, we investigate the number of overlay networks to distribute all chunks of a stream to all peers smoothly and how to construct multi tree-based to maximize upload capacity utilization with minimum average streaming delay.

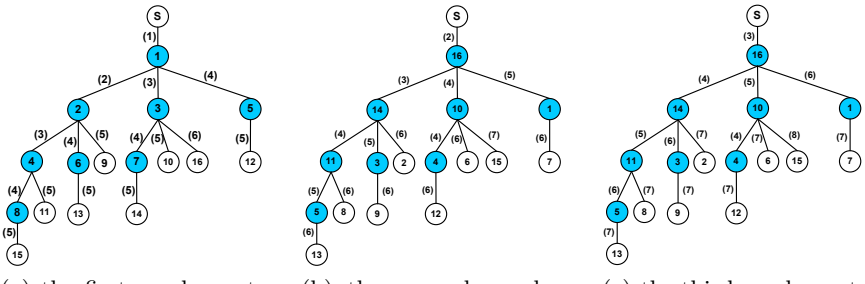
**Lemma 3.** *The minimum number of overlay networks to obtain minimum streaming delay in the multi-tree P2P streaming system in which the upload capacity of all peers is $r_s$ per a time slot and peer out-degree is $M$ is*

$$\min\{L\} = M. \tag{7}$$

*Proof.* Since each node has $M$ out-degree, the total upload capacity of all overlay networks is $N{\cdot}M{\cdot}r_s$. In addition, the required bandwidth to play a stream in all peers is $N{\cdot}r_s$. Therefore, the minimum number of overlay networks is $N{\cdot}M{\cdot}r_s/N{\cdot}r_s{=}M$.

Let us investigate how to construct multi tree-based P2P streaming system to maximize upload capacity utilization and what the upload capacity utilization of this system is. For consistency of analysis, we consider the case when $L = 3$, $M{=}3$, and $N{=}16$ to describe the overlay networks for the multi tree-based P2P streaming system with the maximum upload capacity utilization. At first, based on the chunk distribution example shown in Fig. 2, we construct the first overlay networks as shown in Fig. 3(a). Then, we construct two overlay networks based on the first overlay network by placing the leaf peers in the first overlay network onto interior peers in the other overlay networks, in order to maximize upload capacity utilization as shown in Fig. 3(b) and 3(c).

In Fig. 3(a), there are eight leaf peers and eight interior peers. The numbers in parentheses indicate the time slot within which the parent peer forwards the received chunk in the previous time slot to the specific peer. In the first overlay network, peer 1 forwards the chunk to peer 2, peer 3 and peer 5 within the time slot 2, 3, and 4 respectively. It also receives new chunk from server within the



(a) the first overlay network. (b) the second overlay network. (c) the third overlay network.

**Fig. 3.** the example of the overlay network construction

time slot 4. Within the time slot 5 and 6, seven leaf peers(peer 9, 10, 11, 12, 13, 14, and 15) receive the chunk and one leaf peer(peer 16) receives the chunk respectively. Therefore, on the other overlay networks, the seven leaf peers of the first overlay network can not receive a chunk within time slot 5, generally time slot $3k$+2 ($k$=0,1,2,...), the one leaf peers can not receive a chunk within time slot 6, generally time slot $3k$ ($k$=1,2,...). In the second overlay network, only one leaf peer of the first overlay network may receive a chunk within time slot $3k$+2 ($k$=0,1,2,...) and three leaf peers of the first overlay network should be in time slot $3k$ ($k$=1,2,..) and $3k$+1($k$=0,1,2,..). In the third overlay network, each of the two leaf peers of the first overlay network may be placed in time slot $3k$($k$=1,2,,...) and $3k$+1($k$=0,1,2,..) respectively. Consequently, the upload capacity utilization is 14/16 (0.875). For $M$=4, $N$=16, the upload capacity utilization is 15/16(0.9375) and the upload capacity utilization is also 15/16(0.9375) for $M$=5, $N$=16. From this analysis, we can conclude the upload capacity utilization in P2P streaming system with $u_i = u_s = r_s$ per time slot is less than 1. The initial playback delays experienced in peers are as follows:

| peer # | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| playback delay | 5 | 5 | 4 | 5 | 5 | 6 | 5 | 5 | 5 | 5 | 5 | 5 | 6 | 6 |

**Algorithm 1.** Overlay tree Construction Algorithm

**Input:** $M, N$, **Output:** $L$ **overlay trees**

*Compute the minimum delay bound* $D_M$
$L \leftarrow M$
**for** $i = 1$ to $L$ **do** ▷ Construct $L$ overlay trees repeatedly
  **if** $i$ is 1 **then** ▷ for the first overlay tree
    **for** $k = 1$ to $N$ **do**
      add *peer* $k$ to a tree $i$.
    **end for**
  **else** ▷ for other overlay trees
    **for** $k = N$ to 1 **do** ▷ Select a peer among leaf nodes
      check whether selected peer conflicts with other peer in the time slot.
      **if** conflicted **then**
        find a peer on the tree $i$ to replace.
        **if** not found **then**
          find a peer on the previous constructed trees to replace.
        **end if**
      **else**
        add *peer* $k$ to a tree $i$
      **end if**
    **end for**
  **end if**
**end for**

From the above table, the average initial playback delay is 5.0625 time slot. The deviation of the initial playback delay is 0.574 time slot and is lower than 1 time slot. The initial playback delay in multi tree-based P2P streaming system is nearby the streaming delay 6 time slot, since the peer, which is an interior peer in an overlay network, may be a leaf peer in other overlay networks.

The construction algorithm of overlay networks with a maximum upload capacity utilization is shown as followings. In the first step, construct the first overlay network. In the second step, create a set of peers, which are leaf peers in the first overlay network. In the third step, place the peers in the set created in the second step in the internal peers of other overlay networks without overlapping the time slot assigned in the previous overlay networks.

In order to construct overlay networks when the upload capacity of peers is different from each other, we will extend the previous result. In this analysis, we assume $u_i \geq r_s$ and $u_i = k \cdot r_s, k = 1, 2, ...$ In this case, we can construct overlay networks of a multi tree-based P2P streaming system with several sub overlay networks consisting of one root peer, which has the upload capacity larger than $r_s$ and several peers which have the upload capacity of $r_s$.

## 4 Conclusion

Our works are motivated by maximizing peer upload capacity utilization and minimizing streaming delay and initial playback delay in multi tree-based P2P steaming systems under peer out-degree bound. We investigate the peer upload capacity utilization in multi tree-based P2P streaming systems under peer out-degree bound and the effect of peer out-degree bound on maximum peer upload capacity utilization. In our analysis, we defined *pure leaf* and peer upload capacity utilization function. In addition, we described the algorithm to construct the pair of overlay networks which gives maximum upload capacity utilization. Particularly, we expanded the multi tree-based P2P streaming construction algorithm to construct the overlay networks in the multi tree-based P2P streaming system with heterogeneous upload capacities.

**Acknowledgement.** This research was supported by Basic Science Research Program through the National Research Found(NRF) funded by bye the Ministry of Education, Science and Technology(NRF-2010-0025069).

## References

1. Coolstremaing, http://www.coolstreaming.com
2. PPLive, http://www.pplive.com/
3. PPStream, http://www.ppstream.com/
4. Anysee, http://www.anysee.com/
5. Hei, X., Liu, Y., Ross, K.: Inferring Network-Wide Quality in P2P Live Streaming Systems. IEEE JSAC (December 2007)

6. Jannotti, J., et al.: Overcast: Reliable multicast with an overlay network. In: Proc. of Operation Systems Design and Implementation (2000)
7. Castro, M., et al.: SplitStream: High bandwidth multicast in cooperative environments. In: Proc. of ACM SOSP (2003)
8. Padmanabhan, V.N., Wang, H.J., Chou, P.A.: Resilient Peer-to-Peer Streaming. In: Proc. of IEEE International Conference on Network Protocols, ICNP (2003)
9. Zhang, X., et al.: DONet/CoolStreaming: A data-driven overlay network for live media streaming. In: Proceeding of IEEE INFOCOM (2005)
10. Magharei, N., Rejaie, R.: PRIME: Peer-to-Peer Receiver-deriven mesh-based streaming. In: Proc. of IEEE INFOCOM (2007)
11. Tang, Y., Luo, J.-G., Zhang, Q., Zhang, M., Yang, S.-Q.: Deploying P2P Networks for Large-Scale Live Video-Streaming Service. IEEE Communication Magazine (June 2007)
12. Liu, S., et al.: P2P Streaming Capacity under Node Degree Bound. In: Proc. of IEEE INFOCOM (2010)
13. Liu, Y.: On the Minimum Delay Peer-to-Peer Video Streaming: How Realtime Can It Be? In: Proc. of ACM Multimedia (2007)
14. Feng, C., Li, B., Li, B.: Understanding the Performance Gap between Pull-based Mesh Streaming Protocols and Fundamental Limits. In: Proc. of IEEE INFOCOM (2009)
15. Bianchi, G., et al.: Fundamental delay bounds in peer-to-peer chunk-based real-time streaming systems. CoRR, vol. abs/090 (2009)
16. Huang, F., et al.: An Approximation algorithm for minimum-delay peer-to-peer streaming. In: Proc. of Peer-to-Peer Computing (2009)
17. Jiang, J.W., Zhang, S., Chen, M., Chiang, M.: Minimizing Streaming Delay in Homogenous Peer-to-Peer Networks. In: IEEE International Symposium on Information Theory (2010)
18. Liu, S., et al.: Performance Bounds for Peer-Assisted Live Streaming. In: Proc. of ACM SIGMETRICS (2008)
19. Kumar, R., Liu, Y., Ross, K.W.: Stochastic Fluid Theory for P2P Steaming Systems. In: Proc. of IEEE INFOCOM (2007)
20. Bonald, T., Massoulie, L., Mathiue, F., Perino, D., Twigg, A.: Epidemic Live Streaming: Optimal Performance Trade-Offs. In: Proc. of ACM SIGMETRICS (2008)

# Content Localization for Non-overlay Content-Aware Networks

Piotr Pecka, Mateusz Nowak, and Sławomir Nowak

Institute of Theoretical and Applied Informatics
ul Bałtycka 5, 44-100 Gliwice, Poland
{piotr,mateusz,emanuel}@iitis.pl

**Abstract.** Content addressing and localization are the basic issues in design of content-aware networks (CAN). The paper summarizes assumptions for content-aware networks and presents content localization algorithm for PI-CAN, developed within FIE (Future Internet Engineering) project. The algorithm allows to find location of desired content in distributed content database in time O(1).

**Keywords:** content-centric networks, content aware networks, content localization.

## 1 Introduction

The TCP/IP principles were created in early '70s. Its communication model is connection and data exchange between exactly two particular devices. At the beginning the network was used for data exchange among small number of devices, mainly in "professional" field – military, supercomputing, databases etc. The basis of its operation was a connection between two particular devices, like terminal and mainframe computer or two mail exchangers. This resulted in placing into IP packets, exactly two addresses – an address of sender and an address of receiver of the packet. Still almost all Internet traffic consists of conversations between two particular hosts.

In the meantime Internet has changed fundamentally. It became affordable for individuals, and network-connected devices are ubiquitous nowadays. One of main fields of Internet usage is entertainment, requiring access to huge amount of multimedia data. Users need not to know *where* the data are, they only want to know *what* content is available for them.

The huge increase of media content available in the Internet led to the concept of CCN *(Content-Centric Networks),* known also as CAN *(Content-Addressable Networks* or *Content-Aware Networks)*[7,12]. The networks are based on new paradigm of *"content-centric networks"*, being opposite to current *"host-centric networks"* based on TCP/IP . Using a network, which is content-centric, the user requests particular content (identified in unambiguous way) without taking care of its location. It's the network layer which ensures localization and delivery of the content to the user. This approach is entirely different from today's TCP/IP based networking, where the user is obliged for identification the piece of content with address of server and its file name and path, e.g. in the form of URL. In case the content is moved from

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 520–528, 2011.

host to host, new location becomes unknown to the user. Meanwhile CAN, having knowledge about content that can be delivered, is able to locate desired content for the user. Moreover, in case more copies of the content object exist, CAN is able to deliver the content from the server assuring required transmission parameters, resulting best multimedia quality for the user. Thus, the user can see CAN as "black-box", being expected to simply deliver desired content. User does not point at host storing data to deliver any more. In the CAN networks an identifier plays the role of address. To unambiguously point at the content only the content identifier is needed.

The paper is devoted to the problems of identification and localization of the content in CAN. Although the problem of content delivery is mentioned, we do not focus on it here. We describe the architecture and basics of operation of CAN in Section 2, considering especially PI-CAN solution in FIE project[1]. Sections 3 and 4 are devoted to problems of content localization and naming scheme. Our proposal of localization algorithms as well as its evaluation is presented in section 5.

## 2 Basic Functional and Architecture Issues of CAN Networks

Content-centric approach exists already in popular peer-to-peer (p2p) networks. However, the networks have overlay architecture – they are built on top of current TCP/IP protocol stack. Therefore, it is hard or impossible for them to take advantages of lower layers architecture. E.g. neighborhood in some p2p networks is defined in completely different way than in IP ([10,13]). Another issue typical for p2p networks is short life-time of a network node. As network client works in application layer, it is typically run at user's workstation. Usually the workstation is switched off after a few hours of work, so the configuration of the network changes relatively rapid, comparing to Internet infrastructure, where routers or servers (e.g. WWW or FTP) have the uptime of tens or hundreds of days.

Content-centric networks can be designed with different presumptions. Jacobson's approach [7] is evolutionary and use existing TCP/IP stack. Such a design expands current overlay networks solutions. On the other side, NetInf project [3] shows revolutionary approach, proposing full new stack of protocols.

In CAN design a few types of nodes can be distinguished. According Subbiah and Uzmi [12] the roles of CAN nodes are: *Content Provider*, that owns and shares the content, *Content Consumer*, who requests the content and receives it, *Content Service Provider*, who provides search and location services and *Content Broker* that deliver the data.

This work is related to the FIE project, focused on virtual *Parallel Internets (PI)* – concept which joins different protocols using virtual network links, in single physical connection [1]. One of the PI, being designed in the project, is PI-CAN. For PI-CAN revolutionary approach is taken, and complete stack of CAN protocols is being implemented. The network shall provide CCN functionality with basic QoS guarantees, and will be built from scratch, as all layers above PHY will be designed [2]. The node names used in the project are: *Content Consumer*, *Content Server* (which stores the data), *Content Mediator* (provides search and location functions) and *Content Forwarder*, which is CAN router. The content can be replicated in more

[1] Future Internet Engineering (Polish *Inzynieria Internetu Przyszłości*) project – `http://www.iip.net.pl`

than one Content Server. Content Consumer is connected to Content Mediator, which receives user's requests for delivering the content, and is called below border Content Mediator (Content Mediators serving only for content resolution, with no users connected to them also exist in CAN). Thus Content Mediators are responsible for localisation all copies of the content and for choice of best one for delivery. The user's request contains the content identifier (ContentId), which is unique and unambiguous. The content is also associated with additional information, referred as meta-data, which are related to its matter (such as full name, author, publisher, performers) and to technical characteristic (like length, size, codec used, bandwidth required etc.). These data are stored by Content Mediators in Content Records, containing also content locations.

## 3 Solutions of Content Localization Problem

As the user, requesting the content, provides only its identifier, the CAN is responsible for localization of the content, and then for delivering it to the user.

The solutions of localization problem, used in contemporary networks, divide into two parts. First group extends the idea of the DNS system (which itself serves for host localization). Second group of localization algorithms falls in the group called Distributed Hash Tables (DHT). Some of DHT algorithms are used in application-level (overlay) networks, like BitTorrent [6] or Skype [4].

Localization algorithms differ in characteristics. Content identifier format depends on localization method, as well as network nodes requirements. Localization efficiency and quality is measured by so called *request hops*, as every request is directed to nearest Service Node, which passes it (hops) to next nodes.

### 3.1 Hierarchical Localization Systems

Most natural way of content localization seems to be extension of DNS idea, which serves for host localization in the Internet. Such an approach is shown in [7], where content identifiers are DNS based. Similar idea stands behind Handle System [15]. Hierarchical systems features:

- long, human-friendly names,
- hierarchical lookup algorithm,
- high requirements for top-level serves,
- number of requests hops depends on depth of hierarchy.

### 3.2 DHT Algorithms

Alternative for hierarchical localization systems are DHT solutions. They are being used in p2p systems as well as in experimental CCN or CAN networks. A number of CAN-type projects based on DHT exists. ROFL [5] uses localization method based on classical Chord DHT algorithm [13]. SEATTLE [8] provides a directory service using flat addressing with a one-hop DHT. Another projects using DHT for content localization are i3 [14] and 4WARD NetInf [3].

DHT methods:

- require unstructured identifier, so no human-friendly names are possible,
- typically, constant-length identifiers are used,
- all hosts in the network are equivalent, as no hierarchy exists,
- number of request hops depends on size of the network, as (in popular DHT methods) it does not exceed O(log n), where n – number of nodes in the network, and in some cases can be even O(1).

## 4 Identification Scheme and Resolution Algorithms

The choice of localization method is strictly connected with chosen identification scheme. The ContentId must fulfill following requirements:

- must be unique within the network,
- must be long enough to allow addressing of all content in the network,
- should be as short as possible to minimize the overhead,
- must support the content localization algorithms used in the network.

The open issues which have to be answered during the CAN architecture design are, whether the identifier:

- has any internal structure or is completely flat (and what the internal structure shall be),
- has constant or variable length,
- is human-readable or is not,
- is assigned locally or globally, and how it influences on the identifier structure (e.g. prefix of an entity granting the identifier).

The choices of localization algorithm and identification scheme depend of each other. The extended DNS solution require structured (hierarchical) names, while DHT-family algorithms will work with unstructured identifiers.

## 5 Proposed Solution for Content Localization

As the authors show in [11], DHT algorithm along with unstructured, 128-bit identifiers looks like optimal trade off for PI-CAN network. We assumed following objectives for PI-CAN's algorithm of content resolution:

- *Decentralization* – the nodes collectively form the system without any central coordination.
- *Scalability* – the system should function efficiently even with thousands or millions of nodes.
- *Fault tolerance* - the system should be reliable even with nodes continuously joining the network, leaving, and failing.
- *Performance* – number of hops and number of control or data messages between them should be minimal

On the other side we assumed the Content Mediators to be elements of network infrastructure, what results in having lifetime in the order of weeks or months rather than hours (like routers in IP networks, and unlike workstations being nodes in p2p networks). This requirement allows us to expect that every change in Content Mediator's presence in the network will be possible to propagate over all nodes in PI-CAN, as changes will be seldom, and traffic overhead related to these updates will not cramp the transport of content

### 5.1 COLOCAN Algorithm

We propose here the method for content resolution (localization) for PI-Content Aware Network – COLOCAN (COntent LOcalisation for CAN). The algorithm is based on the Distributed Hash Tables general idea, however it is not based on any existing solutions like Chord or Kademlia. The network of Content Mediators acts like distributed database of information about content, its descriptions and locations. ContentId is the search key in distributed content database, and Content Record (containing i.e. addresses of servers storing copies of desired content) is search result. As network user directs his request to border Content Mediator, and Content Records are also stored on Content Mediators, the whole algorithm is performed by Content Mediators. CM acronym is used for Content Mediator below.

Every piece of content, which should be available in PI-CAN, requires registration. By registration we mean putting Content Record into proper CM, and ipso facto into distributed localization database (this approach is similar as in DONA[9] and some other projects, and different than e.g. in Jacobson's CCN, where no registration was required). As mentioned above, 128-bit unstructured key is used as ContentId for determining the CMs where the piece of content should be registered and – later on – where the Content Record is being retrieved from. The algorithm is redundant, securing basic level of fail-safety, as described in section 5.2. Main idea of localization algorithm is described in section 5.3. Behavior of the system in case of changes in CMs network, particularly when adding, removal or failure of CM is described in section 5.4

### 5.2 Redundancy

For the fail-safety reasons we decided to register every piece of content on K different CMs. This assures that in case of failure of some CMs a copy of particular Content Record will still be available. We use 128-bit random number generator to obtain K content-keys associated with piece of content having given id, which is used as generator seed. The same content-key generation procedure is used while registering content and while localizing it, so the same keys are generated in both cases. For the purposes of experimental implementation K=3 was assumed.

### 5.3 Localization

The basic data structure for the COLOCAN algorithm is binary tree, stored in every CM's memory, called CM-tree. The tree stores network addresses of CMs, and CM's Id (CMID) is determined by its position within the tree (see fig. 1). After receiving the request related to given ContentId (resolution or registration) border CM has to find

out, which CM stores the Content Record bound to this id, or which CM should store it (in case of registration). Using the procedure described in section 5.2, CM generates K content-keys, which are subsequently used as search keys in the CM-tree. *k* lowest bits from content-key are taken for lookup the tree, and *k* is equal to maximum depth of the tree (if the tree was balanced, `k = ceiling(log N)`, where *N* – number of Content Mediators; balancing of the tree in not required for proper operation of COLOCAN, but CMs having shorter CMID have to store more Content Records and answer to more queries).

If address of CM bound to given key is found, request to deliver Content Record is sent to it. For higher reliability all K searches are performed at once and all delivery requests are send simultaneously.

In case the network configuration is well reflected at the node, and address information about given Content Mediator are up-to-date, the search finishes after receiving the Content Record from queried CM. In case of problems with connection with given CM we can still count on other *K-1* CMs to respond.

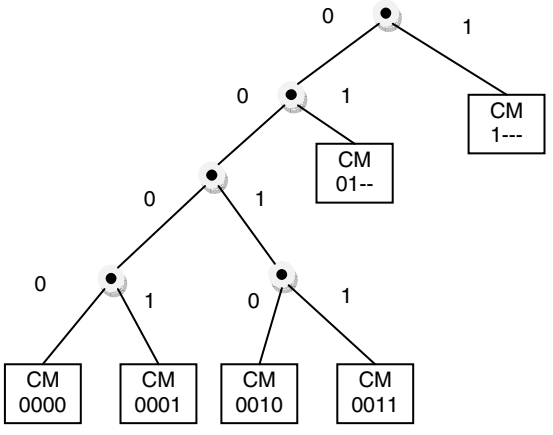



**Fig. 1.** COLOCAN binary tree of Content Mediators

### 5.4 Changes in Network Configuration

Changes in Content Mediators topology in the network can come out from following reasons:

- adding new Content Mediator to CAN,
- controlled removal of Content Mediator from CAN,
- permanent failure of Content Mediator.

In all cases network configuration stored in remaining Content Mediators has to be updated, but in the meantime between change in the network and information propagation we expect resolution mechanism to work appropriate. Let's consider all cases mentioned above.

**Addition**

In case of addition of new CM to the network, nearest operational CM must be found. Proper node in the CM-tree "splits" into two new leafs, adding one level to the tree, as shown at fig. 2. CMID of old node is extended by adding "0" at its end, new node receives the same CID, differing only with the last digit. The change in CM-tree is then propagated over the network, and simultaneously Content Records, having ContendId matching CMID of the new node are transferred from the old node to the new one.

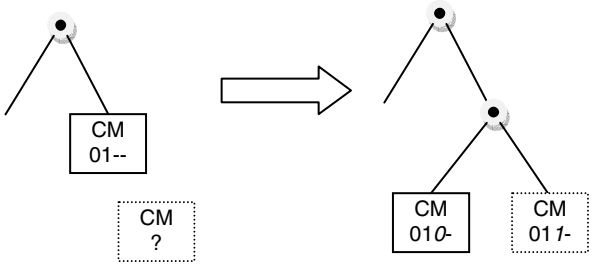



**Fig. 2.** New CM in the network

In the period when the address and CMID of new CM is not propagated, queries are directed to "old" CM – it must respond to them properly. Before all the content records belonging to "new" CM are migrated, no record is removed from "old" CM and the queries can be responded as if no change took place. After the migration is done, "old" CM removes all the migrated record from its storage and if it receives the bad-directed query, it responds with actualization packet, sending up-to-date information about new CM, its CMID and network address.

**Controlled removal**

In case the CM is switched off, having the time and possibility of migration of stored data, the process is the reversal of process of CM addition. Content Records must be transferred to neighboring (from the CM-tree perspective) CM, and its CMID must be shortened.

As the CM does not exist in the network any longer, CM trying to obtain Content Record from it will receive no answer. As neighboring CM, having formerly CMID differing only with the last digit is still alive, the request should be directed to it.

**Permanent failure**

As permanent failure we consider the situation the node is switched off unexpectedly and remains in this state longer than presumed time (e.g. 24 hours). Such a situation must be discovered and Content Records, which became unavailable must be recreated on Content Mediator which is proper for them in current configuration. In general it requires re-registration of the content by Content Publisher. Unfortunately, no way of notifying Content Publishers of the failure is provided. Temporary the content will still be available thanks to redundancy mechanism described above, but after longer period of time all K CMs storing given Content Record may fail. At the

current stage of development it is publisher's interest to periodically check, whether the Content Records are still available, and re-register them when needed. We consider also implementation of K-warnings mechanism. The warning is sent to the publisher by border Content Mediator, which requests given piece of content and receives less than K answers in assumed time. Publisher would re-register the content after a few warnings, without having to do periodic check.

## 6 Summary

The PI-CAN network is part of Future Internet Engineering project: http://www.iip.net.pl/en. The project covers the development and testing of infrastructure and services for the future generation of internet networks. The aim of this project is to develop and to test a new architecture as the replacement of current TCP/IP (especially IPv4). The project is implemented by a consortium composed of leading technical institutes in Poland and is sponsored by EU funds.

The algorithm presented in the paper is provided for the networks consisted of Content Mediator nodes having long live-time, used for resolution of the content. However, description of its operations is shortened, especially for the situations of controlled removal and failure of the node. The method is simple, but first evaluations show its effectiveness. In most cases it is able to find desired Content Record in one step (the request makes one hop – from border Content Mediator to Content Mediator storing the Content Record). Only in transitory state, when changes in the network structure are not yet mirrored in the border Content Mediator's memory, two hops will be needed. Thus the complexity of the algorithm is O(1) and number of hops is independent of size of the network. Admittedly its resistance to failures and changes in high scale (thousands of CMs) requires further research, mainly simulative, current solution looks very prospective.

**Acknowledgements.** The work presented in present paper is supported by Future Internet Engineering (Polish: *Inżynieria Internetu Przyszłości)* project, EU Funds 2007-2013, contract no. POIG.01.01.02-00-045/09-00.

## References

[1] Future Internet Architecture – state of the art and requirements – first version, Future Internet Engineering (IIP) report, EU Funds 2007-2013, contract no. POIG.01.01.02-00-045/09-00, http://www.iip.net.pl/

[2] Specification of Parallel Internet 3: Content Aware Network, Future Internet Engineering (IIP) report, EU Funds 2007-2013, contract no. POIG.01.01.02-00-045/09-00, http://www.iip.net.pl/

[3] D-6.2. Second NetInf architecture description. EU 7th Framework Programme, The Network of the Future Project 216041 4WARD - Architecture and Design for the Future Internet

[4] Baset, S.A., Schulzrinne, H.G.: An Analysis of the Skype Peer-to-Peer Internet Telephony Protocol. In: Proc. of INFOCOMM 2006 (2006)

[5] Caesar, M., Condie, T., Kannan, J., Lakshminarayanan, K., Stoica, I., Shenker, S.: ROFL: Routing on Flat Labels. In: SIGCOMM (2006)
[6] Cohen, B.: The BitTorrent Protocol Specification, http://bittorrent.org/beps/bep_0003.html
[7] Jacobson, V., Smetters, D.K., Thornton, J.D., Plass, M.F., Briggs, N.H., Braynard, R.L.: Networking named content. In: Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies, CoNEXT 2009 (2009)
[8] Kim, C., Caeser, M., Rexford, J.: Floodless in SEATTLE: A Scalable Ethernet Architecture for Large Enterprises. In: SIGCOMM (2008)
[9] Koponen, T., Chawla, M., Chun, B.-G., Ermolinskiy, A., Kim, K.H., Shenker, S., Stoica, I.: A Data-Oriented (and Beyond) Network Architecture. In: SIGCOMM (2007)
[10] Maymounkov, P., Mazieres, D.: Kademlia: A peer-to-peer information system based on the XOR metric. In: Druschel, P., Kaashoek, M.F., Rowstron, A. (eds.) IPTPS 2002. LNCS, vol. 2429, pp. 53–65. Springer, Heidelberg (2002)
[11] Nowak, M., Nowak, S., Pecka, P., Grochla, K.: Content identification in PI-CAN network (accepted for print in Springer. CCIS, vol. 160)
[12] Subbiah, B., Uzmi, Z.A.: Content aware networking in the internet: issues and challenges. In: IEEE Int. Conf. on Communications, ICC 2001, vol. 4, pp. 1310–1315. IEEE, Los Alamitos (2001)
[13] Stoica, I., Morris, R., Karger, D., Kaashoek, F., Balakrishnan, H.: Chord: A Scalable Peer-To-Peer Lookup Service for Internet Applications. In: SIGCOMM (2001)
[14] Stoica, I., Adkins, D., Zhuang, S., Shenker, S., Surana, S.: Internet Indirection Infrastructure. In: SIGCOMM (2002)
[15] Sun, S., Lannom, L., Boesch, B.: Handle System Overview. RFC 3650, IETF (November 2003), http://www.ietf.org/rfc/rfc3650.txt

# On Modelling of Fair Throughput Allocation in Overlay Multicast Networks

Michal Kucharzak and Krzysztof Walkowiak

Department of Systems and Computer Networks
Wroclaw University of Technology, Poland
{michal.kucharzak,krzysztof.walkowiak}@pwr.wroc.pl

**Abstract.** Overlay-based multicast has been proposed as an efficient solution to ensure multicast communication in the Internet. This paper focuses on modelling of various aspects of fairness in overlay multicast networks aimed at throughput maximization. Two distinct linear programs are proposed. The first formulation employs conceptual flows and provides optimal data rate allocation, whereas the second, relaxed version is derived from fractional Steiner trees. In addition, performance of two optimization tools (CPLEX and GUROBI) is compared in relation of time and result quality meanings. The paper reports that the fractional trees-based formulation provides better time efficiency while its solutions' quality might reach maximum obtainable data rates.

**Keywords:** Overlay multicast, simulated annealing, tree packing.

## 1 Introduction

Over the years, a lot of research and testing efforts have been devoted to support multicast communication. Initially, network-aware protocols implemented in the IP layer has been developed [5]. However, IP multicast is afflicted with many technical problems derived form scalability, addressing or congestion control and it is still far from being widely deployed. The main reasons for that include different system configurations depending on ISP and the lack of business model supporting inter ISPs IP-based multicast. To tackle some of the problems, overlay-based multicast has been proposed. It comprises systems which realize multicast routing by forming virtual topologies on the top of network and transportation layers. This virtual-oriented concept expands end-system multicast [4,15] and provides potential scalability and ease of deployment of new, network-layer-independent protocols at relatively low costs. Contrariwise to IP multicast, in overlay multicast data packets are replicated and forwarded by participating nodes instead of IP routers and logical multicast flows are realized as multiple unicast flows at the level of the network layer. Many various concepts towards implementation of overlay multicast have been already proposed [1,2,10,11,16,21,24]. Since the overlay multicast utilizes the same network link multiple times, it is less efficient than IP multicast in terms of network redundancy.

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 529–540, 2011.

Most of previous works on optimization of overlay multicast focus mainly on satisfying minimum flow costs or delays for guaranteeing streaming on given data rate (e.g. [9,17,19]). On the other hand, the objective of content distribution might also consider maximization of system's throughput. For example, in [20] the authors employ conceptual flows and formulate linear program with node capacity constraints for maximizing data rate allocation. In our paper, we develop an optimization model and apply constraints for fair distribution of data rates. Fairness is a general term, derived from ethical and justice theory, which was discussed in many works in various aspects [13,14,23], it is also widely adapted in communication network problems (see [3,12,22]). Moreover, we present a slightly different idea of throughput maximization that is based on predefined tree candidates selection.

The paper focuses on comparison of both linear approaches in the meaning of time efficiency and result quality. In order to obtain optimal results, CPLEX [7] and GUROBI [6] optimizers are used and their performance is also discussed. The main contribution of the paper includes new formulations of fairness applied to overlay multicast systems and quality-like comparison of two linear models, which express the throughput maximization problem.

## 2 Data Rate Allocation Problem with Fairness Constraints

We consider an overlay system and the problem of information multicasting, when every user participating in the system obtains the same amount of data from a single source. A general problem of data rate allocation in overlay multicast network is to realize as high throughput as possible.

The first linear formulation is derived from the optimal data rate allocation problem presented in [20]. It employs conceptual flows in order to realize consistent multicast streams. With regard to modelling an optimal data rate allocation in overlay multicast system, two basic sets of variables are introduced: $x$ representing data rates assigned to each overlay arc, and $f$ for defining conceptual flows in the system. Let $r$ to be data rate realized by the overlay system, thus the goal of the problem is to maximize $r$. Additionally, $\Psi$ models fairness coefficient so that every participant of the system behaves fair in relation to all users.

**Data Rate Allocation in Overlay Multicast System (DRA)**

**indices**

$i, j, k = 1, 2, ..., V$ end hosts (peers, overlay system nodes)

**constants**

| | |
|---|---|
| $s$ | source, root node ($s \in \{1, 2, ..., V\}$) |
| $u_i$ | upload capacity limit of node $i$ |
| $d_j$ | download capacity limit of node $j$ |
| $\Psi$ | fairness coefficient |

**variables**

$r$ data rate in the system (throughput)
$x_{ij}$ flow on overlay arc $(i, j)$, auxiliary variable
$f_{ijk}$ flow on overlay multicast link from $i$ to $j$ realizing throughput from the source to $k$ (auxiliary variable)

**objective**

$$\max \quad F_1 = r \tag{1}$$

**constraints**

$$\sum_j x_{ij} \leq u_i \quad i = 1, 2, ..., V \tag{2}$$

$$\sum_i x_{ij} \leq d_j \quad j = 1, 2, ..., V, j \neq s \tag{3}$$

$$f_{ijk} \leq x_{ij} \quad i, j, k = 1, 2, ..., V, j \neq i, j \neq s, k \neq s \tag{4}$$

$$\sum_{j \neq s} f_{sjk} = r \quad k = 1, 2, ..., V, k \neq s \tag{5a}$$

$$-\sum_{j \neq k} f_{jkk} = -r \quad k = 1, 2, ..., V, k \neq s \tag{5b}$$

$$\sum_{j \neq s,i} f_{ijk} - \sum_{j \neq i,k} f_{jik} = 0 \quad i, k = 1, 2, ..., V, k \neq s, k \neq i, i \neq s \tag{5c}$$

$$\Psi r \leq \sum_{j \neq i,s} x_{ij} \quad i = 1, 2, ..., V, i \neq s \tag{6}$$

The objective (1) maximizes data rate $r$ in the overlay system comprising $V$ nodes with a single source $s$ of the content. Equations (2) and (3) represent upload and download limits for each node, respectively. Both sets of constraints (2) and (3) refer to node capacity constraints, and every node $i$ can neither contribute to the system exceeding its upload capacity $u_i$ nor download more than its download limit $d_i$. Since we consider the system where source node $s$ only uploads the content, the download limit capacity of the node $s$ is not taken into consideration in the allocation problem. Constraints (4) connect variables $f$ and $x$. In case that an arc $(i, j)$ realizes flow from the source to node $k$, the flow $f_{ijk}$ cannot be greater than total throughput carried on that arc $(x_{ij})$. A set of equations (5) is refereed to as conceptual flow and describes flow balance at every node in the system. Constraints (5a) define that outflows from the source $s$ to each participating node $k$ are equal to $r$, which represents throughput obtained

in the system. On the other hand, every node $k$ is supposed to have inflows of the value of $r$, thereby (5b) express that the sum of inflows to $k$ from all nodes $j$ must equal to exactly $r$. Finally, the flow balance at each relaying node $i$ different than the source node $s$ and sink $k$ is required to be 0, i.e. the amount of inflows to $i$ must be equal to outflows from $i$ if the flow is directed from $s$ to $k$ and traverses $i$. The condition is defined by constraints (5c).

Note that, without loss of generality, we define the problem for the overlay system in which all nodes might be connected to each other using a direct overlay arc (complete directed graph), but for some systems the model can be specialized in order to formulate constraints for predefined set of virtual (overlay) arcs $A$. In order to model a graph which is partially connected, $\forall(i,j) \in A$ should be used instead of $\forall i$ and $\forall j$.

According to [3] and [22] the overlay structured peer-to-peer system should be fair in terms of the throughput served by the individual nodes. Consequently, users should not be forced to upload much more than they have downloaded or, on the other hand, node cannot be allowed to download more than it can contribute by uploading. The generic goal is to ensure capability of sharing fairly the available bandwidth and to utilize resources in more cooperative way. Fairness can be viewed as a kind of an incentive for nodes to participate, especially in situations where, e.g. ISPs charge based on uplink usage or uplink bandwidth is scarce. To enforce fairness in the overlay multicast system, we propose several various models which formulate different interpretations of fairness.

We start from a basic fairness interpretation described by fairness coefficient $\Psi$. Similar fairness relation concept was introduced in our previous work dedicated to flow optimization in file sharing overlay systems [18]. A set of equations (6) models fair behaviour of every node $i \neq s$, which downloads the content. It ensures that users are enforced to contribute to the multicast system by devoting their upload capacity not less than a $\Psi$-proportion of total system's throughput $r$. The important point to note about the above constraints is that to yield a solution to be positive (non-zero), $\sum_{j \neq i,s} x_{ij}$ must be greater than 0 for all $i$. And since condition (6) is satisfied with $\sum_t x_t > 0$, it is guaranteed that there are no free riders in the system.

Next, let us consider a overlay multicast system which permits a presence of free riders an avoids a zero-throughput solution in case even a single misbehaving user affects the system's performance. It is impossible to realize nonzero throughput in a fair system defined by (6) while at least one free rider is present because selfish nodes cannot be forced to upload any content. This leads the objective to be equal to zero. To avoid such a situation the fairness might be relatively relaxed. Let $\Psi^*$ to denote allowable throughput that can be downloaded by a node without any upload contribution to the multicast system.

Thus constraints (6) might be modified to

$$\Psi r \leq \sum_{j \neq i,s} x_{ij} + \Psi^* \quad i = 1, 2, ..., V, i \neq s \tag{7}$$

Both (6) and (7) consider fair bahaviour, which enforce nodes to contribute in content uploading, but on the other hand, fairness might refer to a situation when a node is ensured not to upload more than a given fairness coefficient $\overline{\Psi}$ in relation to maximum obtained throughput:

$$\overline{\Psi} r \geq \sum_{j \neq i,s} x_{ij} \quad i = 1, 2, ..., V, i \neq s \tag{8}$$

The set of constraints (8) helps active users in the system not to drain their networks resources by misbehaving users or free riders.

The main advantage of the abovemetioned linear formulation is that the optimal result packs maximum multicast throughput to an overlay system defined by a set of nodes with capacity constraints. However, it might also suffer from some disadvantages which reduce its practicality and applicability in real overlay multicast protocols. First of all, let us consider given optimal rate allocation in the overlay multicast system, the problem that has arisen is how to realize routing, i.e. how to divide the content and assign it to overlay arcs. Second, in order to implement additional constraints which can model levels of nodes in the multicast routing, delay based constraints or some reliability and survivability conditions may require highly extended formulation with many sets of additional variables and constraints. Last, complexity of the problem can lead to that computational time required for performing optimization and obtaining result reaches unacceptable values or even CPU's memory errors might occurs.

To tackle the problems that are caused due to the complexity or generality of the data rate allocation formulation presented in this subsection, we also develop a fractional spanning tree problem which is aimed at overlay multicast throughput maximization. This formulation provides a kind of tradeoff between complexity and applicability of the maximization of throughputs.

## 3 Fractional Spanning Tree Packing Problem with Fairness Constraints

We assume that multicast stream can be split into several separate substreams which might be illustrated as separate spanning trees including all overlay nodes and rooted in the same vertex. Such a concept provides advantages that load in the network is balanced and users' resources are utilized in more effective way. Based on natural relaxation of the Steiner tree packing problem proposed in [8] and known as fractional Steiner tree packing problem, we employ its special case applying features of overlay networks in order to maximize throughput in such systems. In contrary to Steiner tree packing and discussion covered in [8], we address the problem of overlay multicasting comprising spanning trees and access link capacities assigned to every overlay node instead of every link (edge) in the topology.

The basic formulation includes a predefined set of trees $t \in \{1, 2, ..., T\}$ and a corresponding set of integer variables $x_t$, which represent flow allocated to tree $t$. The following version of a problem comprises trees represented by vector $\beta$

which contains number of children of every node in each tree $t$. For example, $\beta_t = [3, 0, 2, 0, 0, 0]$ refers to a tree, in which a node 1 is a parent of three nodes, node 3 has two children and nodes 2, 4, 5 and 6 are leaves. The problem is to find a maximal throughput assignment to predefined trees regarding available capacities of participants' access links. Every tree $t$ represents a fractional multicast stream. Note that $\beta$ does not define a topology of every tree (actual arcs in the tree cannot be resolved) but such a tree representation is sufficient in order to formulate capacity and fairness constraints.

**Fractional Spanning Tree Packing Problem**

**indices (additional)**

$t$ $= 1, 2, ..., T$ predefined trees

**constants**

$\beta_{ti}$ number of $i$'s children in tree $t$; 0 if $i$ is a leaf

**variable**

$x_t$ throughput assigned to tree $t$ (e.g. streaming rate in kbps, integer)

**objective**

$$\text{maximize} \quad F = \sum_t x_t \tag{9}$$

**constraints**

$$\sum_t \beta_{ti} x_t \leq u_i \quad i = 1, 2, ..., V \tag{10}$$

$$\sum_t x_t \leq min\{min_{i \neq s}\{d_i\}, u_s\} \tag{11}$$

$$\Psi \sum_t x_t \leq \sum_t \beta_{ti} x_t \quad i = 1, 2, ..., V, i \neq s \tag{12}$$

The objective function $F$ (9) maximizes summarized throughput of all fractional multicast trees in the overlay system. Constraints (10) formulate upload limit based on available upload capacities $u_i$ of every node in the system. As regards node $i$ is a parent node in tree $t$, it uploads content of size $x_t$ exactly $\beta_{ti}$ times. Therefore, $i$'s total upload given by $\sum_t \beta_{ti} x_t$ cannot exceeds its upload limit $u_i$. By analogy to upload bound (10), a set of constraints (11) is introduced in order to guarantee a download limits of nodes are not exceeded. Basically, the download limit expresses that $\sum_t x_t$ cannot be greater than $d_i$, for every $v$. However, taking into account the overlay system, where all overlay nodes are connected to all trees, the actual throughput $\sum_t x_t$ is restricted not to exceed the minimal download limit $d_i$ among all overlay nodes ($min_{i \neq s}\{d_i\}$). Moreover, a source's upload limit affects the total throughput $\sum_t x_t$ must be lass or equal than $u_s$, i.e. all the content originates from root node $s$ and it cannot produce more throughput than its upload limit $u_s$. Finally, a minimum value among

$min_{i \neq s}\{d_i\}$ or $u_s$ limits overall throughput packed into the overlay system. In such a way a number of constraints that correspond to the overall throughput in the system is decreased to exactly one. The left-hand side of (12) denotes total throughput downloaded by every node (except of root) in the overlay multicast system. Analogously, the right-hand side is the throughput uploaded by $v$. Constant $\Psi$ denotes the fairness of the system that must be accomplished and, as far as we consider a system in which all nodes download the same content of the same size, to provide a feasible solution $\Psi$ should be less than 1. That means, in case $\Psi = 0.2$ each node $v$ is enforced to upload at least 20% volume of throughput that it downloads.

By analogy to (7) and (8), the following sets of constraints represent a fair system with allowable background throughput $\Psi^*$ and a fairness coefficient $\overline{\Psi}$ which ensures contributors not to upload more than $\overline{\Psi}$ of the total obtained throughput $r$, respectively.

$$\Psi \sum_t x_t \leq \sum_t \beta_{ti} x_t + \Psi^* \quad i = 1, 2, ..., V, i \neq s \tag{13}$$

$$\overline{\Psi} \sum_t x_t \geq \sum_t \beta_{ti} x_t \quad i = 1, 2, ..., V, i \neq s \tag{14}$$

With regard to obtain feasible and positive solutions, $\overline{\Psi}$ must be greater than 0.

In order to generalize the fairness constraints a specialized coefficients depended on user $i$ might be introduced, namely $\Psi_i$ with $\Psi_i^*$ and $\overline{\Psi}_i$ for enforcing upload participation, and ensuring not to upload 'more than', respectively.

Note that in the formulation above all of the variables of the spanning tree packing problem $x_t$ and variables $r$, $x$ and $f$ defined for data rate allocation problem as well, may take both, continuous or integer values. While continuity of variables is not directly applicable in a majority of real networks, this model can serve as continuous relaxation for actual models, often used in algorithm development and as approximations.

## 4 Optimal Results

The computational experiments were carried out on an Intel Core 2 Duo CPU with 2.13 GHz clock and 4GB RAM, with x64 Windows 7 Professional. All algorithms were implemented in C++ under MS VS2008. Optimization tools used for optimization includes: *IBM ILOG CPLEX Optimization Studio Academic Research Edition 12.2* and *Gurobi Optimizer 3.0* and their C++ interfaces.

To compare the optimization tools and both linear approaches as well, we examine several instances of overlay multicast systems comprising nodes connected to the network using ADSLs. Root node can upload 1536 kbps, whereas upload limits of access links of other nodes are proportionally distributed among values 512, 1024 and 1536 kbps. For the sake of simplicity, we assume download capacities $d_i = 4096$ kbps are the same for all nodes in the system.

In the first experiments we focus on observing an impact of two concurrent linear formulations without fairness constraints and the performance of different

**Table 1.** Optimization results (problems without fairness constraints)

| | DRA | | | | FST | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | $T = 10000$ | | | $T = 5000$ | | | $T = 1000$ | | |
| | Result | | Time [s] | | Gap | Time [s] | | Gap | Time [s] | | Gap | Time [s] | |
| Size $V$ | CPL | GRB | CPL | GRB | [%] | CPL | GRB | [%] | CPL | GRB | [%] | CPL | GRB |
| 10 | 1194.67 | 1194.67 | 0 | 0 | 0.0 | 1 | 2 | 0.0 | 0 | 2 | 0.0 | 0 | 0 |
| 20 | 1077.89 | 1077.89 | 3 | 2 | 0.0 | 1 | 8 | 0.0 | 0 | 3 | 0.0 | 0 | 0 |
| 30 | 1059.31 | 1059.31 | 83 | 26 | 0.0 | 1 | 8 | 0.0 | 0 | 3 | 5.1 | 0 | 0 |
| 40 | 1063.38 | 1063.38 | 711 | 125 | 0.0 | 1 | 15 | 0.3 | 0 | 3 | 11.0 | 0 | 0 |
| 50 | x | 1044.90 | 3600 | 644 | 2.4 | 1 | 11 | 8.1 | 0 | 6 | 20.5 | 0 | 0 |
| 60 | x | 1041.36 | 3600 | 2646 | 8.6 | 2 | 23 | 13.8 | 1 | 6 | 21.9 | 0 | 0 |
| 70 | x | 1041.36 $f$ | 3600 | 3600 | 10.4 | 2 | 24 | 14.3 | 1 | 14 | 28.6 | 0 | 0 |
| 80 | x | 1038.40 $f$ | 3600 | 3600 | 13.7 | 2 | 26 | 15.8 | 1 | 5 | 28.8 | 0 | 0 |
| 90 | x | 1035.51 $f$ | 3600 | 3600 | 17.7 | 2 | 17 | 21.0 | 1 | 11 | 31.2 | 0 | 0 |
| 100 | x | 1039.52 $f$ | 3600 | 3600 | 18.8 | 2 | 18 | 23.5 | 2 | 8 | 31.3 | 0 | 0 |

solvers on the overall optimization performance. Table 1 compares optimization time and results of DRA and FST approaches solved in CPLEX (CPL) and GUROBI (GRB) for 10 scenarios of different sizes. For the unconstrained version of the throughput maximization problem modelled by the DRA formulation, the most effective solver is implemented in GUROBI's tool. First, it provides optimal results explicitly faster than the CPLEX solver (columns 4 and 5), second, GUROBI tool proves optimality of some scenarios with 50 and 60 nodes, whereas CPLEX is unable to find any feasible solution within one hour optimization. Last, for bigger instances (70-100), the GUROBI solver ensures at least feasible results ($f$) in one hour in contrary to its competitor which works effectively only for scenarios with smaller number of nodes.

On the other hand, the relaxed version of the throughput maximization problem, which comprises a set of predefined trees is optimized faster in the CPLEX solver, where trees are generated in a pure random way. Even for the scenarios with $T = 10000$ unique fractional trees it takes just a few seconds in order to obtain the optimum, while the optimizer of GUROBI computes even 26 seconds. Moreover, based on problem dimension in the meaning of number of nodes in the system $V$, generally affects computational time of algorithms provided by GUROBI. The more nodes are present in the problem, the longer duration time of the GUROBI optimizer should be expected.

Table 1 also reports the FST's time-based advantage of getting optimal results for the overlay multicast system aimed at content delivering at maximized throughput. The optimum for problems based on DRA formulation might be obtained in relatively long time which leads the model to be impractical from the network's point of view, especially if we consider an online multicast system which is supposed to route a traffic almost instantaneously. Since then, the FST formulation which provides the optimum in seconds of optimization duration is more applicable approach. However, as far as the FST is a relaxed version of the DRA, its best results might be distant to the DRA's optimums. Numerical experiments show that, the instances with $V = 20$ nodes can be effectively solved

using FST approach without any gap to the optimum obtained with DRA. Furthermore, despite the gap of FST results related to the results of DRA increases in the function of problem size, the tradeoff in using of FST might be provided if computational time is taken into account. For instance, FST results with 2.14% gap or 8.6% gap in relation to the DRA's ones are compensated by about 720 or 1300 times shorter computational time for a network which consists of 50 and 60 users, respectively. The gap parameter refers to the results' relation of DRA/FST-100%.

Similar relation in optimization time meaning and the result qualities might be shown for the problems with fairness constraints. Table 2 reports the optimization results for the problem with fairness coefficient $\Psi$ set to 60%. Considering the DRA formulation, the computational time required by solvers is almost the same as in case of unconstrained version of the problem. However, the additional fairness constraints applied to the FST model leads to increased computational time needed for obtaining optimal solution. It is rather noticeable for the solver of GUROBI because CPLEX algorithms still provide a solution within seconds. For $\Psi = 0.6$, where participating users are obligated to upload at least 60% value of their downloaded throughput, the result qualities of FST are comparable to solutions of DRA even for systems comprising $V = 80$ nodes and optimization time difference between the formulations shows FST outperforms DRA in explicit way. DRA formulation is solved approximately in the same amount of time in cases of unconstrained throughput maximization problem and both version of fairness as well. Note that, CPLEX and GUROBI results for the problem with $\overline{\Psi}$ coefficient (Tab. 3) which indicates maximum upload of total throughput $r$ (equations (8)) requires almost the same amount of computational time as in cases of fairness defined by (6) and unconstrained version of the problem. The FST relaxation for $\overline{\Psi} = 0.8$ provides extreme results depending on the network size $V$, i.e. for scenarios with less than 50 nodes, using even $T = 1000$ yields solution with the same objective value as DRA formulation, if $V = 60$ it is desired to have 10000 tree candidates to obtain non-zero throughput. In all other cases

**Table 2.** Optimization results (fairness coefficient $\Psi = 0.6$)

| | DRA | | | | FST | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | $T = 10000$ | | | $T = 5000$ | | | $T = 1000$ | | |
| | Result | | Time [s] | | Gap | Time [s] | | Gap | Time [s] | | Gap | Time [s] | |
| Size $V$ | CPL | GRB | CPL | GRB | [%] | CPL | GRB | [%] | CPL | GRB | [%] | CPL | GRB |
| 10 | 853.33 | 853.33 | 0 | 0 | 0.0 | 0 | 37 | 0.0 | 0 | 5 | 0.0 | 0 | 0 |
| 20 | 853.33 | 853.33 | 2 | 3 | 0.0 | 0 | 70 | 0.0 | 0 | 5 | 0.0 | 0 | 0 |
| 30 | 853.33 | 853.33 | 80 | 26 | 0.0 | 1 | 70 | 0.0 | 0 | 11 | 0.0 | 0 | 0 |
| 40 | 853.33 | 853.33 | 711 | 125 | 0.0 | 1 | 89 | 0.0 | 1 | 23 | 0.0 | 0 | 1 |
| 50 | x | 853.33 | 3600 | 701 | 0.0 | 3 | 71 | 0.0 | 1 | 26 | 2.6 | 0 | 1 |
| 60 | x | 853.33 | 3600 | 2649 | 0.0 | 3 | 75 | 0.0 | 1 | 15 | 4.7 | 1 | 2 |
| 70 | x | 853.33 $f$ | 3600 | 3600 | 0.0 | 4 | 84 | 0.0 | 2 | 21 | 12.8 | 1 | 2 |
| 80 | x | 853.33 $f$ | 3600 | 3600 | 0.0 | 4 | 126 | 0.0 | 2 | 74 | 13.4 | 1 | 2 |
| 90 | x | 853.33 $f$ | 3600 | 3600 | 0.2 | 5 | 252 | 4.3 | 2 | 83 | 16.5 | 1 | 2 |
| 100 | x | 853.33 $f$ | 3600 | 3600 | 1.1 | 5 | 93 | 6.8 | 2 | 99 | 16.8 | 1 | 2 |

**Table 3.** Optimization results (fairness coefficient $\overline{\Psi}$ 80%)

| | DRA | | | | FST | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | $T = 10000$ | | | $T = 5000$ | | | $T = 1000$ | | |
| | Result | | Time [s] | | Gap | Time [s] | | Gap | Time [s] | | Gap | Time [s] | |
| Size $V$ | CPL | GRB | CPL | GRB | [%] | CPL | GRB | [%] | CPL | GRB | [%] | CPL | GRB |
| 10 | 731.43 | 731.43 | 0 | 0 | 0.0 | 0 | 37 | 0.0 | 0 | 9 | 0.0 | 0 | 0 |
| 20 | 404.21 | 404.21 | 3 | 3 | 0.0 | 1 | 81 | 0.0 | 0 | 10 | 0.0 | 0 | 0 |
| 30 | 264.83 | 264.83 | 78 | 29 | 0.0 | 1 | 70 | 0.0 | 1 | 10 | 0.0 | 0 | 1 |
| 40 | 196.92 | 196.92 | 711 | 270 | 0.0 | 2 | 88 | 0.0 | 1 | 20 | 0.0 | 0 | 1 |
| 50 | x | 156.73 | 3600 | 1209 | 0.0 | 4 | 71 | 100 | 1 | 24 | 100 | 0 | 1 |
| 60 | x | 130.17 | 3600 | 2649 | 100 | 5 | 75 | 100 | 2 | 16 | 100 | 1 | 2 |
| 70 | x | 111.30 $f$ | 3600 | 3600 | 100 | 6 | 85 | 100 | 2 | 23 | 100 | 1 | 2 |
| 80 | x | 97.21 $f$ | 3600 | 3600 | 100 | 6 | 128 | 100 | 3 | 73 | 100 | 1 | 2 |
| 90 | x | 86.29 $f$ | 3600 | 3600 | 100 | 8 | 254 | 100 | 3 | 85 | 100 | 1 | 2 |
| 100 | x | 77.57 $f$ | 3600 | 3600 | 100 | 10 | 97 | 100 | 4 | 98 | 100 | 1 | 2 |

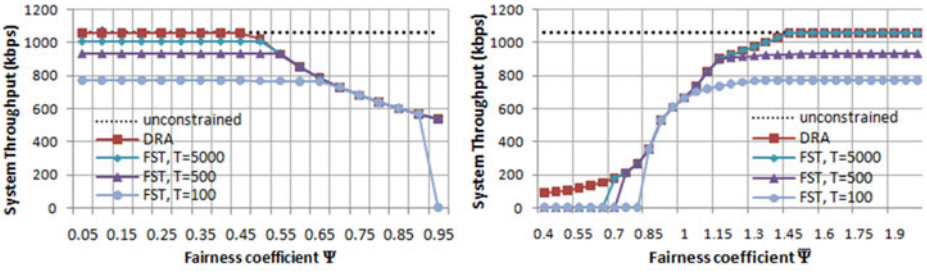



**Fig. 1.** Results for 30 node scenario with fairness constraints: required upload ratio $\Psi$ (left) and restricted upload ratio in relation to overall maximized throughput $\overline{\Psi}$ (right)

the relaxation with fractional spanning trees (even with 10000 random trees) cannot assign any positive data rate with regard to satisfy fairness given by $\overline{\Psi}$, thus the relative gaps to optimum provided by DRA equal to 100%.

Next, Fig. 1 presents optimal results in the case when fairness is incorporated into the system. An influence of fairness coefficients $\Psi$ and $\overline{\Psi}$ on overlay multicast throughput is shown for the scenario with $V = 30$ nodes. The left-hand side shows fairness which obliges every node to upload at least $\Psi$ fraction of it downloads, and right-hand side shows fairness where nodes are ensured not to upload more than $\overline{\Psi}$ of maximum throughput in the system. For $\Psi \leq 0.45$ results of the DRA model are not deteriorated in relation to results of unconstrained problem and degradation of the objective values starts from $\Psi \geq 0.5$. Considering the FST based model, an instance with $T = 10000$ provides equivalent results to the DRA. But in case of that the smaller number of predefined trees is available, the worse solution is obtained. An interesting result appears starting from $\Psi \geq 0.65$, for which values are equal for problems defined also by DRA and FTS with different number of trees. Thereby, if the fairness $\Psi$ exceeds

a given value it is sufficient to use FST with limited number of predefined tree candidates in order to obtain optimum as fast as possible. In turn, if $\overline{\Psi}$ version of fairness is applied, by allowing nodes to upload relatively more throughput than $r$ (or $\sum_t x_t$), leads to increased data rate in the whole overlay network. It can be easily noticed that, if the FST based model is used, it might be difficult to provide a non-zero feasible solution for small values of $\overline{\Psi}$. Since then, the efficient generating of spanning trees dedicated to throughput maximization might be a new challenge for such kinds of problems.

## 5 Conclusion

This paper presents the problem of throughput maximization in overlay multicast systems with fairness constraints. Two various interpretation of fairness are discussed. The first, required upload ratio obliges users to participate in uploading the content in relation to the value of its downloads. The second, restricted upload ratio limits a possible node's upload in order to avoid its resources are drained by misbehaving nodes. To model the problem, two linear programs are formulated. The more generic - data rate allocation formulation - employs conceptual flows, and its relaxed version - fractional spanning tree packing - relies on a set of predefined multicast tree candidates. Both of them are characterized with various advantages and might find their application in different situations. The FST-based program is solved explicitly faster than the DRA's one, while the result of DRA formulation might provide better throughputs allocation. By applying fairness constraints, the time required for solving the problems remains almost the same for DRA models and increases in case FST is used. However, the gap between results of DRA and FST is reduced if fairness conditions constraints the network, which leads the FST formulation to be much more applicable in practical systems. The models not only allow us to analytically study the interplay between features of fairness in overlay networks, but with the provided results also offer us generic guidelines in designing or implementation of the practical systems.

**Acknowledgements.** This work is partially supported by The Polish Ministry of Science and Higher Education.

## References

1. Akbari, B., Rabiee, H.R., Ghanbari, M.: An optimal discrete rate allocation for overlay video multicasting. Computer Communications 31(3), 551–562 (2008)
2. Benslimane, A. (ed.): Multimedia Multicast on the Internet. ISTE (2007)
3. Bharambe, A.R., Herley, C., Padmanabhan, V.N.: Analyzing and improving a bittorrent networks performance mechanisms. In: Proc. 25th IEEE Int. Conf. Computer Communications, INFOCOM 2006, pp. 1–12 (2006)
4. Chu, Y.-h., Rao, S.G., Zhang, H.: A case for end system multicast. In: Proceedings of ACM Sigmetrics, pp. 1–12 (2000)

5. Deering, S.E., Cheriton, D.R.: Multicast routing in datagram internetworks and extended lans. ACM Transactions on Computer Systems 8, 85–110 (1990)
6. Gurobi Optimization: Gurobi optimizer 3.0 (2010), http://www.gurobi.com
7. IBM ILOG CPLEX 12.1: User's Manual for CPLEX (2009)
8. Jain, K., Mahdian, M., Salavatipour, M.R.: Packing steiner trees. In: SODA 2003: Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 266–274. Society for Industrial and Applied Mathematics, Philadelphia (2003)
9. Kucharzak, M., Walkowiak, K.: Optimization of flows in level-constrained multiple trees for p2p multicast system. In: The Second International Conference on Advances in P2P Systems, AP2PS 2010, Florence, Italy (October 2010)
10. Lao, L., Hong Cui, J., Gerla, M.: Multicast service overlay design. In: Proc. of Second International Symposium on Wireless Communication Systems, ISWCS 2005, Philadelphia, Pennsylvania, USA (2005)
11. Leuf, B.: Peer to Peer: Collaboration and Sharing over the Internet. Addison-Wesley Longman Publishing Co., Inc., Boston (2002)
12. Pióro, M., Medhi, D.: Routing, Flow, and Capacity Design in Communication and Computer Networks. Morgan Kaufmann Publishers, San Francisco (2004)
13. Rawls, J.: The Theory of Justice. Harvard University Press, Cambridge (1971)
14. Sen, A.: On Economic Inequality. Clarendon Press, Oxford (1973)
15. Sentinelli, A., Marfia, G., Gerla, M., Kleinrock, L., Tewari, S.: Will iptv ride the peer-to-peer stream? peer-to-peer multimedia streaming. IEEE Communications Magazine 45(6), 86–92 (2007)
16. Shi, S., Turner, J.S.: Multicast routing and bandwidth dimensioning in overlay networks. IEEE Journal on Selected Areas in Communications 20, 1444–1455 (2002)
17. Walkowiak, K.: Network Design Problem for P2P Multicasting. In: International Network Optimization Conference, INOC 2009 (April 2009)
18. Walkowiak, K.M.: Offline Approach to Modeling and Optimization of Flows in Peer-to-Peer Systems. In: New Technologies, Mobility and Security, NTMS 2008, pp. 1–5. IEEE Press, Tangier (2008)
19. Wu, C., Li, B.: Optimal peer selection for minimum-delay peer-to-peer streaming with rateless codes. In: P2PMMS 2005: Proceedings of the ACM Workshop on Advances in Peer-to-Peer Multimedia Streaming, pp. 69–78. ACM, New York (2005)
20. Wu, C., Li, B.: Optimal rate allocation in overlay content distribution. Networking, 678–690 (2007)
21. Wu, C., Li, B.: On meeting p2p streaming bandwidth demand with limited supplies. In: Proc. of the Fifteenth Annual SPIE/ACM International Conference on Multimedia Computing and Networking (2008)
22. Wu, G., Chiueh, T.: Peer to peer file download and streaming. rpe report, tr-185 (2005)
23. Young, H.P.: Equity in Theory and Practice. Princeton University Press, Princeton (1994)
24. Zhu, Y., Li, B.: Overlay networks with linear capacity constraints. IEEE Trans. Parallel Distrib. Syst. 19(2), 159–173 (2008)

# Improving IPTV On-Demand Transmission Scheme over WiMax

Boris Goldstein and Gerges Mansour

St. Petersburg State University of Telecommunications, Russia
{bgold,gergesmansour}@niits.ru

**Abstract.** IPTV service, especially Video on Demand (VoD) service like famous sport games, popular movies will be a technology winner in the near future. Multicast transmission will be used in mobile IPTV over wireless access network like WiMax. In this paper, we will look multi-channel multicast algorithm to allocate content packets into several channels over WiMax. Proposed algorithm utilizes hybrid mechanism which combines multi-channel multicasting and unicast scheme to enhance not only service blocking probability but also reduce overall bandwidth consumption of the IPTV networks. In order to evaluate the performance, we compare proposed algorithm against traditional unicast and Multicast schemes.

**Keywords:** IPTV, multicast, unicast, Video on-demand.

## 1 Introduction

The Video-on-demand (VoD) service over IP based network is increasingly provided with dedicated unicast stream. In the VoD server's point of view, this unicast transmission is simple and makes no problem when there is enough capacity or the service request arrival rate is moderately low. However, in the real world, the video request popularity is highly skewed, so there are often concentrated requests for one particular content and burst requests in case of special events. Those cases are critical for the unicast transmission servers because the server's capacity is limited from its initial design so it results in high service blocking probability. By transmitting highly requested videos through multi-session multicasting and normal videos with unicast, we have achieved improvement on service blocking probability.

## 2 Categories of Video Services

The proposed Multi-channel Multicasting scheme was originally motivated from the Fast Broadcasting (FB) [1], which is implemented over cable TV environment as a Near-VoD service. Unlike TV broadcasting, IPTV stream cannot broadcast into all over the internet, therefore we modified to multi-channel concept for serving the highly request contents to provide an efficient way to transmit diverse multimedia stream to multiple users.

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 541–549, 2011.

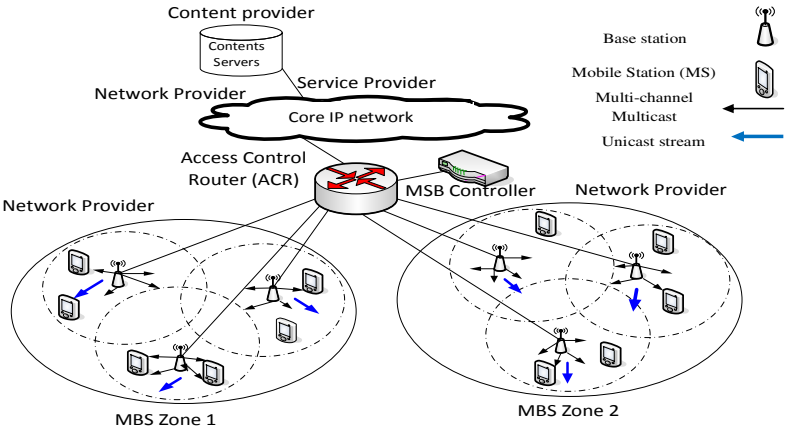



**Fig. 1.** Network architecture for MBS over multi-BSs

The decision of which contents are going to be transmitted by the multi-channel multicasting is decided by proposed adaptive resource allocation method that results in the lowest blocking probability. WiMAX supports both single-BS (Base station) mode and multi-BS MBS mode. The multi-BS mode uses Single Frequency Network (SFN) operation. The multi-BS MBS does not require the mobile station (MS) to be registered to any base station. MBS can be accessed when MSS is in the idle mode to allow low MSS power consumption [3]. The network architecture of video broadcast over multi-BS is shown in Fig. 1. The flexibility of Mobile WiMAX to support integrated MBS and unicast services enables a broader range of applications.

## 3 PTV VoD Service Scenarios

According to ITU-T IPTV service scenarios [2], VoD service is defined that "VoD is a video service which allows the end-users to select and watch video content at any point of time. The end-user has full control over choosing which program or clips to watch and when starting to watch." A proposed service scenario for on-demand IPTV is shown in Figs.1 and Fig. 2:

1) Video content with its metadata and Content Protection data that are produced and managed by the content provider are delivered to the service provider.
2) The service provider prepares the content as per the agreement between the content provider and the service provider.
3) As shown in Fig.2, when an end-user selects a VoD content, the request(s) is sent to the service provider and checks that the requested content already sending by multicasting or not. If there already exist multicast stream for the$V_i$, then the server let the client know the video's multicast group address and then the client can join the multicast stream. This procedure may include service negotiation (e.g., QoS, price, packaging option, etc.).

4) On the other hand, if there is no multicast stream for the video, the service provider interacts with the network provider checks within some threshold—by newly updated arrival rate from IPTV VoD service control function—to decide this content is highly requested content or not. In this time, if it is not highly requested contents, it delivered by unicast manner. However, if it is highly requested contents, MBS controller starts multi-channel multicast algorithm for content then to possibly negotiate the conditions of forwarding the content to the end-user. This procedure may include the network resource reservation to guarantee the contracted service level.
5) Upon completion of the above step 4, the service provider supplies the content access information (e.g., the multichannel multicast address that will be used to forward the content) and the end-user can then receive the video.

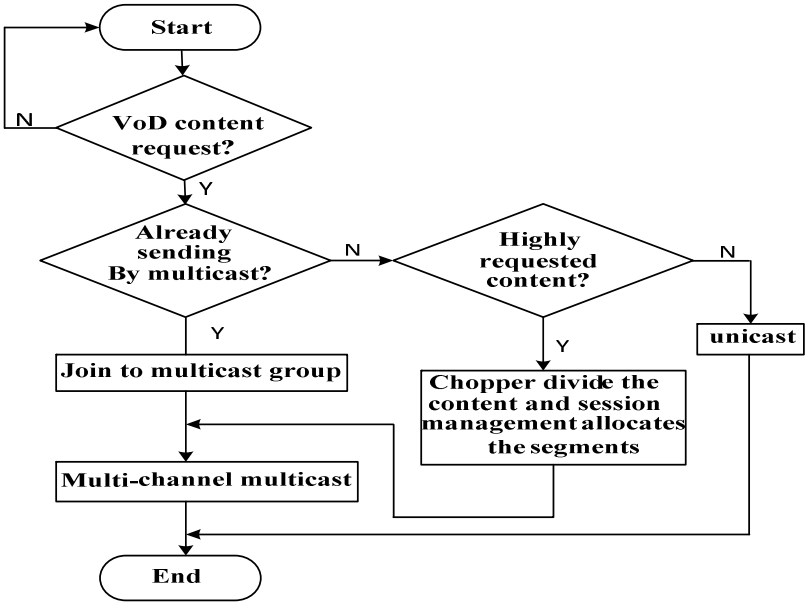



**Fig. 2.** Proposed service procedures for mobile IPTV VoD

## 4 The Proposed Algorithm

The overall network architecture of mobile IPTV VoD service over multi-BS is shown in Fig.1, The WiMax standard defines two types of MBS: single-BS access and multi-BS access. In general, multiple BS composes an MBS zone in mobile WiMAX [4]. The BS can send multicast data synchronously through the same connection identifier (CID) and security association (SA) carrying MBS data in the same MBS zone. Note that when all BS are in the same MBS zone, they have the same Multicast CID (MCID) for the same MBS multimedia stream transmission. Therefore, the multi-BS MBS does not require the MS to be registered to any BS. Moreover, multi-BS access allows all MS to use the same multicast connection ID during

handover within a single MBS zone, such that the MS can receive the MBS packets while moving within the MBS zone. Session Management/Transmission: An MBS session refers to a logical connection, established between an MS and the MBS controller, on which an MBS program is delivered to the users. The MBS controller creates and maintains session information. It also transmits the packets from the content chopper. Content chopping—In order to deliver the multicast VoD contents as if it is unicast transmission and satisfy the start up delay requirement, the content is needed to be chopped. According to ITU-T recommendation G.1010 [7], the start-up delay for VoD is less than 10 seconds. Therefore the maximum start-up delay in our proposed VoD system is less than 10 seconds. Suppose there is a $i_{th}$ movie with length $L_i$ . In multi-channel multicast, the number of channel, $n_i$, required for the $i_{th}$ video, with length $L_i$ that meets the requirement for the VoD start-up delay, $st$<10 sec by following equation:

$$n_i = \left[Log_2\left(\frac{L_i}{st} + 1\right)\right] \quad \text{Where} \quad L_i\text{: video length} \tag{1}$$

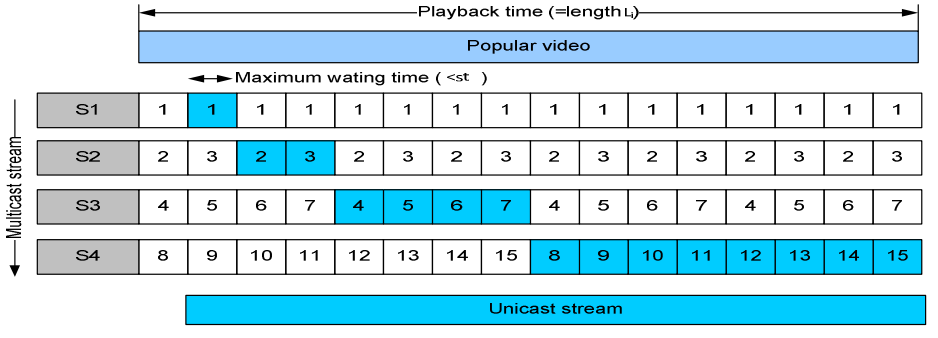



**Fig. 3.** Contents allocation for multi-channel multicasting

This equation was originally from the Fast Broadcasting (FB) and modified to adopt minimum start-up delay requirement. After calculate the number of channels, chopper divide the content equally into N segments.

$$N = \sum_{j=0} 2^j = 2^{n_i} - 1 \quad \text{whwe} \quad n_i\text{: number of channel} \tag{2}$$

Then session management allocates the segments to proper channels and segments are streamed continuously and periodically to its channel by geometrical series of 1, 2, 4… N. The example of allocation is shown in the Fig.3.

## 5 System Model and Numerical Analysis

In order to analyze the proposed mechanism [6], we consider a group of $K$ videos $V_1, \ldots V_K$ each of length L with arrival rates $\lambda_i, \ldots, \lambda_i$ respectively that are being transmitted using $C$ channels.

In addition, we assumed that $\lambda_i < \lambda_j$ for $1 \le i < j \le K$ , i.e., popularity of these videos increases gradually with the index so that $V_1$ and $V_K$ be the least and the most popular content respectively. The content's popularity (request frequency) ranking is calculated and given by the service provider based on their statistical data and their own service policy. For the numerical analysis, we assumed that request arrival for each video follow Zipf distribution [5]. Based on the assumptions and video request model, we calculate the blocking probability by Erlang's loss formula with the state space S and M/M/C/C model.

$$S = \{x | 0 \le x \le C\} \tag{3}$$

Fig.4 indicates the state transition diagram for the proposed scheme. The detailed notations used in this diagram are shown in Table I. First of all, based on this model, we will develop the formulas for steady state probability and blocking probability.

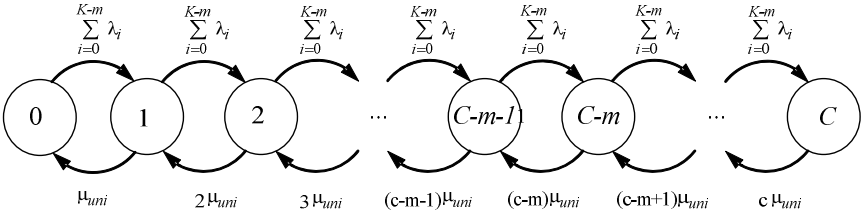



**Fig. 4.** Transition diagram for the proposed algorithm

**Table 1.** Notation for numerical analysis

| Notation | Explanation |
|---|---|
| $C$ | System capacity (the maximum number of concurrent system) |
| $K$ | Total number of contents |
| $m$ | Total number of multicast contents |
| $M$ | Total number of reserved channel for multicast system |
| $x$ | Number of connections in the coverage |
| $\boldsymbol{L_i}$ | Length of content $i$ |
| $\boldsymbol{\lambda_i}$ | Request rate of content $i$ |
| $\boldsymbol{\mu_i}$ | Service rate of stream $i$(= residual life of $L_i$) |
| $\boldsymbol{\mu_{uni}}$ | Average service rate of unicast stream |
| $\boldsymbol{P\_uni_i}$ | Probability that the unicast video $V_i$ is being served |

Let $P_x$ denote the steady-state probability that there are unicast streams, we can obtain

$$P_x = \begin{cases} p_0 \prod_{j=o}^{x-1} \frac{\sum_{i=1}^{K-m} \lambda_i}{(j+1)\mu_{uni}} = p_0 \left( \frac{\sum_{i=1}^{K-m} \lambda_i}{\mu_{uni}} \right) . \frac{1}{x!}, & for\ 0 \le x \le C - M \\ p_0 \left( \frac{\sum_{i=1}^{K-m} \lambda_i}{\mu_{uni}} \right)^x \left( \frac{\sum_{i=m}^{K} \lambda_i}{\mu_{uni}} \right)^{C-x} . \frac{1}{x!}, & for\ C - M < x \le C \\ 0 \quad . & for\ C < x \end{cases} \tag{4}$$

Since the steady state probability is not dependent on the service time distribution, but dependent on the mean service time, we calculate average service rate as follows:

$$\mu_{uni} = \sum_{i=1}^{K-m} P_{uni_i} \cdot \mu_i \;\; where \;\; P_{uni_i} = \frac{\lambda_i}{\sum_{j=1}^{K} \lambda_i} , \mu_i = \frac{1}{L_i} \tag{5}$$

From the normalization equation, we also obtain

$$p_0 = \left[ \sum_{j=0}^{C} \left( \frac{\sum_{i=0}^{C} \lambda_i}{\mu_{uni}} \right)^j \frac{1}{j!} \right]^{-1} \tag{6}$$

As a result, we can obtain the formulas for service blocking probability as follows:

$$P_{blocking} = \frac{\frac{\rho^{C-M}}{(C-M)!}}{\sum_{i=0}^{C-M} \frac{\rho^i}{i!}} , \text{ where } \rho = \frac{\sum_{i=1}^{K-m} (\lambda_i . \mathrm{P_{uni_i}})}{\sum_{i=1}^{K-m} (\mu_i \, . \mathrm{P_{uni_i}})} \tag{7}$$

Since the proposed on-demand system divides contents into two subgroups—popular contents and others—such that the former subgroup is assigned M channels for multi-channel multicasting and the latter subgroup is assigned the remaining C–M channels for unicast. Equation (8) shows minimum blocking probability of mobile wireless system and optimal multicast resource allocation can be determined by this formula. This formula tries to adaptively search the optimal number of channel assigned to the video by the newly updated arrival rate so as to minimize the bandwidth requirement.

$$P_{min}(C, M) = \min_{0 \le M < C} \left( \frac{\rho^{C-M}}{(C-M)!} \cdot \frac{1}{\sum_{i=0}^{C-M} \frac{\rho^i}{i!}} \right) \tag{8}$$

Next, we introduce the service provider's reward/penalty cost model to expect IPTV service providers' revenue. We assumed that when the base station successfully serves the IPTV VoD without blocking, the service provider receives a reward value of R. On the other hand, if a user is rejected, we assume that the service provider loses a value of L immediately. In prior art under the resource allocation policy, for example, if the system on average services N clients per unit time and rejects M clients per unit time, then the system revenue is

$$\sum N \cdot R - \sum M \cdot L \tag{9}$$

Finally, we define the total system revenue as follow:

$$\text{Revenue} = \sum_{i=1}^{C} i_\mu \times R \times \frac{\frac{\rho^i}{i!}}{\sum_{j=0}^{C} \frac{\rho^i}{j!}} - L \times \lambda \times \frac{\frac{\rho^C}{C!}}{\sum_{j=0}^{C} \frac{\rho^j}{j!}} \tag{10}$$

## 6 Performance Analysis

Based on the calculation result of the best number of videos to be multicasted, we show the required number of sessions for the popular videos compared with conventional unicast scheme where request arrival ratio λ = 6/min and the video length is fixed as 50 minutes in Fig. 5 show clearly that to serving the highly popular videos, using multi-session multicast takes much less number of video streaming sessions so that enables better performance for overall VoD services in blocking probability point of view, under the same condition.

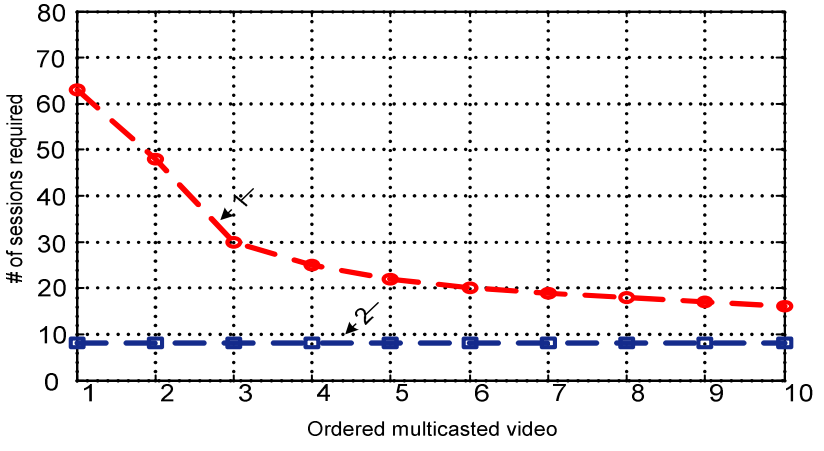



**Fig. 5.** Comparing the number of sessions required for highly requested Video with fixed video length (1- unicast, 2- multicast)

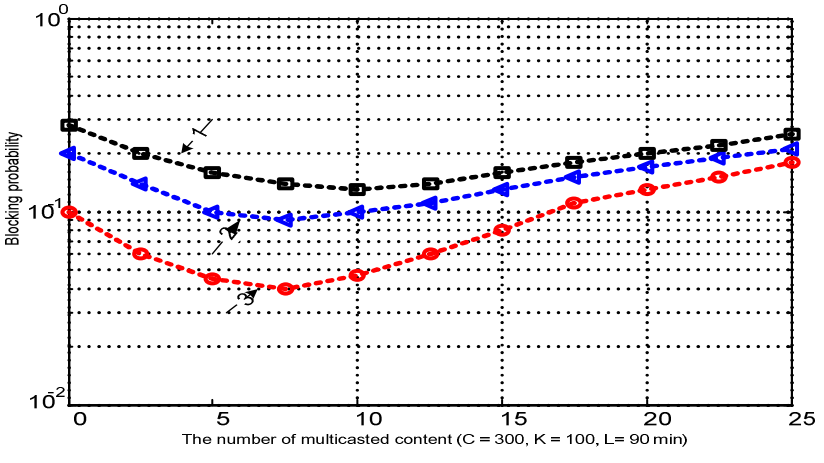



**Fig. 6.** Contents allocation for multi-channel multicasting (1- Average request rate 12/60, 2- average request rate 10/60, 3- average request ate 8/60)

For the performance evaluation, we assumed some parameters with specific values. The MBS server can serve 300 concurrent channels, and there are 100 contents that their lengths are average 90 minutes (5400 seconds). The request arrival ratio λ varies from 5/60 to 20/60, and λ is calculated by Zipf distribution with skew factor 0.271 [5]. We applied our adaptive resource allocation method to find the best resource allocation that minimizes service blocking probability. Fig.6 depicts the blocking probability with different request rates where the number of multicast contents varies from 1 to 25 that mean the total number of multi-channel multicast contents (m) in previous analysis model is changed 1 to 25. Under our environment, the results show that when we serve top 8 popular videos with multi-session multicasting, then we can have the best performance. This result shows how many contents need to be delivered by multi-channel multicast to achieve the best performance.

Based on the previous contents allocation, we show the performance of blocking probability of proposed mechanism compared with the existing mechanisms. As we can see from Fig.7, the VoD service blocking probability increases as the total service request rate increases. It is observed that when service request rate is lower than 3/min, customers cannot recognize the difference. However, as service request rate increases, their blocking probabilities become different. In such case, our proposed algorithm can offer the low blocking probability.

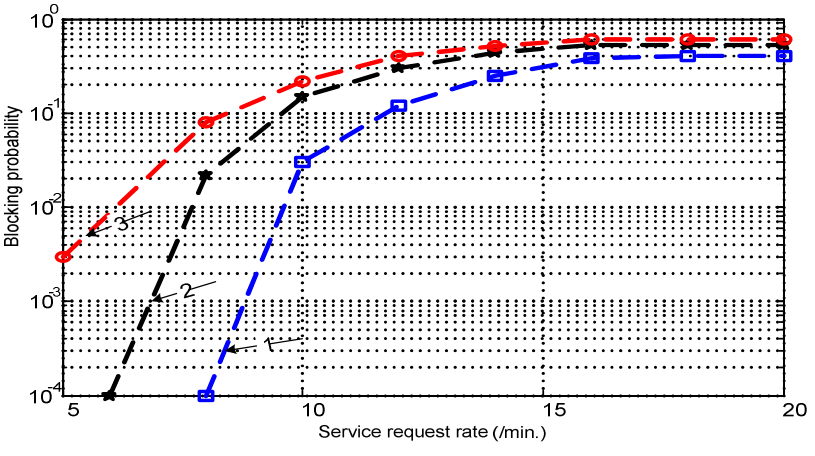



**Fig. 7.** Blocking probability of VoD services (1- Proposed algorithm, 2- unicast only, 3- multi-channel multicast only)

Now we evaluate the revenue of each scheme in the aspect of IPTV service provider. Fig.8 shows the results obtained by service providers' reward/penalty cost model. We assumed that if the base station successfully serves the IPTV VoD without blocking, the system receives a reward value of R (=$10). On the other hand, if a user is rejected, we assume that the service provider loses a value of L (=$5) immediately. This figure shows that as the traffic load increases, the revenue of each algorithm is slightly decreasing because of its blocking probability of services. As we mentioned before, this is the case that contents are highly requested, for example, the famous sports game or the popular movies. In such case, our proposed algorithm can offer the higher revenue.

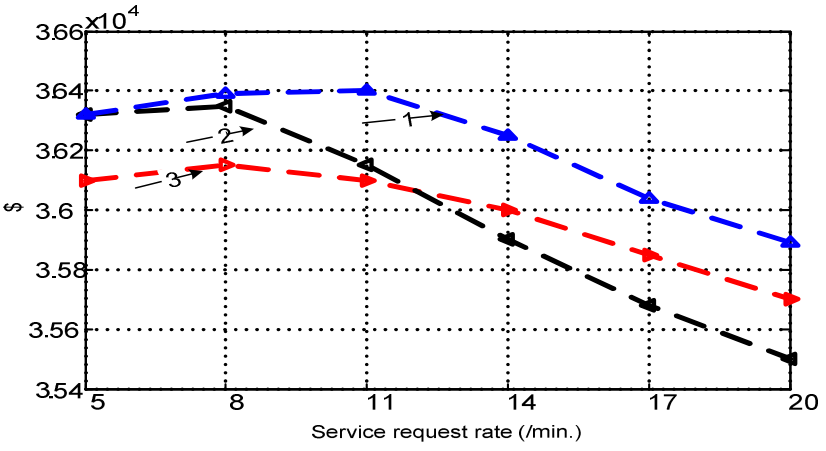



**Fig. 8.** Revenue analysis of IPTV service provider services (1- Proposed algorithm, 2- unicast only, 3- multi-channel multicast only)

## 7 Conclusion

In this paper, we have addressed a new algorithm that can efficiently provide mobile IPTV service over broadband wireless access network. Proposed algorithm combined unicast and multi-channel multicast mechanism that enhances not only service blocking probability but also reduces overall bandwidth consumption of the system. In order to analyze the performance of proposed algorithm, we use the one-dimensional Markov chain model. From the numerical analysis, we compared proposed algorithm against traditional unicast and multicast schemes. As a result, proposed scheme is able to improve IPTV service blocking probability over broadband wireless access network. In addition, by reducing the blocking probability, we can also achieve the higher revenue of IPTV service providers.

## References

1. John, S.: Fast data broadcasting and receiving for popular video service. IEEE Trans. Broadcasting 44(1), 100–105 (1998)
2. IPTV service scenarios, ITU-T draft Recommendation (December 2007)
3. Retnasothie, F., Ozdemir, M., Yucek, T.: Wireless IPTV over WiMAX: challenges and applications. In: Proceedings of the IEEE Annual Wireless and Microwave Technology Conference, pp. 1–5 (2006)
4. Pavel, M., Robert, B.: WiMAX Performance Evaluation. In: Proceedings of the Sixth International Conference on Networking 2007, p. 17 (2007)
5. Alonso, A.: Open-Loop Streaming Near-VOD Client. Master's thesis, University of Pamplona/Institute Eureka, Computer Science, Pamplona (2001)
6. Gerges, M.S.: Improvement transmission patterns IPTV-VoD in broadband networks. In: 63rd Scientific-Technical Conference, pp. 186–187. St. Petersburg State University of Telecommunications, Russia (2011)
7. End-user multimedia QoS categories, ITU-T Recommendation (November 2001)

# Event-Driven Content Management System for Smart Meeting Room

Victor Yu. Budkov, Alexander L. Ronzhin, S.V. Glazkov, and An.L. Ronzhin

SPIIRAS, 39, 14th line, St. Petersburg, Russia
{budkov,ronzhinal,glazkov,ronzhin}@iias.spb.su

**Abstract.** Context awareness is one of the key issues at the development of a content management system for meeting recording and teleconference support. An analysis of participant behavior including position in the meeting room, speech activity, face direction, usage of a projector or whiteboard allows the content management system to select actual multimedia streams for recording. The system operation diagram explaining meaningful events for context prediction and equipment actions during the meeting are considered. Experimental results have shown that the graphical content was selected correctly in 97% of whole meeting time.

**Keywords:** Smart space, context awareness, action recognition, speaker detection, content management.

## 1 Introduction

In a smart meeting environment, context-aware systems analyze user' behavior based on multimodal sensor data and provide proactive services for meeting support, including active control PTZ (pan, tilt and zoom) cameras, microphone arrays, context dependent automatic archiving and web-transmission of meeting data at the interaction. However, there is a lack of universal approaches to the problem of context prediction, especially for acting on predicted context [1]. Two ways to act on predicted context can be marked: (1) rule-based engines containing action rules for every particular prediction result; (2) machine learning techniques based on neural networks, dynamical Bayesian networks, Markov models, etc.

Problems of context representation, sensor uncertainty and unreliability are considered in the recently developed theory of context spaces [2]. However, there is no any accepted opinion on types and number of context spaces and their attributes. For example, user's location, environment, identity and time were analyzed at the context definition by Ryan et al. [3]. In [4], Dey described context as the user's emotional state, focus of attention, location and orientation, date and time, objects and people in a user's environment. Also W3C standard includes the delivery context representation, but selection of meaningful parameters influenced on context description is open issue and it is disscussed yet [5]. Three different categories of contexts were proposed in [6]: (1) real-time (location, orientation, temperature, noise level, phone profile, battery level, proximity, etc.); (2) historical (for instance, previous location, and previous humidity

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 550–560, 2011.

and device settings); (3) reasoned (movement, destination, weather, time, user activity, content format, relationship, etc.).

In [7], the context information used for service personalization and designing of multimedia applications for heterogeneous mobile devices were divided into the five categories: spatio-temporal (place, time), environment, personal, task, social. A personalization service based on user profile retrieves user context and context history information from context management services. It helps the user to get relevant content and services in the current situation.

Three types of contexts are proposed to use at fusion of multimodal information [8]: (1) a context of a subject domain, which contains some a priori knowledge, user's preferences, situation models, descriptions of objects and subjects, their possible activities and their locations relatively to each other; (2) a conversational context describing possible dialogues with the system and current e conditions; (3) a visual context including an analysis of gaze direction, gestures, actions of the user in the course of the observable situation.

The human beings, the physical and informational environments were considered by Dai et al. in the framework of two types of contexts [9]: interaction context representing interactive situations among people and environment context describing meeting room settings. They use propositions that the interaction context of a meeting has a hierarchical structure and expresses the context as a tree. User's standing-sitting states, changing user's location, face orientation, head gestures, hand actions, speaker turns and other events are analyzed for the context prediction. A Finite State Machine framework was introduced in order to classify these meaningful participants' actions. However, before the classification an event should be detected, so particular issues of signal capturing and feature extraction are appeared. The appropriate audio and video processing techniques are used for tracking participants in the smart environment.

Automatic analysis of audio and video data recorded during a meeting is not a trivial task, since it is necessary to track a lot of participants, who randomly change positions of their bodies, heads and gazes. Audio-visual tracking has been thoroughly investigated in the framework of CHIL and AMI/AMIDA projects [10].

Use of panoramic and personal cameras is suitable for recording a small-sized meeting, where all the participants are located at one table. In a medium size meeting room (~50 people), a larger space should be processed that affects on the cost of recording technical equipment too [11]. Distributed systems of microphone arrays, intelligent cameras and other sensors were employed for detecting participant's location and selection of a current speaker in the medium meeting room.

The developed smart room is intended for holding small and medium events with up to forty-two participants. Two groups of devices are used for tracking participants and recording speakers: (1) personal web-cameras serve for observation of participants located at the conference table; (2) four microphone arrays with different configurations and five video cameras of three types are used for audio source localization and video capturing of participants, who sit in rows of chairs in another part of the room.

In our research, three major types of proactive services are studied: (1) an active controlling PTZ camera to point on active speakers; (2) an automatic archiving of meeting data, including photos of participants' faces, video records of speakers, presentation slides and whiteboard sketches based on online context analysis;

(3) selection and web-transmission of the most actual multimedia content during the meeting in the smart room. The meeting web-transmission system, which deals with the latter service and uses some results of other services, is considered in the paper.

The rest of the paper is organized as follows. Section 2 describes the components of the meeting web-transmission system and their communications during online meeting. An example of the system functioning diagram explaining the analysis of meaningful events for context prediction and facilities actions is presented in Section 3. Experimental results are considered in Section 4. Conclusions and plans for future work are outlined in Section 5.

## 2 The Architecture of the Meeting Web-Transmission System

The developed meeting web-transmission system (MWTS) consists of five main software complexes and one control server. Figure 1 presents all six modules, which are marked by digits. The first complex is Multimedia Device Control System (MDCS), which joins modules that control all multimedia hard-software. This multimedia hard-software records behavior of participants and displays some presentation data. Second complex is Multichannel Personal Web Camera Processing System (PWPS), which captures and processes both audio and video streams from the personal web-cameras. The third complex stores the recorded audio and video data of the meeting in the smart room. The fourth complex is a database, which includes information about the meeting. Number six in Figure 1 is Meeting Control Server (MCS), which receives and analyses data from all other modules and gives information about received data to displaying web-system (DWS), which is showed as number five in Figure 1. DWS joins modules, which transmit multimedia content to remote participants. Content Management System (CMS) consists of third, fifth and sixth complexes.

The complex MDCS is responsible for multimedia devises work. Sketch Board System (SBS) allows subjects to use the plasma panel with the touch screen for drawing and writing notes. Presentation Control is responsible for loading, displaying and switching presentation slides. Multichannel Sound Localization System (MSLS) gives information about audio activity in the smart room. Multichannel Video Processing System (MVPS) is responsible for processing and recording of video streams incoming from the cameras, which are focused on the auditorium, presenter and sitting participants in the zone of chairs.

MPVPS consists of client modules, such as PWC, which supports work of personal cameras located on the conference table, as well as PWPS, which processes data from the PWCS modules. Audio files in the *wav* format and video files in the *avi* format, which were received from the personal cameras and processed by the MCS (change of the format, resolution and file name) images from MVPS, PCS, SB and PWC are added to the file storage.

The meeting database is realized by MySQL server and includes two tables: (1) basic information on all scheduled meetings and; (2) information about the current meeting, which includes some data for the meeting display system. DWS works as a web-page with several forms [12]. The data about form content are processed based on the AJAX technology. The transmitting of audio data to the client-computer is

based on the RTMP stream server and the Abobe Flash technology. MCS receives and analyses data from all the modules, as well as chooses of audio and video content for DWS. This analysis is based on the logical-time model [12].

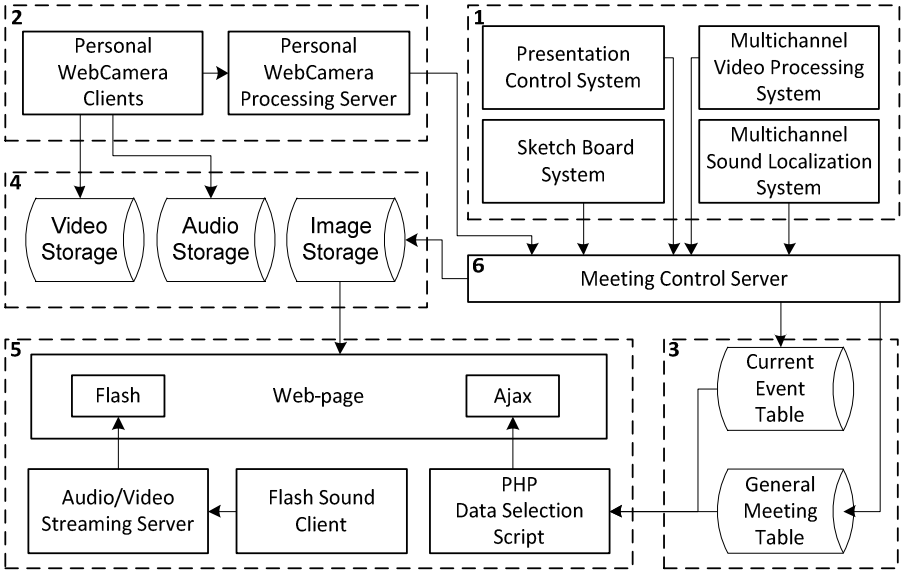



**Fig. 1.** Architecture of the meeting web-transmission system

Software modules of MWTS were installed on several personal computers joined in one local network, connection between them is based on transmitting messages in a string format by UDP packets. The first symbol in message is module identification number, then an identification number of message type follows. The list of main messages with parameters is described in Table 1.

Let us consider in detail the purpose of each module and its main functions at the teleconference. The module of Sketch Board System is used to extend capabilities of a lector in providing materials of presentation. Handwritten diagrams, formulas, graphs, made by SB, allow the speaker to bring to the participants the matter better. A graphic menu allows a user to choose paint brushes with varying thickness and color, to clear the screen and to switch between sheets of sketches. The application interface is optimized for use with touch screen devices. The image is saved each time when a user completes the movement of the hand on the touch screen or releases the mouse button. When the image is updated, this module sends a message to MCS, which copies the new image to a temporary folder, formats it if necessary, changes the format and/or resolution, and then updates the image in the file repository.

When the image is updated, this module sends a message to MCS, which copies the new image to a temporary folder, formats it if necessary, changes the format and/or resolution, and then updates the image in the file repository.

**Table 1.** List of messages for connection between modules

| System/ transmitting module | Message | System/ receiving module | Message description |
|---|---|---|---|
| MDCS/ MSLS | $A_1$={int i} | MCS | Audio activity is detected in the smart room. $i$ is a number of current speaker position (list of numbers: 0-31 – chairs in the right part of the room, 32-41 – chairs at the conference table and 42 – main speaker). |
| MDCS/ MVPS | $V_1$={int $i_1$, int $i_2$, bool $b_1$, bool $b_2$} | MCS | If there is a participant in the room. $i_1$ is the number of last recorded frame of view on the auditorium; $i_2$ is the number of last recorded frame from the camera focused on presenter if there is such, or -1 in the other case; $b_1$ is a flag about presence of a main speaker in the presentation zone; $b_2$ is a flag about detection of presenter's face in a frame. |
| MDCS/ PCS | $P_1$={int i} | MCS | Presentation is stared. $i$ is the total amount of slides in the current presentation. |
| | $P_2$={int $i_1$, int $i_2$} | | Slide is switched. $i_1$ is the total amount of slides in current presentation; $i_2$ is the number of current slide. |
| | $P_3$ | | Presentation is ended. |
| MDCS/ SBS | $S_1$ | MCS | SB module is turned on. |
| | $S_2$={int $i_1$, int $i_2$, string $s_1$} | | New note was written on SB, $i_1$ is the number of current image; $s_1$ – is its path. |
| | $S_3$ | | SB module is turned off. |
| MPVPS/ PWCC | $W_1$={int i} | MPVPS / PWCPS | A connected client controls the camera $i$. |
| | $W_2$={int $i_1$, int $i_{15}$, bool $b_1$, bool $b_2$} | | Data from personal cameras received. $i_{1..11}$ is a set of audio signal energies; $i_{12}$ is a message order number; $i_{13}$ is the number of current frame from camera; $i_{14}$ is the number of audio file; $i_{15}$ is a message time; $b_1$ is information about presence of a participant en face in the current frame; $b_2$ is information about presence of a participant half face in the current frame. |
| MPVPS/ PWCPS | $C_1$={bool b} | MPVPS / PWCC | Control recording data from personal web-cameras. $b$ is a flag for start/end of the recording. |
| | $C_2$={int $i_1$, int $i_2$, int $j_{1\_1}$, int $j_{1\_2}$, bool $b_{1\_1}$, bool $b_{1\_2}$, int $j_{1i\_1}$, int $j_{i1\_12}$, bool $b_{i1\_11}$, bool $b_{i1\_12}$} | MCS | Data about sitting participants at the conference table. $i_1$ is the total amount of participants; $i_2$ is a number of active participant client; $j_{1\_1}$ is a number of participant client; $j_{1\_2}$ is the number of frame from the $j_{1\_1}$ client; $b_{i1\_11}$ is information about presence of en face in the current frame; $b_{i1\_12}$ is information about presence of half face in current frame. |

The module of the Presentation Control System (PCS) controls by presentation with formats PPT/PPTX using the automation technology of Microsoft Office Power Point. The presentation starts by selecting a file. Once it starts, it alerts to MCS. Switching slides is made with the mouse, Bluetooth remote control, a voice command or via the web interface. Voice operated presentation, in particular switching slides, was implemented using a speech recognition system. When starting or switching slides, slide image is saved to a file and the MCS command is sent, containing the number of the current slide and total amount of slides. MCS copies this image, processes it and updates the image in the file repository. On the completion of a presentation the application notifies MCS about it.

MSLS captures audio streams by four arrays of microphones, calculates coordinates of the sound source in the room and assesses the boundaries of verbal communication. In the presence of a source, it starts the process of detection boundaries of speech. The recorded audio data are transferred to the MCS module for further recognition. Data on sound source position are averaged over all the arrays of microphones, and on the basis of the coordinates of the sound source the module calculates the identification number of the closest chair to the source. Then, numbers of chairs with speakers or a number assigned to the keynote speaker are sent to MCS.

PWC modules are responsible for receiving audio-visual data from the personal web-cameras mounted on the conference table and directed to sitting participants. Each module records audio signal in *wav* format, video data in *avi* format and frames from the camera in *bmp* format. After connecting to the server, PWCPS module sends its serial number. After registration, it expects the command from the server to start recording. When this command has been received, it starts recording both audio and video data. At specified intervals, the client sends to PWCPS a command that contains averaged energy, the number of command, time of sending the command and the file number, the data on the presence of the participant in front of the camera and the data on finding faces in the frame.

PWPS server processes data from PWC modules and selects the active speaker that is made using the algorithm of multi-channel voice activity detection [13]. For simultaneous start of the recording with all the modules, PWCC server expects the command from all the modules connected to it and synchronizes data coming from clients. Data are analyzed, and a message about the data from the cameras is sent to MVC. After receiving the command, server MVC processes images from the cameras with the participants, and stores them in the file storage.

DWS is implemented with a web-page. The web-page is divided into several forms that display the image coming from the source image data (web-cameras, IP-cameras, slides from projector, SB). The choice of graphical content, which appears in these forms, is obtained by using the AJAX technology of database events. Every 500 ms a request is made to update data on the web-page. If there is a necessity to update images, they are downloaded from the file storage and displayed in the selected form. Transferring audio data from the event is implemented on the basis of Adobe Flash modules and RTMP server based on open source server Red5. For data transfer module FSC is used, which is connected to the microphones of the personal web-cameras and microphones arrays and transmits audio data from them to the server. Selecting an audio stream to listen can be made automatically or manually by the user.

## 3 Work Diagram of the Content Management System

The work of the meeting web-transmission system and its components depends on the situation in the room. Figure 2 shows an example of component status switching and synchronization of audio and video content that depend on the incoming events from the modules for audio localization, video monitoring, multimedia devices control. CMS manages by the multimedia content output, which is accessible for remote meeting participant.

The events, which are generated by MCS, influence on the meeting web-transmission system work. Such events are showed on the right part in Figure 2 on the time axis. The presented events can be divided into four types by the following criteria: (1) by time; (2) by activity of the main speaker; (3) by activity of sitting participants; (4) by use of the presentation devices.

The first event showed in the diagram is "20 minutes before meeting". When the time of such event (event *E1*) comes, the meeting web-transmission system changes its mode to preparation and starts MSLS, MVPS and PWPS modules. Also the system sends messages to scroll down screen and to turn on lights in the smart room. A meeting logotype is displayed on the web-page. When an audio activity in the room was detected (event *E2*), then the sound activity detection (SAD) algorithm of the MSLS starts.

When appearance of a participant in the room was detected (event *E3*), MVPS changes its mode to "participant tracking" (PT) and the view on the auditorium is displayed on the web-page, as well as audio stream from the microphone, which analyses the whole room space for creating an acoustical map, is transmitted to web-page too. The projector and the plasma panel with the touch screen are turned on and SBS, PCS modules are started, when time before the meeting is about ten minutes (event *E4*).

If there is a participant in the zone of chairs (event *E5*), MVPS starts work in the mode "participant's registration". If there is a sitting participant at the conference table (event *E6*), then the PWC modules start. If MVPS detects a main speaker (event *E7*), and MSLS detects his/her audio activity (event *E9*, event *E12*), then frames from the camera focused on him/her are displayed on the web-page. The system starts recording audio stream from the microphone, which analyses the presentation zone. When the presentation is completely uploaded (event *E8*), its first slide is displayed on the web-page. If an audio activity was detected at the conference table (event *E10*), the system receives audio stream from the web-camera in front of the sitting participant.

When a slide was changed (event *E11*) the image of the new slide is transmitted to the web-page. If a participant uses the smart board (event *E13*), then the displayed presentation slide on the web-page is changed with the image from SBS. If audio activity was detected in the zone of chairs (event *E14*), then PTZ camera focuses on speaking participant and MVPS records his/her talk. At the same time image on the web-page changes with frames from PTZ camera and the audio steam is received from the microphone, which analyses this zone. When main speaker leaves the presentation zone (event *E15*), the view on the auditorium is displayed for remote participants.

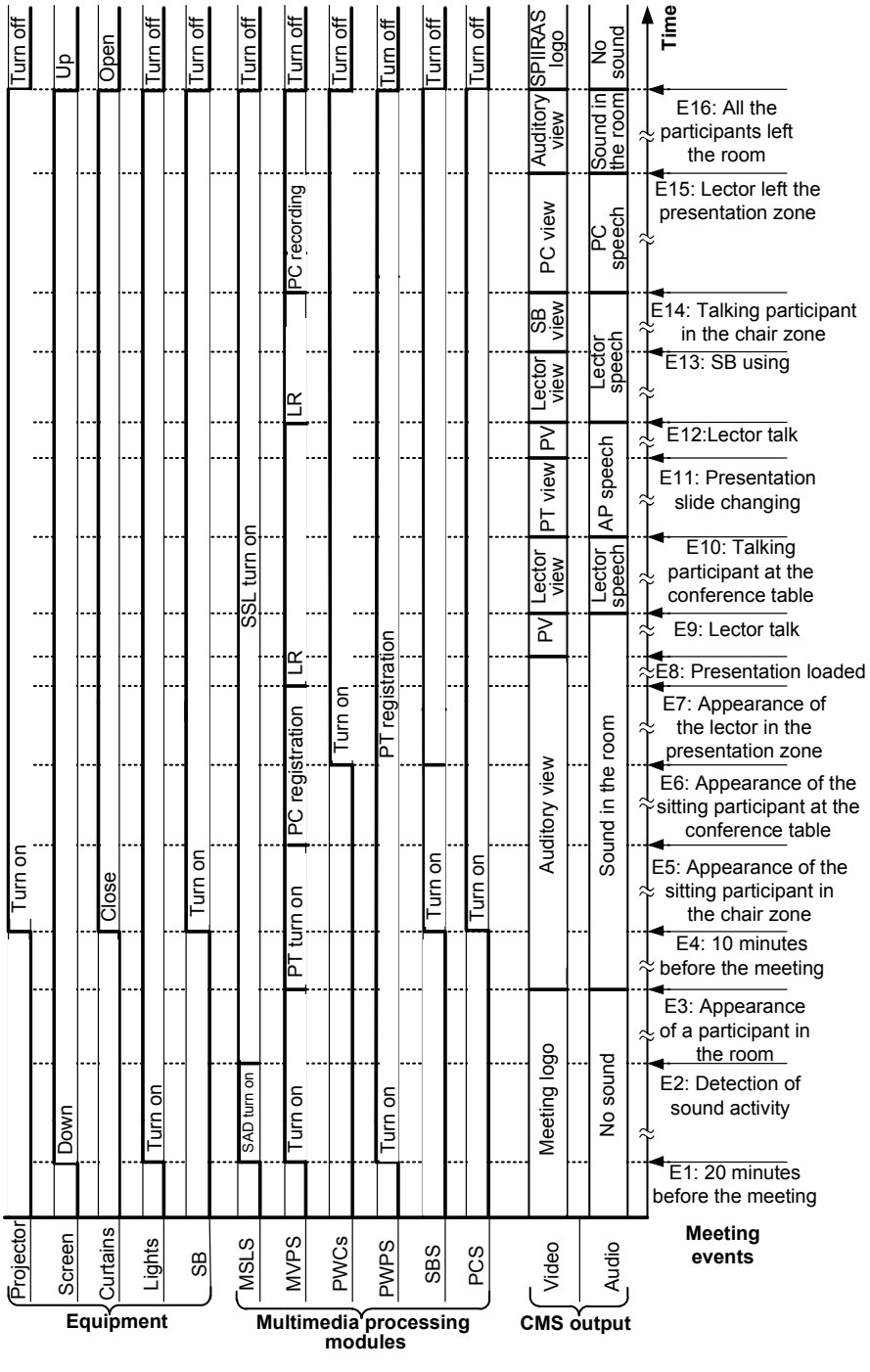



**Fig. 2.** An example of the meeting events processing in the system

When all meeting participants leave the room (event *E16*), then all the modules and devices are turned off. After ending of the meeting, the SPIIRAS logotype is displayed on the web-page, and no sound is transmitted. Such state of the web-page will be changed 20 minutes before the next meeting.

## 4 Experimental Results

Experimental results were obtained with a natural scenario, where several people discussed a problem in the SPIIRAS meeting room of 8.85x7.15x4.80m. One of the participants stayed in the presentation area and used the smart desk and the multimedia projector. Other participants were located at the conference table. The main speaker started his talk, when all the participants came together in the meeting room. Every participant could ask any questions after finish of the presentation.

The sound source localization system and the multichannel speech activity detection system were used for selection of a source of audio stream transmitted to remote participants [12]. Speech of the presenter was recorded by the microphone located above the presentation area. The built-in microphone of the web-camera assigned with the active participant sitting at the conference table was used for recording his/her speech.

The statistics of events, which effect on changing situation in the meeting room and selection of graphical components is presented in Table 2. During the registration stage all the graphical components have own layout, so incorrect web-page content could be compiled during presentation only. Changing states of the smart desk and the projector was correctly detected. Most part of the errors arose at detection of speech activity of the participants sitting at the conference table. This type of errors leads to selection of the camera, which captures another participant, as the result image of the active participant is not displayed on the web-page. However, his speech captured by currently selected camera is transmitted with some attenuation of the sound signal.

**Table 2.** List of the informational events occurred during the meeting

| **Event description/parameter** | **Amount of the event** | | | |
|---|---|---|---|---|
| | **Determined manually** | **Determined automatically** | | |
| | | **Num** | **FA** | **MS** |
| Participants | 5 | 5 | 0 | 0 |
| Participants sitting at the conference table | 4 | 4 | 0 | 0 |
| Slide changes | 22 | 22 | 0 | 0 |
| Smart desk usages | 1 | 1 | 0 | 0 |
| Speech activities of main speaker | 10 | 15 | 5 | 0 |
| Speech activities of sitting participants | 9 | 32 | 24 | 1 |
| Temporal inactivities in the auditorium | 2 | 2 | 0 | 0 |
| Turns from main speaker to sitting participants | 8 | 13 | 6 | 1 |
| Turns from sitting participants to main speaker | 7 | 13 | 6 | 0 |
| Turns from sitting participants to other sitting participants | 1 | 18 | 17 | 0 |
| Turns in total | 16 | 44 | 29 | 1 |

Experiment's results show that the most of errors (false alarm FA and miss rate MS) were made by the algorithm for detection of the active speaker, such errors occur when a participant at the conference table asks a question, but an image of other participant, which sits nearly, was displayed in the presenter dialog window. Such errors occur because there are mistakes made by SSL, because the smart room has a high-level of reverberation. In addition, distance between sitting participants at the conference table increases number of errors, because inaccuracy in SSL for the room is 0.5m. But the accuracy of switching between the active participant and the presenter is higher, because the distance is more than between participants at the conference table.

Also, miss of a speech activity of the main speaker leads to selection of the camera with view to the audience or other participant, whose speech was incorrectly detected. In total, about 37% graphical content were correctly selected at the analysis of the current situation in the meeting room, but time of false active participant displaying was about 3% of whole meeting time. At this moment, the developed web-page layout model was tested with the task of support of passive remote participants. To enhance his potentials a toolbar allowing a participant located outside the meeting room to ask a question to the presenter and to share the current discussion will be developed.

## 5 Conclusion

Context modeling, context reasoning, knowledge sharing are the most important challenges of the ambient intelligence design. Development of the context-aware meeting processing systems gives appreciable benefits for automation of recording, archiving and translation of the meeting stream. The multichannel audio-visual signal processing system based on the AdaBoost classifier for face detection and GCC-PHAT (General Cross-Correlation) method for sound source localization was developed for tracking participants in the medium-sized smart room. The analysis of participants' behavior and presentation equipment statuses is used for the context prediction and selection of audio and video sources, which transmit the most actual multimedia content for perception of the meeting. The system operation diagram explaining meaningful events for context prediction and equipment actions during the meeting was considered. Experimental results have shown that the graphical content was selected correctly in 97% of whole meeting time. The developed meeting web-transmission system allows remote participants to perceive whole events in the meeting room via personal computers or smartphones. Further work will be focused on enhancement of abilities of remote participation during events in the intelligent meeting room.

**Acknowledgments.** This work is supported by the grants #10-08-00199-a, #MD-501.2011.8, #14.740.11.0357 and performed in the framework of the joint "Audio-visual support system for Mobile E-meeting Participant" project between SPIIRAS and FRUCT.

## References


1. Boytsov, A., Zaslavsky, A.: Extending context spaces theory by proactive adaptation. In: Balandin, S., Dunaytsev, R., Koucheryavy, Y. (eds.) ruSMART 2010. LNCS, vol. 6294, pp. 1–12. Springer, Heidelberg (2010)
2. Padovitz, A., Loke, S.W., Zaslavsky, A.: Towards a Theory of Context Spaces. In: Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshop, Orlando, USA, pp. 38–42 (2004)
3. Morse, D.R., Ryan, N.S., Pascoe, J.: Enhanced reality fieldwork using hand-held computers in the field. Life Sciences Educational Computing 9(1), 18–20 (1998)
4. Dey, A.K., Salber, D., Abowd, G.D.: A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications. The Human-Computer Interaction 16(2-4), 97–166 (2001)
5. Delivery Context Overview for Device Independence, http://www.w3.org/TR/di-dco/
6. Moltchanov, B., Mannweiler, C., Simoes, J.: Context-Awareness Enabling New Business Models in Smart Spaces. In: Balandin, S., Dunaytsev, R., Koucheryavy, Y. (eds.) ruSMART 2010. LNCS, vol. 6294, pp. 13–25. Springer, Heidelberg (2010)
7. Goh, K.H., Tham, J.Y., Zhang, T., Laakko, T.: Context-Aware Scalable Multimedia Content Delivery Platform for Heterogeneous Mobile Devices. In: Proceedings of MMEDIA 2011, Budapest, Hungary, pp. 1–6 (2011)
8. Chai, J., Pan, S., Zhou, M.: MIND: A Context-based Multimodal Interpretation Framework. Kluwer Academic Publishers, Dordrecht (2005)
9. Dai, P., Tao, L., Xu, G.: Audio-Visual Fused Online Context Analysis Toward Smart Meeting Room. In: Indulska, J., Ma, J., Yang, L.T., Ungerer, T., Cao, J. (eds.) UIC 2007. LNCS, vol. 4611, pp. 868–877. Springer, Heidelberg (2007)
10. Waibel, A., Stiefelhagen, R. (eds.): Computers in the human interaction loop. Springer, Berlin (2009)
11. Rui, Y., Gupta, A., Grudin, J., He, L.: Automating lecture capture and broadcast: Technology and videography. Multimedia Systems 10, 3–15 (2004)
12. Ronzhin, A., Budkov, V., Karpov, A.: Multichannel System of Audio-Visual Support of Remote Mobile Participant at E-Meeting. In: Balandin, S., Dunaytsev, R., Koucheryavy, Y. (eds.) ruSMART 2010. LNCS, vol. 6294, pp. 62–71. Springer, Heidelberg (2010)
13. Ronzhin, A., Prischepa, M., Karpov, A.: A Video Monitoring Model with a Distributed Camera System for the Smart Space. In: Balandin, S., Dunaytsev, R., Koucheryavy, Y. (eds.) ruSMART 2010. LNCS, vol. 6294, pp. 102–110. Springer, Heidelberg (2010)

# A Semantic Model for Enhancing Network Services Management and Auditing

Carlos Rodrigues[1], Paulo Carvalho[1,⋆],
Luis M. Álvarez-Sabucedo[2], and Solange Rito Lima[1]

[1] University of Minho, Department of Informatics, 4710-057 Braga, Portugal
[2] University of Vigo, Dept. of Telematics, Vigo, Spain
pmc@di.uminho.pt

**Abstract.** The road toward ubiquity, heterogeneity and virtualization of network services and resources urges for a formal and systematic approach to network management tasks. In particular, the semantic characterization and modeling of services provided to users assume an essential role in fostering autonomic service management, service negotiation and auditing.

This paper is centered on the definition of an ontology for multiservice IP networks which intends to address multiple service management goals, namely: (i) to foster client and service provider interoperability; (ii) to manage network service contracts, facilitating the dynamic negotiation between clients and ISPs; (iii) to access and query SLA/SLSs data on an individual or aggregated basis to assist service provisioning in the network; and (iv) to sustain service monitoring and auditing. In order to take full advantage of the proposed semantic model, a service model API is provided to allow service management platforms to access the ontological contents. This ontological development also takes advantage of SWRL to discover new knowledge, enriching the possibilities of systems described using this support.

**Keywords:** Ontology, Network Service Management, SLA/SLS, Semantics, Multiservice IP Networks.

## 1 Introduction

The evolution of the Internet as a convergent communication infrastructure supporting a wide variety of applications and services poses new challenges and needs to network management, which has to be more focused on managing services instead of network equipment. This approach requires the capability of viewing the network as a large distributed system, offering an encompassing set of services to users.

Commonly, the type of service, its Quality of Service (QoS) requirements and other technical and administrative issues are settled between customers and Internet Service Providers (ISPs) through the establishment of Service Level Agreements (SLA). The technological component of this agreement is defined through Service Level Specifications (SLS). SLSs provide a valuable guidance to service deployment of network infrastructures and monitoring of contracts' compliance. Attending to the ever growing

⋆ Corresponding author.

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 561–574, 2011.

number of home and business customers, contracted services and network heterogeneity, the implementation and management of network services are very demanding tasks for ISPs. Besides the inherent complexity, this process may lead to inefficient policy implementation and poor resource management.

In fact, under the current variety of available services, e.g. IP telephony, 3-play or 4-play solutions, the interaction between service providers and end customers is rigid and rather limitative regarding service negotiation and auditing tasks. For instance, from a user point-of-view, the possibility of a short-term upgrade on access bandwidth to the Internet or a tight quality control of the subscribed service would be of undeniable relevance. From a service provider perspective, providing this sort of facilities, would clearly improve the level of service being offered, increasing competitiveness and resource management efficiency. These aspects are impelling ISPs to pursue autonomic solutions for service negotiation, configuration and management.

Although several proposals exist in the literature toward achieving dynamic service negotiation and management [1,2,3,4], the lack of a strong formal ground in addressing these tasks is evident and overcoming it is essential [5]. A formal specification of network services management semantics is required as the building blocks to create reasoning mechanisms to allow developing self-managed ISPs. By using a knowledge-based formal framework and an inference engine capable of reasoning over concepts, relations and changes of state, it is possible to create a more flexible and robust ground for specifying and implementing autonomic and adaptive management tasks.

As a contribution in this context, this work proposes an ontology specification in the domain of multiservice networks, which formally specifies the contractual and technical contents of SLAs, the network service management processes and their orchestration, promoting service autonomic management and configuration. This model provides support for a Service Management Platform that facilitates client and service provider interoperability, and service contracts management, including service data querying by the provider and, at some levels, by the client. This is enabled through a developed ServiceModel API, which allows the applicational use of the proposed ontology. The multiservice network semantic model is developed in Web Ontology Language (OWL), assisted by the Protégé-OWL tool. The use of Semantic Web technologies enhances service management modeling expansiveness and reusability.

This paper is structured as follows: research work on ontologies related to service definition and QoS is debated in Section 2; the developed model and its main modules are presented in Section 3; the way semantics is applied based on the developed API is discussed in Section 4; examples of practical use of the proposed model are provided in Section 5; the conclusions and future work are included in Section 6.

## 2 Related Work

Several research studies focusing on ontologies for network services support and QoS are found within the research community. While part of the ontologies focuses on Web Services (WS) QoS requirements, other concentrate on SLA/SLSs support.

QoSOnt [6] is an OWL ontology that centers on comparative QoS metrics and requirements definition. Although this ontology supplies the correct semantics for matchmaking, this was never demonstrated due to datatype limitations in OWL. To overcome

this limitation, a pure XML based solution was used, losing all of the virtues of OWL [7]. The DAML-QoS [8] is a QoS metrics ontology for WS developed in DAML+O. The ontology is divided in three layers: QoSProfile Layer, QoS Property Definition Layer and QoS Metrics Layer. In [9] a new Service Level Objective (SLO) concept, metrics' monitoring and statistical calculation semantics are presented. MOQ [10] is another proposal of a QoS semantics model for WS, but it is not exactly an ontology. It only specifies axioms and does not present a taxonomy structure or a dictionary of concepts. MonONTO [11] ontology aims at creating a knowledge base to support a client recommendation system. The ontology serves as a support to a decision recommendation tool by providing high-level information to the user about the compliance of the network facing the service level demands. In [12], an ontology aiming at the automation of network services management and mapping of services requirements into the network is proposed. The ontology is viewed in three perspectives: (i) the network service classification; (ii) the service level specification; and (iii) the deployment of network services. A group of generic ontologies to provide a framework for building SLAs is presented in [13]. In this context, the Unit Ontology contains all the comparable elements of an SLA, with the intention of supporting the creation of any type of measurable unit. It also allows the definition of unit supported comparators and the creation of comparison operations. The other examples of available ontologies are: the Temporal Ontology for temporal occurrences such as events and intervals; The Network Units Ontology for units related to telecommunications networks; and the SLA Ontology for basic SLA specification. Therefore, rather than a QoS ontology, a set of reusable ontologies is proposed for providing support for other QoS semantic model implementations. The OWL-based ontology NetQoSOnt [14] intends to be the support of a reasoning tool for service requirements matchmaking. It promotes the definition of SLSs containing quality parameters belonging to the following levels: the Quality of Experience, the Quality in the Application Level, the Quality in the Network Level and the Quality in the Link Level.

In the proposals discussed above, the lack of an unified and encompassing approach for semantic modeling of services and corresponding contracts in a multiservice environment is clear. In fact, most of the proposals are more focused on specification of network services metrics than on integrated service management. In the present work, a holistic model for modeling multiservice networks is provided paying special attention to the characterization and auditing of services quality. This ontology focuses on service contracts to assist network services' implementation by specifying how the defined contract elements are deployed in the network infrastructure, a feature not considered in the reviewed works. Although the proposed model is still evolving, its modular structure and the usage of Semantic Web technologies leaves room to model expansion and integration with other proposals.

## 3 Multiservice Network Ontology

The proposed model is divided in two main modules: the service management module and the network module. As illustrated in Figure 1, these modules are organized as a layered structure where the upper layer has a dependency relation with the lower layer. This

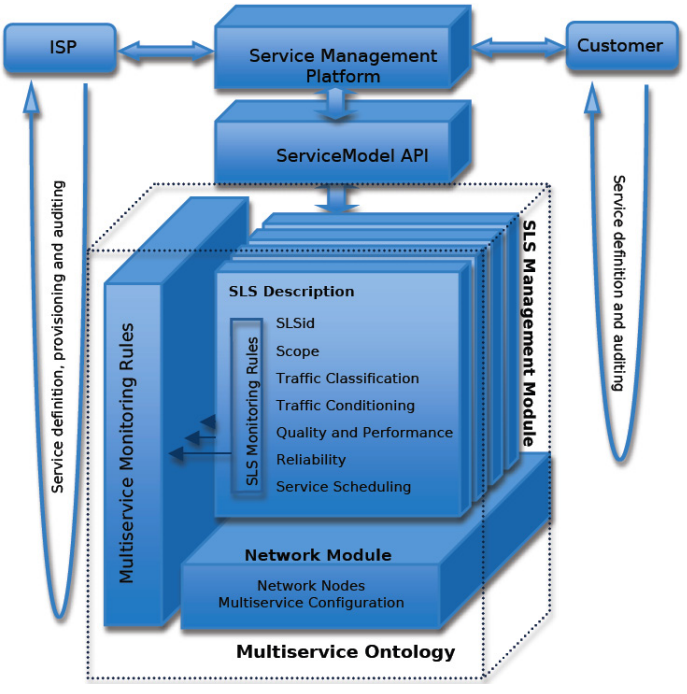



**Fig. 1.** Service model diagram

structure mimics real scenarios where the management component is, indeed, above the physical network. This formal representation of a network is expressed in formal terms using the support of OWL, following the principles from Methontology [15].

The network module, as stated above, acts as the base layer. It includes concepts of network node, network interface and network equipment configuration elements related to the implementation of contracted services in the network. The management module covers the domain network service management related to service contracts, including service monitoring rules. This module uses several elements of the network module. Services are categorized by relating them to a type of SLS [16,17]. According to recommendations from [18], ITU Y.1541 and 3GPP standards, current service types include: real-time services, multimedia services, data services, and default traffic service.

Another important component of the proposed service model regards to multiservice monitoring (see Figure 1). This implies the definition of the main monitoring issues to include in the multiservice ontology to assist auditing of Internet services both from an ISP and customer perspective. To service providers, it will also allow a tight control of services, network resources and related configuration procedures.

On top of the Multiservice Ontology, a complete ServiceModel API offers to a Service Management Platform the access to the ontological contents. Without detailing the construction of the ontology at this point, it is relevant to highlight the identification of competence questions. These are the first and the last step in this methodology and fulfill the need to establish the requirements and the outcomes of the ontology itself, i.e.,

which questions the ontology will be able to answer. In the present case, the definition of an ontology for multiservice IP networks intends to address multiple service-oriented goals. Possible competence questions include:

(i) *from a customer perspective*: Which type of service packs are available for subscription? Which service parameters are customizable? Which is the available bandwidth at a specific access point, for a particular service? Is the contracted service being delivery within the negotiated QoS?

(ii) *from an ISP perspective*: At an aggregate level, which is the allocated bandwidth for a particular service type? Which are the negotiated parameters per SLS? Which are the configuration parameters on each interface of an edge network node and the available bandwidth per interface? Which services are supported between specific ingress and egress interfaces? Are the QoE/QoS requirements of a particular service being accomplished? On which network points are occurring QoS violations?

In the description of modules provided in the sections below, a top-down approach will be followed to allow a broad view of the multiservice ontology.

### 3.1 Management Module

The management module is where service contracts or SLAs are defined and managed. The first concept is the Client which identifies the customer part of the contract and stores all client information. A client is related to at least one SLA which represents a service contract. An SLA can have more than one SLS. The SLS structure, illustrated in Figure 2, follows the recommendations in [16,17], and is briefly described below.

- *SLS Identification:* This field identifies the SLS for management purposes, being used by both provider and customer. It is composed of a unique SLS id parameter and a Service id parameter, allowing to identify multiple SLSs within the same service.
- *Scope:* The scope specifies the domain boundaries over which the service will be provided and managed, and where policies specified in a service contract are applied. Normally, SLSs are associated with unidirectional flows between at least one network entry point and at least one exit point. To cover bidirectionality, more than one SLS is associated with a service. The entry points and the exit points are expressed through ingress and egress interfaces, respectively (see Section 3.2). At least two `Interfaces` (ingress and egress) instances must be specified. The interface identification must be unique and is not restricted to the IP address (the identification can be defined at other protocol layer).
- *Traffic Classifier:* The `Traffic Classifier` specifies how the negotiated service flows are identified for differentiated service treatment. Following Diffserv terminology, multifield (MF) classification and behavior aggregate (BA) classification are supported (see Section 3.2). Usually, BA classification takes place over previously marked traffic, e.g. in network core nodes or, in the case of SLSs, between ISPs. Two traffic classifiers can be specified, an ingress traffic classifier and optionally an egress one. The ingress/egress classifier is then applied to each ingress/egress interface within the scope of the SLS.

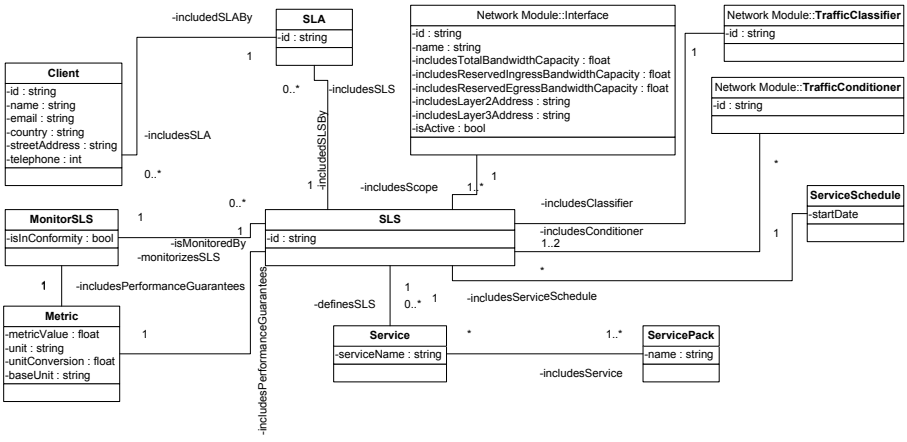



**Fig. 2.** SLS class diagram

– *Traffic Conditioner:* This field specifies the policies and mechanisms applied to traffic flows in order to guarantee traffic conformance with the traffic profiles previously specified. Traffic conditioning occurs after traffic classification, so there is always a relation between the traffic classifier and the traffic conditioner specified within a SLS.

  An unlimited number of `TrafficConditioner` instances can be specified. As in the traffic classifier property, the conditioners are divided into ingress and egress depending on their role. The ingress/egress conditioner is articulated with the ingress/egress classifier on each interface defined in the SLS scope as an ingress /egress QoS policy. This property is not mandatory.

– *Performance Guarantees:* The Performance Guarantee fields specify the guarantees of service quality and performance provided by the ISP. Four quality metrics are considered: delay, jitter, bandwidth and packet loss, expressed through instances of the `Bandwidth`, `Delay`, `Jitter` and `PacketLoss` `Metric` subclasses. The definition of at least one instance of these `Metric` subclasses is mandatory, except on the Default Service type of SLS. Whenever there is a performance guarantee specification, a traffic conditioning action must also be specified. Delay and jitter are usually specified by their maximum allowed value or by a pair consisting of a maximum upper bound and a quantile. Packet loss (edge-to-edge) is represented by the ratio between the packet loss detected at the egress node and the number of packets sent at ingress node. Instead of quantitative, quality and performance parameters can also be specified in a qualitative manner.

– *Reliability:* The Reliability is usually specified by the mean downtime (MDT) and by the maximum allowed time to repair (TTR). The no compliance of the negotiated parameters may result in a penalty for the ISP.

– *Service Schedule:* The Service Schedule defines the time period of service availability associated with an SLS. While a start date is always specified, an end date is only specified in case of a reservation, `ReservedServiceSchedule`, in which the client requests the service during a specific period of time. In the default case,

`StandardServiceSchedule`, only the service start date is specified, i.e., the contract must be explicitly terminated by the client.

- *Monitoring:* Monitoring refers to SLS' performance parameters monitoring and reporting. For that purpose, a measurement period, a reporting date and a threshold notification are specified. Other parameters such as the maximum outage time, total number of outage occurrences, reporting rules and reporting destination may be specified.
- *Type of Service:* The type of service is described by the `Service` class. This class allows the definition of services offered by the ISP to customers from a business-oriented perspective. Offered services are described through a set of qualitative metrics. The mapping from a qualitative service description to a quantitative service specification is assured by the ISP. The `Service` class allows to relate the SLS with a specific instance of service offering. It also helps establishing SLS templates on an application level. Services can be offered as a package (e.g. triple or quadruple play services) through the `ServicePack` class.

### 3.2 Network Module

At present, an ISP is represented as a cloud network, where only edge (ingress and egress) nodes are visible. The abstract representation of domain internal nodes and inherent internal service configuration mechanisms are left for future work. Therefore, instead of representing configuration elements at per-hop level, the model is focused on a per-domain level. In this module (see Figure 3) there are three key elements:

- *Node:* The `Node` class represents a network node (on the current model, corresponds to a domain border node). It is related to a set of `Interface` class instances.
- *Interface:* The `Interface` class represents ingress and egress points of the ISP domain. Specifically it allows the mapping of external network interfaces or entry/exit points of ISP border nodes. The interface supports a two-way traffic flow. It

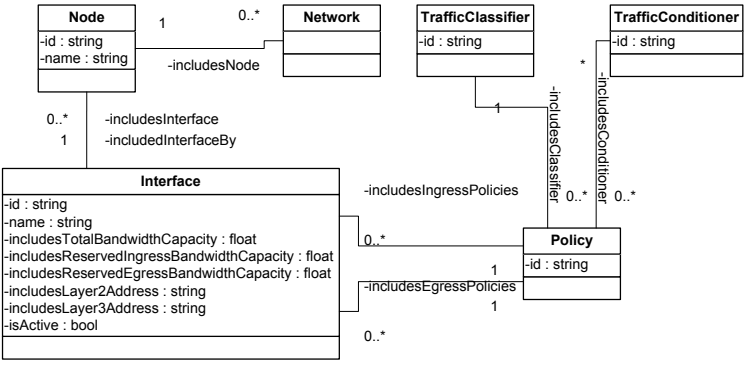



**Fig. 3.** Interface class diagram

is possible to attach layer 2 and layer 3 addresses to the interface concept in order to relate it to a real network interface in the ISP domain. Each interface has a total bandwidth capacity and a reserved bandwidth capacity specified dynamically for ingress traffic and egress traffic. For QoS purposes, it is possible to specify a set of QoS policies. In this case, a QoS policy is a relation between a traffic classifier instance and a set of traffic conditioner instances applied to traffic classified by the former. A QoS policy can be an ingress policy or an egress policy.

The `Interface` class, as illustrated in Figure 3, is defined by an identifier, link and network layer addresses and total bandwidth capacity both downstream and upstream. It includes two counters for ingress and egress reserved bandwidth of all contracted services applied to this interface. Each instance can be related to a set of QoS policies applied on incoming and outgoing traffic. A boolean value is also defined for interface state indication.

- *Traffic Classifier:* The `TrafficClassifier` class has two subclasses: MF and BA. The `BA` class instances, applied to previously marked traffic, only have one field, a relation with a `Mark` class instance. The `Mark` class contemplates all forms of aggregated traffic marking (such as DSCP, IPv6 FlowLabel, MPLS Exp, etc.). The `MF` class allows the definition of traffic classification rules with multiple fields. There are no constraints on the number of allowed fields and these are divided into: link, network and transport header fields. This means that several types of fields can be used: IPv4 and IPv6 addresses, IPv6 Flow Label, ATM VPI/VCI and MPLS Labels. The fields used in the classification rule are combined through a logic operator represented through the `LogicOperator` class instances `AND` and `OR`. For a more complex classifying rule definition, other `TrafficClassifier` class instances can be stated as fields, working as nested classification rules.

- *Traffic Conditioner:* The traffic conditioner is designed to measure traffic flows against a predetermined traffic profile and, depending on the type of conditioner, take a predefined action based on that measurement. Traffic conditioning is important to ensure that traffic flows enter the ISP network in conformance with the established service profile. It is also an important policy for handling packets according to their conformity level facing a certain traffic profile with the purpose of differentiating them in the network. According to their features, there are three `TrafficConditioner` subclasses: the `Marker`, `Policer` and `Shaper` classes. The policer usually takes an immediate action on packets according to their compliance against predefined traffic profile. A `Policer` class instance must have a set of traffic measurement parameters and at least two levels of actions defined. Three different policers are defined in the current model. The `TokenBucketPolicer` represents a single rate policer with two level actions (for in profile and out of profile traffic). The `SingleRateThreeColorMarker` and `TwoRateThreeColorMarker` are examples of policers with three levels of conformance actions. The `Shaper` is the only conditioner subclass where no immediate action is taken on traffic flows. Instead, all packets are buffered until traffic profile compliance is verified. The `Marker` class is a special type of conditioner which performs traffic marking and may be combined with other traffic conditioner elements.

### 3.3 Multiservice Monitoring Module

As illustrated in Figure 1, the aim of the monitoring system to develop is twofold:

(i) to monitor and control SLSs parameters in order to ensure that measured values are in conformance with the negotiated service quality levels. This auditing purpose involves a prior characterization of each service requirements, monitoring parameters and corresponding metrics, and the definition of appropriate measurement methodologies and tools to report multilevel QoS and performance metrics to users and system administrators;

(ii) to measure and control the usage of network resources. This includes the identification of network configuration aspects impacting on services performance, namely scheduling and queuing strategies on network nodes. In fact, monitoring network resources and triggering traffic control mechanisms accordingly will allow to maintain consistent quality levels for the supported services and the fulfillment of the negotiated SLSs.

Another main concern of this task is to congregate users and ISPs perspectives regarding the description and control of services quality. This means that the perceived service quality for users (Quality of Experience - QoE), commonly expressed through subjective parameters, has to be identified and mapped into objective and quantifiable QoS parameters, able to be effectively controlled by network service providers. Therefore, the articulation of QoE and QoS, and the identification of appropriate measurement methodologies for evaluating and controlling service quality levels in both perspectives (users and ISPs) is a main concern to cover in the present module.

In this context, multiservice monitoring is expected to provide a clear identification and layering of all monitoring issues to include in the multiservice ontology to assist auditing and control of negotiated service levels through the proposed Service Management Platform.

### 3.4 The VoIP Service as Example

As mentioned before, a service provider may describe each provided service through a set of qualitative metric values, which are then mapped to quantitative values to assist, for instance, configuration and service control. For example, a VoIP type of service can be described as:

*VoIP Service*
Bandwidth: `Low_Bandwidth` = at least 1 Mbps
Delay: `VeryLow_Delay` = at most 100 ms
Jitter: `VeryLow_Jitter` = at most 50 ms
Packet Loss: `VeryLow_Loss` = at most 0.001 % of lost packets

This type of service description is used for SLS classification in accordance with the specified metrics. In other way, SLSs can be built based on the type of service description when required. An example of an SLS instance for the VoIP service is shown in Figure 4.

When the SLS instance is set, the `TrafficClassifer` and `Traffic Conditioner` specified lead to QoS policy instances. A relation is then established between each QoS policy and network interfaces instances specified in the scope of the

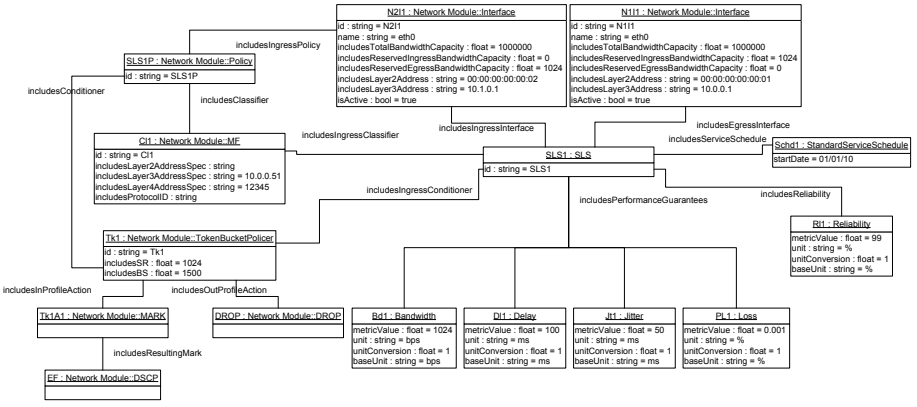



**Fig. 4.** SLS example diagram for VoIP service

SLS (Figure 4). This policy information is useful for automating the deployment of QoS mechanisms in the ISP network infrastructure.

By establishing relations among all these entities, a change in one of them affects all other related entities. For example, a change in an SLS parameter is spread through all the corresponding SLS configurations in the network infrastructure.

## 4 Applying Semantics

This section discusses how the presented model is converted into an ontological support. Thus, the characterization of the multiservice domain can be used in further software solutions ranging from web contents to complex software agents responsible for decision making. This ontology was developed according to the basis proposed by Methontology [15]. In this way, it is guaranteed its conformance with a set of methodological rules and the final product can be traced to its origin and reused in a simple and cost-effective manner.

The proposed ontology provides the main concepts and properties required to describe multiple services levels and corresponding quality in a network domain. For its implementation, according to the terminology proposed in Methontology, it was used Protégé to generate the OWL representation. This representation uses not only classes and properties, but it also includes restrictions on the values of the previous ones. Therefore, it is ensured the conformance of current contents and future pieces of information to the established parameters of the system.

Besides per-class restrictions, a set of general rules are defined for establishing new rule-based relations between individuals. These rules are expressed using SWRL and they are applied to check information in order to discover new possible instances and properties within the system. So far, there are defined rules for (i) validation of interfaces capacity included in a contract scope; (ii) compliance verification of monitored metrics in relation to service contract specifications; (iii) changing interfaces network status; (iv) qualitative classification of performance metrics; and (v) classification of

SLSs according to the type of service. For example, the following rule states that if all SLS performance metrics have qualitative values matching a definition of a type of service then the SLS specifies a service of that type.

$$
\begin{aligned}
&SLS(?sls) \wedge Service(?service) \\
&\wedge includesBandwidth(?sls, ?bandwidth) \\
&\wedge includesBandwidthQualValue(?service, ?qualiBandwidth) \\
&\wedge includesDelay(?sls, ?delay) \\
&\wedge includesDelayQualValue(?service, ?qualiDelay) \\
&\wedge includesJitter(?sls, ?jitter) \\
&\wedge includesJitterQualValue(?service, ?qualiJitter) \\
&\wedge includesLossQualValue(?service, ?qualiLoss) \\
&\wedge includesPacketLoss(?sls, ?loss) \\
&\wedge includesQualitativeValue(?bandwidth, ?qualiBandwidth) \\
&\wedge includesQualitativeValue(?delay, ?qualiDelay) \\
&\wedge includesQualitativeValue(?jitter, ?qualiJitter) \\
&\wedge includesQualitativeValue(?loss, ?qualiLoss) \\
&\rightarrow definesSLS(?service, ?sls)
\end{aligned}
$$

Additional rules are defined for the above mentioned issues. Nevertheless, for other purposes, it is suggested to define rules at application level due to the complexity and limitations of SWRL [19] at using knowledge from different sources and involving advanced logical checks.

On the top of the provided ontology, it is developed a complete software API. This API, referred as the ServiceModel API, is implemented following the diagram presented in Figure 5.

The Jena Framework [20] plays a major role in the developed software. It provides support for working with RDF and OWL based archives. The handling of OWL entities (classes, individuals and restrictions) is provided by the Jena Ontology API. Recall that the ontological content can be accessed from the local computer or from a remote server. The Pellet [21] engine is the reasoner used due to its SWRL [22] support.

Working on top of the Jena framework, the Jena beans API binds RDF resources (in this case, OWL classes) to Java beans simplifying the process of Java-to-RDF conversion. This feature enables users to work with individuals as Java objects.

The persistence of the knowledge is guaranteed by the TDB [23] technology, which is clearly simpler and more efficient than the SDB solution (uses SQL databases for

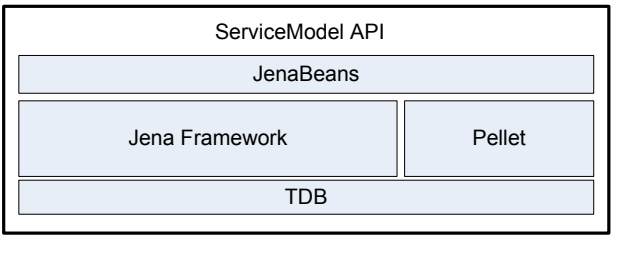



**Fig. 5.** ServiceModel API structure diagram

storing RDF datasets). However, the API integration of an SDB solution is not totally abandoned.

The ServiceModel APl intents to assist future projects in several goals: (i) to foster client and service provider interoperability; (ii) to manage network service contracts, facilitating the dynamic negotiation between clients and ISPs; (iii) to access and query SLA/SLSs data on a individual or aggregated basis to assist service provisioning in the network; and (iv) to assist service monitoring and auditing. Therefore, this API, aimed to sustain further developments, supports in a straightforward manner for software developers the following features: (i) the insertion and removal of information on the Knowledge Base (creating/destroying individuals); (ii) the validation of the Knowledge Base information (classification and realization); (iii) establishment of more complex rules based relations (not possible through SWRL); (iv) Knowledge Base querying, implemented through SPARQL [24] and the ARQL Jena API; and (v) Knowledge Base persistence.

## 5 Practical Application

Once the semantic model for fully describing SLAs/SLSs is set, services can be provided on top of it. Semantics is not, in general, an end by itself. On the contrary, its use is motivated by achieving further goals. In the case of this proposal, it is generated a framework to boost interoperability and advanced data mining features. Bearing that in mind, services were derived as proof-of-concept. Firstly, it is suggested a RDFa support to introduce annotations on xhtml descriptions about SLAs and SLSs. Afterwards, it is suggested the use of a semantic engine to recover information from a repository of information regarding the formers.

RDFa [25] is a semantic technology that empowers users to include semantic annotations on XML (or xhtml) contents. These annotations are invisible for human user but easily recovered for software agents using GRDDL [26]. It is important to keep in mind that both technologies are official recommendations from W3C.

Taking as a basis the provided OWL model for describing the system, annotations can be included in xhtml describing SLAs and SLSs. For the sake of clarity, it is included the following example:

```
<p xmlns:sla="http://owl.det.uvigo.es/sls/">
  The provided connection under the interface
  <span property="sls:id">Service1</span>,
  provides a total BW of
  <span property="sls:includesTotalBandwithCapacity">
   20 MBps</span>.
</p>
```

As shown in this xhtml snippet, a network interface and some of its properties are described. This definition of capacities of the Network Module can be directly recovered using GRDDL. The use case expected for this functionality is related to the web pages of ISPs. Service providers, when offering their services, can include this information into their web pages. Users will be able to recover this information through software agents on their behalf, and include it into a data repository for further decisions.

Once these pieces of information are included in a Semantic Database, regardless of its origin, either from GRDDL extraction or from other sources, it is possible to get added-value services. Using SPARQL Queries [24], for example, it is possible to locate SLAs/SLSs fulfilling specific properties the user is interested in. The only requirement is to identify the graph matching the desired properties and implement the corresponding SPARQL query. Authors successfully tested this feature by means of the API provided. It is actually rather simple to deploy a software tool that looks for, for instance, the cheapest SLA in the market or the one offering the fastest network access, among other features.

## 6 Conclusions and Future Work

This paper has presented an innovative approach to the development of a semantic model in the domain of multiservice networks. This model formally specifies concepts related to service and SLS definition, network service management, configuration and auditing, creating the reasoning mechanisms to ground the development self-managed ISPs. Although being conceptually aligned with the differentiated service model, the solution is generic without being tied to a specific QoS paradigm.

The usefulness of the present semantic service modeling has been pointed out for multiple applications in the context of multiservice management. In particular, aspects such as dynamic service negotiation between service provides and end customers, and auditing of Internet services being provided may be strongly improved as consequence of using the proposed ontology.

Possibilities and features from this ontology are also presented to software developers by means of a ServiceModel API. The functionality within this library can be used for the above mentioned goals. Due to the modular schema for this software component, its inclusion on future projects constitutes a simple task that will provide a useful support in further developments.

## References

1. D'Arienzo, M., Pescapè, A., Ventre, G.: Dynamic service management in heterogeneous networks. Journal of Network and System Management 12(2) (2004)
2. Sarangan, V., Chen, J.: Comparative study of protocols for dynamic service negotiation in next generation internet. IEEE Communications Magazine 44(3), 151–156 (2006)
3. Cheng, Y., Leon-Garcia, A., Foster, I.: Toward an autonomic service management framework: A holistic vision of SOA, AON, and autonomic computing. IEEE Communications Magazine 46(5), 138–146 (2008)
4. Zaheer, F.E., Xiao, J., Boutaba, R.: Multi-provider service negotiation and contracting in network virtualization. In: IEEE NOMS 2010, pp. 471–478 (2010)
5. Atkinson, E., Floyd, E.: IAB concerns and recommendations regarding internet research and evolution. RFC 3869, Internet Engineering Task Force (August 2004)
6. Dobson, G., Lock, R., Sommerville, I.: Qosont: a qos ontology for service-centric systems. In: EUROMICRO 2005: Proceedings of the 31st EUROMICRO Conference on Software Engineering and Advanced Applications, pp. 80–87. IEEE Computer Society, Washington, DC (2005)

7. Dobson, G., Sanchez-Macian, A.: Towards unified qos/sla ontologies. In: SCW 2006: Proceedings of the IEEE Services Computing Workshops, pp. 169–174. IEEE Computer Society, Washington, DC (2006)
8. Zhou, C., Chia, L.T., Lee, B.S.: Daml-qos ontology for web services. In: IEEE International Conference on Web Services. IEEE Computer Society, Los Alamitos (2004)
9. Zhou, C., Chia, L.T., Lee, B.S.: Qos measurement issues with daml-qos ontology. In: IEEE International Conference on E-Business Engineering, pp. 395–403. IEEE Computer Society, Los Alamitos (2005)
10. Kim, H.M., Sengupta, A., Evermann, J.: Moq: Web services ontologies for qos and general quality evaluations. Int. Journal of Metadata, Semantics and Ontologies 2(3), 195–200 (2007)
11. Moraes, P., Sampaio, L., Monteiro, J., Portnoi, M.: Mononto: A domain ontology for network monitoring and recommendation for advanced internet applications users. In: Network Operations and Management Symposium Workshops, IEEE NOMS 2008, pp. 116–123 (April 2008)
12. Alípio, P., Neves, J., Carvalho, P.: An ontology for network services. In: International Conference on Computational Science, vol. (3), pp. 240–243 (2006)
13. Green, L.: Service level agreements: an ontological approach. In: ICEC 2006: Proceedings of the 8th International Conference on Electronic Commerce, pp. 185–194. ACM, New York (2006)
14. Prudencio, A.C., Willrich, R., Diaz, M., Tazi, S.: Quality of service specifications: A semantic approach. In: IEEE International Symposium on Network Computing and Applications, pp. 219–226 (2009)
15. Fernández-López, M., Gómez-Pérez, A., Juristo, N.: Methontology: From ontological art towards ontological engineering. In: Symposium on Ontological Art Towards Ontological Engineering of AAAI, pp. 33–40 (1997)
16. Morand, P., Boucadair, M., Levis, P., Egan, R., Asgari, H., Griffin, D., Griem, J., Spencer, J., Trimintzios, P., Howarth, M., Wang, N., Flegkas, P., Ho, K., Georgoulas, S., Pavlou, G., Georgatsos, P., Damilatis, T.: Mescal D1.3 - Final Specification of Protocols and Algorithms for Inter-domain SLS Management and Traffic Engineering for QoS-based IP Service Delivery and their Test Requirements. Mescal Project IST-2001-37961 (January 2005)
17. Diaconescu, A., Antonio, S., Esposito, M., Romano, S., Potts, M.: Cadenus D2.3 - Resource Management in SLA Networks. Cadenus Project IST-1999-11017 (May 2003)
18. Babiarz, J., Chan, K., Baker, F.: Configuration Guidelines for DiffServ Service Classes. RFC 4594 (Informational) (August 2006)
19. Zwaal, H., Hutschemaekers, M., Verheijen, M.: Manipulating context information with swrl. A-MUSE Deliverable D3.12 (2006)
20. McBride, B.: Jena: a semantic web toolkit. IEEE Internet Computing 6(6), 55–59 (2002)
21. Sirin, E., Parsia, B., Grau, B., Kalyanpur, A., Katz, Y.: Pellet: A practical OWL-DL reasoner. Journal of Web Semantics 5(2), 51–53 (2007)
22. Horrocks, I., Patel-Schneider, P.F., Boley, H., Tabet, S., Grosof, B., Dean, M.: Swrl: A semantic web rule language combining owl and ruleml. W3c member submission, W3C (2004)
23. Owens, A., Seaborne, A., Gibbins, N., Schraefel, M.: Clustered tdb: A clustered triple store for jena. In: WWW 2009 (November 2008)
24. Prud'hommeaux, E., Seaborne, A.: SPARQL query language for RDF. W3C recommendation, W3C (January 2008)
25. Birbeck, M., Adida, B.: RDFa primer. W3C note, W3C (October 2008)
26. Connolly, D. (ed.): Gleaning Resource Descriptions from Dialects of Languages (GRDDL). W3C recommendation, W3C (January 2007)

# Behavior of Network Applications during Seamless 3G/WLAN Handover

Nickolay Amelichev, Mikhail Krinkin, and Kirill Krinkin

Open Source & Linux Lab, St.-Petersburg Electrotechnical University,
5 Prof. Popova St., St.-Petersburg, Russia
namelichev@acm.org, krinkin.m.u@gmail.com, kirill.krinkin@fruct.org
http://osll.fruct.org

**Abstract.** 3G- and WLAN-capable mobile devices supporting 3GPP Release 8 can perform seamless handover between 3G and WLAN. IP address of the device then stays constant, whereas the physical network interface changes. Behavior of common network applications (downloads, buffered video, streaming audio, VoIP) in such situation is tested using a Nokia N900 device. Network delay estimates and TCP RTT graphs are obtained. Common measures to mitigate problems induced by handover are briefly discussed.

**Keywords:** application testing, seamless handover, Nokia N900, 3G, WLAN.

## 1 Introduction

Offloading 3G traffic through WLAN to deal with increasing traffic volumes is a compelling solution for service providers. Public WLANs are almost ubiquitous and support for 3G/WLAN handover is available in a number of popular *User Equipment* (UEs), including Nokia N900. For traffic offloading to be effective, handover must be *seamless*: ideally, applications should not notice it.

Seamless 3G/WLAN handover using DSMIP or PMIPv6 mobility protocol was first specified in 3GPP Release 8 (specifically, in [1]). PMIPv6 protocol [4] is most promising: it was shown to cause small handover delay and have little signalling overhead [3,6,7]. Some of its modifications eliminate packet loss during handover [2], or significantly reduce both packet loss and handover latency [5].

In this paper, we present results of testing seamless handover from the UE perspective, i.e., testing application behavior on the device during handover. For testing, we used a vastly simplified PMIPv6-like handover procedure.

- Standalone server with static IP acted both as *mobile access gateway* (the host providing Internet connectivity to the device) and *local mobility anchor* (the host routing data to and from the UE).
- Physical network interface and routing table of the device and gateway was changed, while its IP address was preserved.

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 575–586, 2011.

Various network applications experiencing periodic 3G↔WLAN handovers were tested. We assessed their ability to "survive" handover (work well after handover) and estimated the network delay incurred by handover.

3GPP specification [1] provides several options for handover. We assumed a *Trusted WLAN*, as this is a simpler case (no authentication, authorization and accounting; no encryption). Furthermore, no access point was used (just ad-hoc WLAN).

## 2 Handover Testbed

To test real-life application behavior during periodically performed 3G/WLAN handovers, we developed a simple testbed. The testbed contained of a Mobile Node *MN*[1] connecting to Internet server(s) through either 3G or WLAN. Internet access was available through the gateway *GW*, which also simulated the mobile operator's network gateway.

The main requirement for the testbed was that the applications had to use *the same IP address regardless of the current access network* (3G or WLAN).

Testbed architecture, hardware and software configurations are described in detail in the following sections.

### 2.1 Architecture

WLAN interfaces *wlan0* were configured to run in Ad-Hoc mode with static IPs on both MN and GW.

Because MN had a dynamic IP on the 3G interface *gprs0*, we created a tunnel between MN and GW. Tunnel interfaces *tun0* on both MN and GW were assigned the same static IPs as the WLAN interfaces.

Creating tunnel was possible, because GW had a public static IPv4 address. (If this scheme is implemented by the mobile operator, creating tunnel is not necessary.)

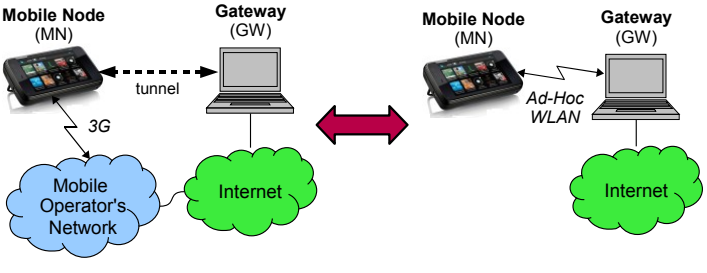



**Fig. 1.** 3G↔WLAN Handovers in a Testbed

[1] The term "Mobile Node" means the same as User Equipment (UE).

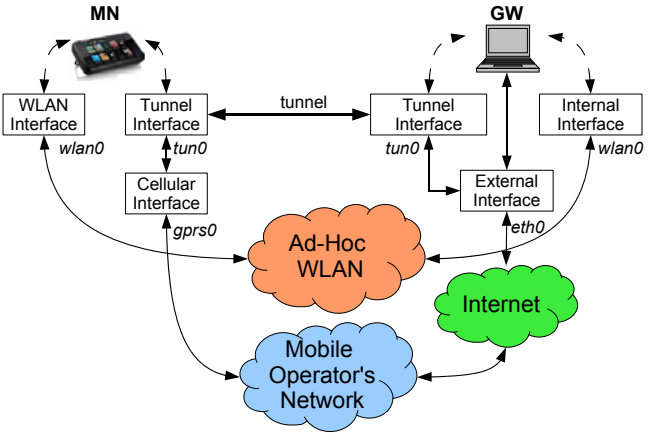



**Fig. 2.** Architecture of the testbed

**Table 1.** Interface States (Up/Down)

| Active Network | MN | | | GW | | |
|---|---|---|---|---|---|---|
| | *wlan0* | *tun0* | *gprs0* | *eth0* | *tun0* | *wlan0* |
| 3G | ↓ | ↑ | ↑ | ↑ | ↑ | ↓ |
| WLAN | ↑ | ↓ | ↓ | ↑ | ↓ | ↑ |

Two interfaces (*wlan0* and *tun0*) with the same IP address could not be simultaneously active, so handover (Fig. 1) was performed by switching them on and off.

The layout of testbed followed Fig. 2. Switching interfaces was done according to Table 1:

- During WLAN connectivity (*wlan0* up on both MN and GW) were up, traffic travelled along the path of MN.*wlan0* ↔ GW.*wlan0* ↔ GW.*eth0* ↔ Remote Server.*some_iface*.
- During 3G connectivity (*gprs0* and *tun0* up on MN, *tun0* up on GW), traffic travelled along the path MN.*tun0* ↔ MN.*gprs0* ↔ GW.*tun0* ↔ GW.*eth0* ↔ Remote Server.*some_iface*.

### 2.2 Hardware Configuration

We used Nokia N900 as the MN, and HP Pavilion dv6700 notebook as the GW. Relevant hardware configuration of the devices is summarized in Table 2.

MN was connected to MegaFon[2] 3.5G (HSDPA) network using unlimited data plan with throttling (max. DL 7.2 Mbit/s; 64 Kbit/s after 10Gb of traffic). GW was connected to the Internet through NNZ-Home[3] local provider with guaranteed 25 Mbit/s (UL/DL).

[2] http://szf.megafon.ru
[3] http://nnz-home.ru

**Table 2.** Testbed: Hardware Configuration

| Parameter | Mobile Node | Gateway |
|---|---|---|
| Device Name | Nokia N900 | HP Pavilion dv6700 |
| Product | `RX-51` | `KU121EA#ACB` (`dv6840er`) |
| Hardware Revision | hw-build `2101` | sys. board `30D2` |
| Processor | ARM Cortex-A8 @700Mhz | Intel Core2 Duo T5850 @2Ghz |
| RAM | 256 Mb | 2 Gb |
| Network Adapters | WLAN: 10/2 Mbit/s (UL/DL) | WLAN: 11 Mbit/s (UL/DL) |
| | WCDMA: 384 Kbit/s (UL/DL) | Ethernet: 100 Mbit/s (UL/DL) |

### 2.3 Software Configuration

Both MN and GW were Linux-based (Maemo 5 and Ubuntu 10.04, respectively). Detailed software configuration of the testbed is given in Table 3.

**Cron.** To perform repeated handovers from 3G to WLAN and back at predefined intervals on both MN and GW, we used *cron* to run handover scripts.

**OpenVPN.** We used OpenVPN to implement tunnel. It was secured using AES-256 in CBC mode with static key. VPN connection was configured with default keepalive setting `10,60` (ping every 10s, reconnect if no answer in 60s).

**OpenSSH.** We used OpenSSH to connect to the MN and start handover and configuration scripts, gather handover logs and perform troubleshooting. During testing, GW and MN were physically connected with USB cable. SSH over USB interfaces (*usb0*) was used to maintain MN shell on the GW.

**OpenNTPd.** To ensure that `cron` works with best precision, clocks on MN and GW were periodically synchronized by `ntpd`. On Ubuntu-based GW `ntpd` was preinstalled and already configured; for Maemo 5-based MN manual installation following instructions at[4] was performed.

## 3 Testing

### 3.1 Pre-test Actions

Before running all the tests, OpenVPN tunnel and ad-hoc WLAN configuration on both MN and GW was performed by *Preparation* scripts. Scripts did the following.

1. Started OpenVPN using appropriate configuration file with settings corresponding to Table 3.
2. Configured the `iptables` firewall to allow NAT.
3. Enabled IP forwarding using `net.ipv4.ip_forward` `sysctl` option.
4. Set `essid` to `sw3g_test`, and the `mode` of WLAN cell to `ad-hoc` using `iwconfig` utility.

[4] http://talk.maemo.org/showpost.php?p=518010&postcount=14

**Table 3.** Testbed: Software Configuration

| Parameter | Mobile Node | Gateway |
|---|---|---|
| **Installed Software** | | |
| Operating System | Maemo 5 (Fremantle) | Ubuntu 10.04 |
| | Linux `2.6.28-omap1` | Linux `2.6.32-30-686` |
| Firmware / BIOS Versions | `PR1.3` Global , nolo `1.4.14` | PhoenixBIOS `F.58` |
| Additional Packages | openssh `1:5.1p1-6.maemo5`[1] | openssh `5.8p1`[4] |
| | openvpn `5:2.1~rc20-3maemo3`[1] | openvpn `2.1.4`[4] |
| | cron `4.1`[2], wget `1.10.2-2osso3`[1] | |
| | wireless-tools `30~pre7-1.3maemo+0m5`[3] | |
| | openntpd `3.9p1-4-maemo8`[1] | |
| **IPv4 Configuration** | | |
| Network Interfaces | To 3G Operator *gprs0* | To Local ISP *eth0* |
| | `Dynamic IP` | `188.65.67.236` |
| | Ad-Hoc WLAN *wlan0* | Ad-Hoc WLAN *wlan0* |
| | `192.168.5.2` | `192.168.5.1` |
| | Tunnel for 3G *tun0* | Tunnel for 3G *tun0* |
| | `192.168.5.2` | `192.168.5.1` |
| | MN USB [5] *usb0* | GW USB [5] *usb0* |
| | `192.168.2.15` | `192.168.2.14` |
| Custom Rules | none | Allow NAT, Forwarding |
| **OpenVPN Configuration (tunnels)** | | |
| Role | Client | Server@`188.65.67.236:5000` |
| Gateway IP | `192.168.5.1` | `192.168.5.2` |
| Tunnel Type | IP (*tun*) | |
| Encryption | AES-256-CBC; Static Key | |
| Keepalive | ping every 10s, reconnect if no answer in 60s | |
| Other Options | Run as Daemon; Persist Tunnel Device; Persist Key | |

[1] From *extras* (http://repository.maemo.org/extras fremantle/free) repo.
[2] From *daylessday.org* (http://maemo.daylessday.org/repo chinook/user) repo.
[3] From *Nokia Maemo* (https://downloads.maemo.nokia.com/Packages) repo.
[4] From standard Ubuntu 10.04 repos.
[5] USB interfaces were used to control MN over SSH during the experiment.

### 3.2 Test Sequence (see Fig. 3)

**`crond` and `tcpdump`.** At the beginning of the test, `crond` was started on the GW and MN. Then, packet capture using `tcpdump` was initiated on the GW. Packet

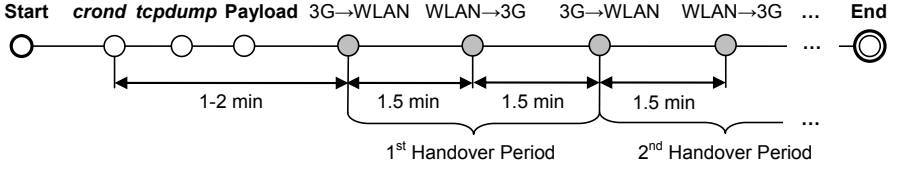



**Fig. 3.** Test Sequence

**Table 4.** Download Sources for FTP and HTTP Scenarios

| | URL | Size |
|---|---|---|
| FTP | `ftp://ftp.kernel.org/pub/linux/kernel/v2.6/linux-2.6.37.2.tar.gz` | 88.2M |
| | `ftp://largedownloads.ea.com/pub/demos/battlefield2standalonedemoserver.exe` | 35.0M |
| | `ftp://largedownloads.ea.com/pub/demos/MOHH/MEDALOFHONORHEROES.zip` | 26.2M |
| HTTP | `http://www.kernel.org/pub/linux/kernel/v2.6/linux-2.6.37.2.tar.gz` | 88.2M |
| | `http://ardownload.adobe.com/pub/adobe/reader/unix/9.x/9.4.0/enu/-AdbeRdr9.4-1_i386linux_enu.deb` | 60.4M |

trace included only packets sent from or to `192.168.5.0/24` network (from/to GW or MN). Trace had absolute timestamps (`-tttt` option of `tcpdump`.)

**Payload.** After `tcpdump`, the *payload* (actual application tested) was started on the MN.

**Handovers.** On every 3rd minute, `cron` launched handover scripts on both GW and MN. Scripts did the following:

1. Handover from 3G to WLAN (by switching the interfaces). On the MN, the route to `0.0.0.0/0` through gateway `192.168.5.1`, using interface `wlan0`, was added.
2. Sleep for 1.5 min.
3. Handover from WLAN to 3G (by switching the interfaces). On the MN, the route to `0.0.0.0/0` through `192.168.5.1` was changed to use `tun0` interface, and route to `188.65.67.236` using `gprs0` interface was added.

Logging of handover start and end times was also done, both on the GW and MN. This helped us identify handover intervals in the PCAP traces captured.

**Ending the Tests.** To end the test, `cron`, scenario payload, and `tcpdump` (in that order) were terminated on both GW and MN. Normal ending conditions for the scenarios are given in the "Testing Scenarios" section. Abnormal stop was declared when the application failed to "survive" handover (e.g. terminated VoIP call, did not resume playing media or downloading files).

### 3.3 Testing Scenarios

**FTP and HTTP Downloads.** `wget` utility was used to sequentially download a number of files totalling ≈150 Mb (Table 4) using FTP and HTTP. The scenario ended when all the files were successfully downloaded.

**Buffered Video.** Flash videos from *YouTube.com* and *vkontakte.ru* (Table 5) were played in the MicroB browser. The scenario ended when 9 handovers were reached (≈27 minutes have passed from the start).

**Internet Radio.** Two Internet radios were used: *16bit.fm* and *Wish FM* (Table 6). For *16bit.fm* we tested 2 bitrates (32 and 128 kbps) and formats (MP3 and AAC+). The scenario ended when 9 handovers were reached (≈27 minutes).

**Table 5.** Videos Used for Testing

| | URL | Resolution | Duration |
|---|---|---|---|
| YouTube | `http://youtu.be/--QTzeFCwtg` | 640 × 360 | 1:04:53 |
| vkontakte | `http://vkontakte.ru/video7085560_138716447` | 512 × 288 | 1:39:25 |

**Table 6.** Internet Radio Stations Used for Testing

| | URL | Bitrate | Format |
|---|---|---|---|
| 16bit.FM (RU) | `http://16bit.fm/16bit.fm_32.m3u` | 32 kbps | AAC+ |
| | `http://16bit.fm/16bit.fm_128.m3u` | 128kbps | MP3 |
| Wish FM (UK) | `http://stream1.radiomonitor.com/Wish.m3u` | 64 kbps | MP3 |

**Skype.** Skype call from *nick.amelichev* to *krinkin.m.u* was initiated. To simulate voice, music was played through external speakers of the GW, and microphone of the MN was placed near the speakers. The scenario ended when 9 handovers were reached.

### 3.4 Network Delay Estimation

After testing, we analyzed the network traces using Wireshark to estimate immediate delay in TCP transmission incurred by the each 3G→WLAN handover (Delay incurred by WLAN→3G handover was observed to be significantly lower during testing, so it was not considered in the estimation.)

1. Frames were filtered by protocol type (e.g. *http* for HTTP Downloads).
2. Filtered frames were sorted by timestamp offset $\Delta t$ from the previous *displayed* frame, descending.
3. For each 3G→WLAN handover time $T_{\text{Handover}}$, the first frame meeting all of the following conditions was chosen:
   - source and destination addresses used by the application tested;
   - the minimum absolute timestamp $T$ which is $> \Delta t + T_{\text{Handover}}$
4. If no such frame could be found, the handover period was marked as having no negative consequences ("—"). "—" periods were excluded from calculation of mean and standard deviation of $\Delta t$.

## 4 Discussion of Results

We observed that 3G↔WLAN handover highly impacts FTP RTT (retransmission time), causing peaks with RTT = tens of seconds, whereas the WLAN↔3G handover does not.

Most of the applications using buffering and/or timeouts (*FTP and HTTP Downloads*, *Buffered Video*) performed well: without failures and perceived delays.

*Internet Radio*, which also used buffering, performed well using 32kbps AAC+, but showed periodic delays when playing streaming 64- and 128kbps MP3 media. This might be caused by the insufficient buffer sizes, and less effectiveness of MP3 coding compared to AAC+.

*Skype* was the worst app in terms of handover survival: it repeatedly dropped calls at a random time after handover. Skype's network usage pattern is complex, and its protocols are not well-known, so it's not immediately apparent what could have caused its bad performance.

### 4.1 FTP and HTTP Downloads

`wget` successfully "survived" handover. Maximum network delays $\Delta t$ are given in Tables 7 and 8. Fig. 4 and 5 show high number of peaks in RTT caused primarily by 3G↔WLAN handovers.

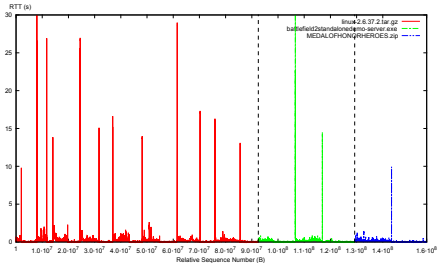



**Fig. 4.** TCP RTT for FTP Downloads

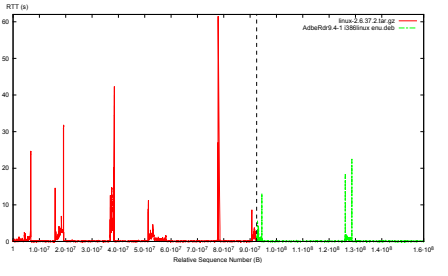



**Fig. 5.** TCP RTT for HTTP Downloads

**Table 7.** Network Delays for FTP Downloads

| # | $T_{\text{Handover}}$ | Delay $\Delta t$ (s) |
|---|---|---|
| 1 | 22:42:01 | 9.2840 |
| 2 | 22:45:01 | 14.2880 |
| 3 | 22:48:01 | 10.4280 |
| 4 | 22:51:01 | 14.5760 |
| 5 | 22:54:01 | 9.7960 |
| 6 | 22:57:01 | 9.3720 |
| 7 | 23:00:01 | 15.8000 |
| 8 | 23:03:01 | 11.492 |
| 9 | 23:06:01 | 10.256 |
| 10 | 23:09:01 | 14.02 |
| 11 | 23:12:01 | 9.596 |
| | Mean: | 11.7189 |
| | St. Dev.: | 2.4538 |
| | Min: | 9.2840 |
| | Max: | 15.8000 |

**Table 8.** Network Delays for HTTP Downloads

| # | $T_{\text{Handover}}$ | Delay $\Delta t$ (s) |
|---|---|---|
| 1 | 22:33:01 | 17.0320 |
| 2 | 22:36:01 | 12.7800 |
| 3 | 22:39:01 | 12.5400 |
| 4 | 22:42:01 | 10.1440 |
| 5 | 22:45:01 | 21.0200 |
| 6 | 22:48:01 | 13.0040 |
| 7 | 22:51:01 | 10.4400 |
| | Mean: | 13.8514 |
| | St. Dev.: | 3.8832 |
| | Min: | 10.1440 |
| | Max: | 21.0200 |

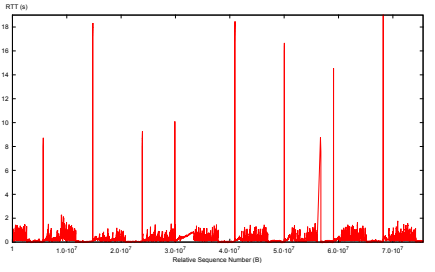



**Fig. 6.** TCP RTT for YouTube Video.

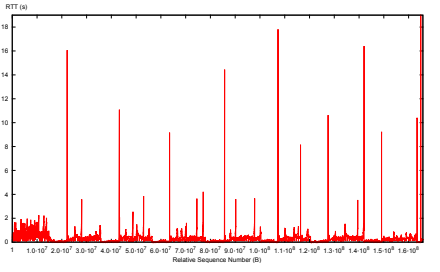



**Fig. 7.** TCP RTT for vkontakte Video.

**Table 9.** Network Delays for YouTube Video Playing

| # | $T_{\text{Handover}}$ | Delay $\Delta t$ (s) |
|---|---|---|
| 1 | 01:48:01 | 12.3920 |
| 2 | 01:51:01 | 15.2560 |
| 3 | 01:54:01 | 14.9520 |
| 4 | 01:57:01 | 69.8000 |
| 5 | 02:00:01 | 10.1040 |
| 6 | 02:03:01 | 9.4960 |
| 7 | 02:06:01 | 10.3920 |
| 8 | 02:09:01 | 11.4120 |
| 9 | 02:12:01 | 10.6720 |
| | Mean: | 18.2751 |
| | St. Dev.: | 19.4309 |
| | Min: | 9.4960 |
| | Max: | 69.8000 |

**Table 10.** Network Delays for vkontakte Video Playing

| # | $T_{\text{Handover}}$ | Delay $\Delta t$ (s) |
|---|---|---|
| 1 | 02:36:01 | 11.0360 |
| 2 | 02:39:01 | 12.7600 |
| 3 | 02:42:01 | 16.6880 |
| 4 | 02:45:01 | 12.7720 |
| 5 | 02:48:01 | 12.1160 |
| 6 | 02:51:01 | 16.7920 |
| 7 | 02:54:01 | 16.3960 |
| 8 | 02:57:01 | 10.1640 |
| 9 | 03:00:01 | — |
| | Mean: | 13.5905 |
| | St. Dev.: | 2.6596 |
| | Min: | 10.1640 |
| | Max: | 16.7920 |

### 4.2 Buffered Video

Both YouTube and vkontakte Flash video used HTTP protocol under the hood. Because of aggressive buffering (the buffering didn't stop until all the media was buffered; buffering readily resumed after handover), large network delays (see Tables 9 and 10) didn't affect the viewing experience at all. 9 (YouTube) and 8 (vkontakte) 3G↔WLAN handovers can be readily seen on the RTT Graphs (Fig. 6 and 7).

### 4.3 Internet Radio

Network delays estimated for *16bit.fm* in 32kbps AAC and 128kbps MP3 formats are given in Tables 11 and 12. The RTT graphs for 32 and 128kbps are given in Fig. 8 and 9. We could not obtain a reliable trace for *Wish FM* radio (64kbps MP3), because it repeatedly stopped playing after first 3G↔WLAN handover occurred.

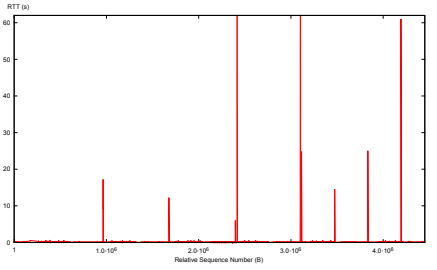



**Fig. 8.** TCP RTT for AAC+ 32kbps

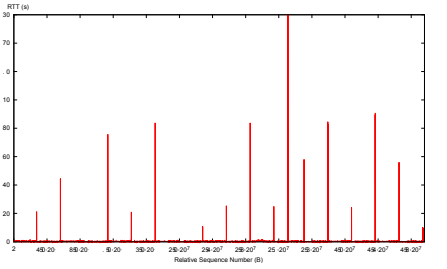



**Fig. 9.** TCP RTT for MP3 128kbps

**Table 11.** Network Delays for 16bit.fm (32 Kbps) Internet Radio

| # | $T_{\text{Handover}}$ | Delay $\Delta t$ (s) |
|---|---|---|
| 1 | 01:12:01 | 9.0000 |
| 2 | 01:15:01 | 14.3840 |
| 3 | 01:18:01 | 9.5240 |
| 4 | 01:21:01 | 10.4280 |
| 5 | 01:24:01 | 14.5080 |
| 6 | 01:27:02 | 39.8240 |
| 7 | 01:30:01 | — |
| 8 | 01:33:01 | — |
| 9 | 01:33:01 | — |
| | Mean: | 16.2780 |
| | St. Dev.: | 9.0000 |
| | Min: | 9.0000 |
| | Max: | 39.8240 |

**Table 12.** Network Delays for 16bit.fm (128 Kbit/s) Internet Radio

| # | $T_{\text{Handover}}$ | Delay $\Delta t$ (s) |
|---|---|---|
| 1 | 01:42:01 | 10.5600 |
| 2 | 01:45:01 | 11.8560 |
| 3 | 01:48:01 | 10.3520 |
| 4 | 01:51:01 | 12.9760 |
| 5 | 01:54:01 | 12.5160 |
| 6 | 01:57:01 | 12.4000 |
| 7 | 02:00:01 | 13.5280 |
| 8 | 02:03:01 | 11.9960 |
| 9 | 02:06:01 | 13.0520 |
| | Mean: | 12.1373 |
| | St. Dev.: | 1.0875 |
| | Min: | 10.3520 |
| | Max: | 13.5280 |

**Table 13.** Network Delays for Skype VoIP

| # | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| $T_{\text{Handover}}$ | 00:12:01 | 00:15:01 | 00:18:01 | 00:21:01 | 00:31:32 | 00:34:31 |
| $\Delta t$ (s) | 8.8200 | 8.6200 | 9.2680 | 8.8440 | 8.8440 | 8.4480 |
| Mean: 8.8073 | St. Dev.: 0.2754 | Min: 8.4480 | Max: 9.2680 | | | |

*16bit.fm* played without interruptions for 32kbps, and with minor interruptions for 128kbps. It seems that the buffer size used by the Nokia N900 media player was set too small when the network delay was small and almost uniform (128kbps stream), and grew too big when sudden large delay occurred (32kbps stream, after 6th handover).

### 4.4 Skype

Skype did not "survive" handover: the best we got is only 6 handovers, not 9 (Table 13). During testing, Skype sometimes terminated calls after handover and set status to "disconnected", only to reconnect to the network a moment later.

## 5 Conclusion

Network delay incurred by 3G↔WLAN seamless handover is large (tens of seconds). We observed different techniques used to deal with handover.

1. *Aggressive buffering* (buffering + timeouts). The app fills buffer with data until everything is buffered. Serving data from the buffer starts when a large part has already been downloaded. When network is unavailable, retrying is performed, typically with an exponential backoff timer. The method is commonly employed by applications using Flash video and is very effective in practice (see "Buffered Video").
2. Using *timeouts* only (if connection is gone, just wait for it). This is suitable if the data does not need to be served to the user immediately, e.g. downloading files. Test results for FTP and HTTP downloads using `wget` (see "FTP and HTTP Downloads") confirm that downloads survive handover well.
3. *Buffering* with the buffer size dependent on the last network delay time(s) experienced. As test results for streaming audio (see "Internet Radio") show, it is less effective than aggressive buffering, especially when the buffer size (in seconds) is ≈ network delay time.
4. *Other techniques.* It is not known what Skype uses, but it could not survive the handover (see "Skype").

To deal with delay incurred by the handover, applications should use aggressive buffering (if information is needed at the time it was requested) or timeouts. To partially minimize loss of packets and handover latency, buffering on the MN and GW [2] or a robust handover prediction scheme [5] might be implemented.

## 6 Future Work

MN OS could be modified to better mitigate handover consequences.

One solution is to make `send()`, `sendto()` and `recv()` socket calls block when called during handover. These calls will return only after handover is completed and the corresponding operation (sending or receiving) is carried out on the new access network.

Another solution would be for the MAC layer to notify TCP implementation when handover begins and ends. During handover, TCP retransmission timer would be "frozen" and the RTT values would stay constant. This way, the retransmission will start much sooner.

These topics will be uncovered in the upcoming publications.

**Acknowledgments.** The authors would like to thank Open Innovations Framework Program FRUCT (http://fruct.org) for provided support and Nokia Siemens Networks experts for their consultacy work. This work was done in Open Source and Linux Lab (http://osll.fruct.org).

## References

1. 3GPP 23.402 Technical Specifcation Group Services and System Aspects, Architecture enhancements for non-3GPP Accesses (Release 9) (2009)
2. Choi, H., Kim, K., Lee, L., Min, S., Han, Y.H.: Seamless Handover Scheme for Proxy Mobile IPv6 Using Smart Buffering. In: Proceedings of the International Conference on Mobile Technology, Applications and Systems (2008)
3. Fu, X., Lei, J.: Evaluating the Benefits of Introducing PMIPv6 for Localized Mobility Management. IFI-TB-2007-02. Tech. rep., Institute of Computer Science, University of Goettingen, Goettingen, Germany (2007)
4. Gundavelli, S., Leung, K., Devarapalli, V., Chowdhury, K., Patil, B.: Proxy Mobile IPv6. IETF RFC 5213 (2008)
5. Kim, G.: Low Latency Cross Layer Handover Scheme in Proxy Mobile IPv6 Domain. In: Balandin, S., Moltchanov, D., Koucheryavy, Y. (eds.) NEW2AN/ruSMART 2008. LNCS, vol. 5174, pp. 110–121. Springer, Heidelberg (2008)
6. Li, Y., Su, H., Su, L., Jin, D., Zeng, L.: A Comprehensive Performance Evaluation of PMIPv6 over IP-based Cellular Networks. In: Proceedings of Vehicular Technology Conference (2009)
7. Udugama, A., Fan, C.: Evaluation of a Network based Mobility Management Protocol: PMIPv6. In: Proceedings of Vehicular Technology Conference (2009)

# Building Programming Abstractions for Wireless Sensor Networks Using Watershed Segmentation

Mohammad Hammoudeh and Tariq A.A. Alsbou'i

Manchester Metropolitan University, Manchester, UK
m.hammoudeh@mmu.ac.uk

**Abstract.** The availability and quality of information extracted from Wireless Sensor Networks (WSNs) revolutionised a wide range of application areas. The success of any WSN application is, nonetheless, determined by the ability to retrieve information with the required level of accuracy, within specified time constraints, and with minimum resource utilisation. This paper presents a new approach to localised information extraction that utilises the Watershed segmentation algorithm to dynamically group nodes into *segments*, which can be used as programming abstractions upon which different query operations can be performed. Watershed results in a set of well delimited areas, such that the number of necessary operations (communication and computation) to answer a query are minimised. This paper presents a fully asynchronous Watershed implementation, where nodes can compute their local data in parallel and independently from one another. The preliminary experimental results demonstrate that the proposed approach is able to significantly reduce the query processing cost and time without involving any loss of efficiency.

## 1 Introduction

Wireless Sensor Networks (WSNs) are currently being employed in a variety of applications ranging from home to industry, and from health to military. These applications have a number of elements in common: (1) The request for information; (2) The answer to this request is usually present in a set of unstructured data streams; (3) WSNs generate large amount of data that is 'imperfect' in nature and contains considerable redundancy. Resource constraints on nodes in the network coupled with the characteristics of returned data means that applications have to be developed with the primary design goal of minimising resource utilisation. Distributed information extraction has been advocated to solve this kind of problems. Information extraction is the sub-discipline of artificial intelligence that selectively structures, filters, and merges data generated by one or more sensor nodes. It adds meaning to unstructured raw data; therefore, the data become structured or quasi-structured making it more suitable for information processing tasks. This definition is concise and covers exactly in what sense the term information extraction will be used throughout this paper.

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 587–597, 2011.

This paper presents an in-network information extraction system that finds and links relevant information while ignoring irrelevant and extraneous information. The final output of the extraction process varies based on user queries; however, it commonly involves the extraction of fragments of information from various nodes within the same segment and linking of these fragments into a coherent answer. We propose the utilisation of Watershed segmentation algorithm [20] that result in a set of well delimited segments of homogenous sensed regions based on nodes location and their corresponding sensor readings. Watershed algorithm is suitable for dynamically changing environments because it uses no thresholding, instead the best option is chosen at each decision stage. We also propose a parallel asynchronous Watershed algorithm implementation that complies with sensor node constraints, i.e. real-time computing, and low power consumption. This in-network information extraction does not return the entire collected data, but it extracts sense data units from one or more network segments, typically simple or multi-modal information of spatio-temporal nature. The goal of node segmentation is to provide a high-level programming model for sensor networks that abstracts away the details of individual sensor nodes. In this paper we examine query-based systems that utilise in-network processing for query response. However, the produced abstractions can be utilised in by other information extraction systems, i.e. event based and time-driven.

Query-based information extraction is a request-response interaction between the sensor nodes and end-user or application component. The end user issues a query in an appropriate language, and then the query is disseminated to the network to retrieve the desired data from the sensors based on the description in the query. Most query-based systems provide a high level interface to the sensor network while hiding the network topology as well as radio communication. The end user does not need to know how the data is collected or processed.

User controlled, query-based, information extraction is usually applied in situations where it is known in advance what type of semantic information is to be extracted from the network. For example, it might be necessary to identify what type of events are happening in a certain part of the monitored environment and at what time these events took place. Depending on the information needs, different queries can be constructed to differentiate various types of events at different levels of semantic granularity. In some applications, for instance, it will be adequate to specify that a part of a query is a temporal expression, while in others it might be necessary to differentiate between different temporal classes, for example between expressions indicating past, present and future. In other applications, not only the semantic nature of the target information is predefined, but also the unit and scope of the event to be extracted. The unit of extraction refers to the granularity of individual information portions that are lifted out of the sensor node. The scope of extraction refers to the granularity of the extraction space for every individual information request. In order to decide which portion of information is supposed to contribute in the answer of a query, an information extraction application uses a set of extraction conditions. These

conditions state what formal properties a particular portion of information must possess to belong to a particular semantic class.

The Watershed transformation is a popular image segmentation algorithm for grey scale images [20,12,7,3,2,5,17,14,19,21]. Its basic concept comes from the field of topography, referring to the partitioning of a landscape to a number of basins or water catchment areas. The authors in [12] use the following analogy to explain how the Watershed algorithm works: The USA can be divided into two main segments, one associated with the Atlantic Ocean and another associated with the Pacific Ocean. All the rain falling on the east segment will flow into the Atlantic Ocean, while the rain falling on the west segment will flow into the other ocean. The water will reach the ocean given that it is not trapped in a local minimum along the way. Both segments are usually named catchment basins, and each one has an associated minimum (the ocean). The boundary line that separates both basins is called the watershed line, corresponding to the continental divide in the example. Therefore, the image is viewed as a topographic surface where each pixel is a point situated at some altitude as a function of its grey level. The grey levels correspond to the altitude associated to the image between 0 and 255.

After the original Watershed algorithm was published, several modifications and variations of this algorithm was published to suit various applications. Watershed algorithms can be classified into two conceptually distinct techniques: immersion and raining.

1. Immersion simulates progressively immersing the entire topographic surface in a water container.
2. Raining simulates the rain fall over a topographic surface. The raining can be considered as a local method because each droplet follows on its own way not considering neighbouring droplets.

On flat areas of the surface, the motion of the water droplet is directed towards the nearest brim of a downward slope and it stops when it reaches a regional minimum. An important aspect of designing a parallel asynchronous algorithm is the exploitation of the data locality for minimisation of the communication overhead. Aiming at the goal, we propose here a reformulation of the raining watershed segmentation due to its suitability for parallel implementation. The presented implementation is capable of computing the Watershed transform according to local conditions. The approach proposed in this paper generates the same result using only one point of synchronisation, thus decreasing the running time without degradation in the effectiveness of the segmentation. Both assertions are demonstrated throughout this paper and supported by experimental results.

The paper is organised as follows: Section 2 describes the Watershed algorithm. Section 3 illustrates the suitability of this algorithm for WSNs. Section 5 presents the parallel asynchronous implementation of the Watershed algorithm. The performance of the proposed implementation is evaluated in Section 6. Section 7 concludes the work.

## 2 Description of the Watershed Algorithm

This section presents Meyer's formalism [9] that is based on local conditions (rain simulation). The Watershed proposed by Meyer is the base for our parallel asynchronous implementation. Roedink et al. [15] summarised approaches for parallel implementations of the Watershed algorithm on powerful (memory and computation) devices. Before our synchronous and parallel implementation details are given, some preliminary definitions are introduced as described by Meyer [9].

Let $f(p)$ be a function of of grey levels, signifying a continous digital image with the domain $\Omega \in \mathbb{Z}^2$. Every pixel $p \in \Omega$ has a grey level $f(p)$ and a set of neighbouring pixels $p' \in N(p)$ with a distance function $d(p, p')$ to each neighbour; in most published algorithms a 4− to 8−neighbours.

(Regional minimum) Is a point or a set of connected points with the same grey level, where none of them have a neighbour with a lower grey level [3]. In keeping with the rain simulation analogy, a regional minimum is the area of the surface where the rain water would get trapped without flowing to lower leveles. (Lower slope) If $p$ exists, the lower slope $(LS)$defines the maximum steepness from a pixel to its lower neighbours.

$$(LS) = max_{\forall p' \in N(p)\left(\frac{f(p)-f(p')}{dist(p,p')} | f(p') \leq f(p)\right)}$$

(Steepest decending path) $\forall_p \in \Omega, SDP(p)$ is the set of points $p' \in N(p)$ defined as follows:

$$\left\{ p' \in N(p) \left| \frac{f(p) - f(p')}{dist(p, p')} = LS(p), f(p') < f(p) \right. \right\}$$

i.e. the $SDP$ is a series of connected points where each point presents a grey level strictly lower than the previous one. There may exist multiple decending paths from a given point, the choice between them depends on the implementation. A descending path is said to be a steepest path if each point in the path is connected to the neighbour with the lowest grey level. According to rain simulation analogy, the $SDP$ is the path a drop of water would follow when travelling down to a regional minimum.

(Cost function based on lower slope) The cost, $cost(p_{i-1}, p_i)$, for walking on the topographic surface from point $p_{i-1}$to $p_i \in N(p_{i-1})$ is:

$$\begin{cases} LS(p_{i-1}).dist(p_{i-1}, p_i) & f(p_{i-1}) > f(p_i) \\ LS(p_i).dist(p_{p-1}, p_i) & f(p_{i-1}) < f(p_i) \\ \frac{1}{2}(LS(p_{i-1}) + LS(p_i)).dist(p_{i-1}, p_i) & f(p_{i-1}) = f(p_i) \end{cases}$$

(Topographic distance) The topographic distance between two points $p$ and $q$ on a surface is the minimal $\pi-$topographical distance among all paths $\pi$between $p$ and $q$on the suraface:

$$TD_f(p, q) = infTD_f^{\pi}(p, q)$$

where $TD_f(p,q) = \sum_{i=2}^{n} cost(p_{i-1}, p_i)$ is the topograpical distance of a path $\pi = (p = p_1, p_2, ..., p_n = q)$, such that $\forall_i, p_i \in N(p_{i-1})$ and $p_i \in \Omega$
(Catchment basin based on topographic distance) $CB_{TD}(m_i)$ of a local minimum $m_i$ is the set of points $p \in \Omega$ where the topographical distance is closer to $m_i$ than to any other regional minimum $m_j$, based on the topographical distance and the grey level of the minima:

$$CB_{TD}(m_i) = \{p \,|\, f(m_i) + TD_f(p, m_i) + TD_f(p, m_j) \,\forall j \neq i\}$$

In other words, the $CB$ is formed by a regional minimum and all the points whose steepest decsending path ends in that minimum [2]. According to the rain simulation analogy, a $CB$ is an area of a topographic surface, such that when a droplet of rain fall in any point in that area, it would poure in to its minimum following the steepest descending path of that point.

## 3 Modified Watershed Segmentation

**1- Neighbourhood definition using Shepard method.** In most published Watershed algorithm variations, a 4− to 8−pixel neighbours are used. Because the distance from a node to its neighbours can vary, we advocate the use Shepard method [16] to select nearby nodes. Shepard defined two criteria:
**(a) Arbitrary distance criterion:** All data points within radius $r$ of the point $p$ are included in computation.
**(b) Arbitrary number criterion:** Only the closest $n$ data points are considered in the computation of any interpolated value.

Shepard has chosen a mix of the two criteria which combined their advantages. An initial radius $r$ is defined depending on the overall density of data points such that seven data points are included on average in a circle of radius $r$. $r$ is written as follows:

$$\pi r^2 = \frac{7A}{N}$$

were $A$ is the area of the largest polygon enclosed by the data points. The suitability of this method for WSN is presented in [6].

**2- Inclusion of the number of hops in the calculation of the steepest path.** In Section 2, the SDP (*Definition* 3) was defined as a series of connected points where each point presents a grey level strictly lower than the previous one. There may exist multiple descending paths from a given point, the choice between them depends on the Watershed algorithm implementation. For WSNs applications, the most energy efficient descending path should be chosen. According to [13], communication is the most power hungry operation. Communication cost can be computed from transmission distance, hop count, delay, link quality, and other factors. Hop count is widely used factor to measure energy requirement of a routing task and for grouping nodes in energy efficient clusters. The power consumed in data transmission is directly proportional to the square of

the transmission distance between the sending and receiving nodes. The power consumption does not only depend on the transmission distance, but also on the scale of the network [18]. Therefore, the lower distances must be computed as a function of the number of hops as well as the inter-sensor Euclidean distances. The new distance function is the sum of the individual inter-node Euclidean distances multiplied by the hop count.

$$Dist(p, p') = \frac{\sum dist(p_i, p_j)}{hc}$$

where $i, j < hc - 1$ and $hc$ is the hope count.

## 4 Abstractions for Local Interactions

In this section, we give a brief description of how Watershed segmentation algorithm can be used for defining abstractions for local interactions and place our work in the context of other existing work in the area. To demonstrate the usefulness of the proposed abstractions we consider events that are caused by multiple elements targets, e.g. a herd of animals in a habitat monitoring application or toxic gas diffusion. Elements triggering these events often exhibit idiosyncratic behaviour, such as splitting into several groups or merging into a single group. This type of events involves several neighbouring sensor nodes to collaborate to acquire aggregate features of the target such as shape, location, coverage, moving direction, speed, etc. Events involving this kind of targets are submitted by [8] as 'region-like' targets. In this paper, Watershed segmentation is proposed to build a cooperative node structure over every network segment covered by the events to collect and aggregate related information.

Watershed groups nodes sharing some common group state into segments. A segment is described as a collection of spatially distributed nodes with an example being the set of nodes in a geographic area with sensor readings in a specific range. Therefore, Watershed segmentation combines the advantages of data-centric (e.g. [4] and topologically (e.g. [22]) defined group abstractions. This makes the generated segments capable of expressing a number of local behaviours powerfully, which allows network programmers to write programs that express higher level behaviour beyond that of the usual query-based methods. In fact, the programmer is concerned not only with the application logic, but also with identifying the network portions to be involved in extracting a particular piece of information and how to reach them. Dealing with this new requirement necessitates new programming abstractions to localise complexity without scarifying efficiency. The logical segments of nodes generated by the Watershed algorithm replace the tight physical neighbourhood provided by wireless broadcast with a higher level, application defined abstractions. Segments are created such that the span of a logical neighbourhood is specified dynamically and declaratively based on the attributes of nodes, along with requirements about communication costs (specifically the diameter of the segment). A network segment formed with a logical notion of proximity determined by applicative information is, therefore, capable to return specified information with high confidence.

Segments generated by Watershed algorithm are automatically labelled with a marker, which is an integer identifier. All nodes within the segment satisfy the logical constraints encoded in the Watershed algorithm. This logical and intuitive Watershed template serves as the membership function that dynamically determines and updates which nodes belong to the segment. Programmers manipulate segments instead of nodes within communication range. The programmers can still reason in terms of nodes and broadcast messages, but now they can specify declaratively which portions of the network to consider and therefore control the span of communication to save energy.

Macroprogramming [22] has been put forward in the literature as an efficient approach to information extraction that provides a more general-purpose approach to distributed computation. Many macroprogramming approaches aim at programming the network as a whole rather than programming the individual nodes that compose the network. Global behaviour can be specified, programmed and then translated to node level code transparently from low level details like network topology, radio communication or power capacity. A significance class of macroprogramming systems are the application-defined, in-network abstractions that are used in data processing. The Regiment [22] and Hood systems are examples of neighbourhood-based abstractions that handle many nodes collectively and a set of operations on it to enable the programmer to extract information about the state of the group. EnviroTrack [1] is a programming abstraction specifically for target-tracking applications, where a group is defined as the set of sensors that detected the same event. In SPIDEY [10], a node is represented as a logical node that has multiple exported attributes (static and dynamic). However, utilising the network topology as an abstraction can require some rigidity in the programming model. It can also be inefficient for systems with mobile nodes due to the cost and complexity of maintaining the mapping between the physical topology and the logical topology. The parallel Watershed-based abstractions are different from these approaches in one important aspect; segmentation runtime loosely synchronises state across nodes, attaining grater robustness and higher efficiency.

## 5 Parallel Asynchronous Watershed Implementation

Each node, $(x, y, z)$, is depicted as a pixel on the image and the colour of the pixel located at $(x, y)$ is obtained as the node sensed modality $(z)$. The resolution of the image is the total number of nodes in the network per area unit. In parallel operation, rain falling simulation Watershed uses less communication between nodes, compared to immersion which has a highly global nature. The new segmentation algorithm can be viewed as an asynchronous relaxation of the 'Hill Climbing' algorithm [9]. All processing is local to each node, which runs a simple finite state machine associated with a single sensed modality. Non-blocking communications allow each node to run independently. Since synchronisation is limited to $N(u)$ and non-blocking communications are used, each node operates independently and the algorithm needs no global scheduling.

The role of the Watershed process is to label each non-minimum node by walking downward on a steepest slope path towards the minimum. Initially, all nodes in the network are considered as non-minima and will be flooded from different sub-domains. At this stage of the segmentation process, nodes are assigned temporary labels because the segmentation results depends on neighbouring nodes readings. When a node detects a steepest neighbour, it changes it status to a non-minimum node and gets labelled from the steepest neighbour or from its predecessor that has the shortest distance among its neighbouring nodes. When the minimum is reached, its label is assigned to all the nodes upward along the path. If no non-minimum were detected, the steepest distances must be calculated based on the entire set of lower borders. Process termination is locally detected on each node. This reduces the amount of communication and the idle time in nodes.

The algorithm builds several paths with different origins and destinations from data communications between nodes. This does not introduce additional cost due to the broadcast nature of wireless communications. Typically, any changes in one node readings may affect all nodes on the steepest slope line between that node and the minimum. Therefore, nodes must keep monitoring and updating their membership. A flag, called *reset*, is maintained by each node to record whether changes occurred or not since the last communication. One advantage of this implementation is that no relabeling and no synchronisation between nodes are needed frequently.

One disadvantage of this parallel asynchronous implementation is that the middle nodes on plateaus cannot locally decide whether they are on a minimum or non-minimum plateau, $(NP)$, and necessitate global synchronisation points to identify and label the minimum plateaus. To avoid global synchronisation, all middle nodes are labelled as *minimum* or *Plateau,* $(MP)$, to allow them makes local decisions. After that, the propagation of data over a non-minimum plateau allows the middle nodes of that plateau flooded by a neighbour to switch to non-minimum.

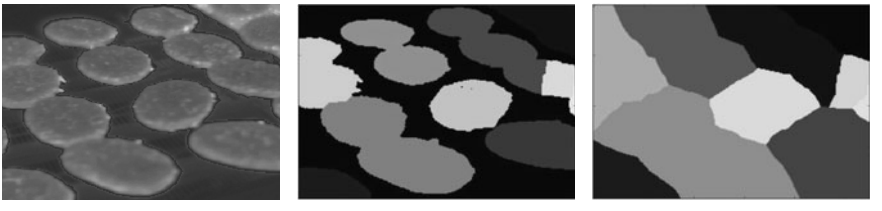
Algorithm 1 presents the pseudocode of the segmentation process in each node.

## 6 Evaluation

Suppose that a WSN has been deployed to monitor temperature of environmentally sensitive areas. An event of interest is predefined if temperature readings with enough numbers go above a certain threshold in a specific geographic area. In our simulation, an event is triggered at random times in random locations followed by issuing a query to locate the hottest spots. Query resolution is implemented using three methods: Watershed-based, nodes are logically grouped into segments that are used to assist query processing; in-network processing via aggregation of messages up a spanning tree of the network; and centralised, were all data is sent directly to the sink for analysis. All simulations were carried out using Dingo [11], which is a scalable python-based package to allow rapid

**Algorithm 1.** The segmentation process of nodes in each state.

**Input** node current state of node $(u)$ $S(u)$; $f(v)$ states of neighbours
**Output** segment membership
**case** $(S(u) =' initial)$
The node $u$ broadcasts its data $d(u)$ and label $l(u)$ to its neighbours $N(u)$, waits for data from each neigbhour $v \in N(u)$, compute the following:
$N(u)^{=}$ all neighbouring readings equal to $(u)$
$N(u)^{\geqslant}$ all neighbouring readings greater than $(u)$
$N(u)^{Ln} = v_i$ a singleton set such that $LS_{max}(v_i)$ and $v_i \in N(u)$
**if** $N(u)^{Ln} = \oslash$
$l(u) \leftarrow min_{v \in N(u)^{=}}(l(v))$; $S(u) \leftarrow' MP'$
**else**
$S(u) \leftarrow' NP'$
**case** $(S(u) =' MP')$
The node $u$ waits to receive new data, $d'(v)$, from any neighbour
**if** $d'(v) < d(u)$
$N(u)^{ln} \leftarrow v$; $S(u) \leftarrow' NP'$
**else**
$l(u) \leftarrow min(l(u), l(v))$; $S(u) \leftarrow' MP'$
**case** $(S(u) =' NP')$
The node $(u)$waits for data from $N(u)^{Ln}$; $S(u) \leftarrow' NP'$
In all states, the node $(u)$ sends its data $f(u)$ to all nodes in $N(u)^{\geq}$ whenever a new reading is recorded

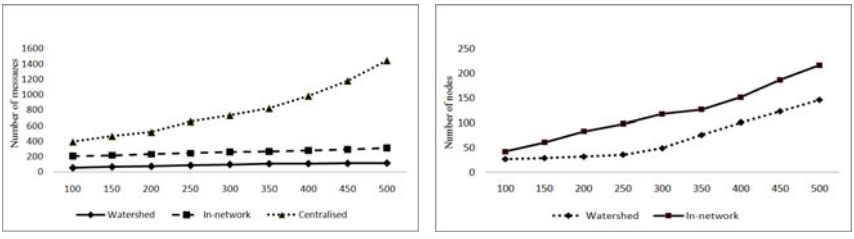


$(a)$ $(b)$ $(c)$

**Fig. 1.** $(a)$ Original thermal map $(b)$ Watershed segmentation results $(c)$ Extend segmentation

prototyping of WSNs algorithms. In all experiments, we make use of a thermal map, Figure 1$(a)$, adopted from goinfrared.com. Figure 1$(b)$ shows the result of segmentation in which nodes in each segment collaborate to solve a query. It is easy to extend the segmentation to the whole monitored terrain by using generalised Voronoi, i.e., each location where there is no sensor node is assigned to its nearest segment, Figure 1$(c)$.

The cost of the query resolution was measured in terms of the number of messages exchanged to answer the query. Multiple runs with different topologies




(*a*) (*b*)

**Fig. 2.** Communication overhead and the number of nodes involved in resolving the query

and different number of nodes were carried out for the three query resolution methods. These results are presented in Figure 2(*a*). The energy cost, response time, and accuracy are affected by which and how many nodes are involved in answering a query. Excluding nodes irrelevant to a certain query not only improve answer accuracy but also saves energy and reduce the time required for data analysis. Figure 2(*b*) shows the number of nodes involved in resolving the same query at different network densities.

The results obtained in the above experiments indicate that segments can be used to support in-network query resolution. The results shows that the communication overhead associated with segment-based query processing is almost 2 folds less than in-network processing and 10 folds less than the centralised query resolution. These results are clearly explained by the analysis presented in Figure 2.

## 7 Conclusion

The preliminary work in this paper indicates that segment-based in-network query processing produces considerable energy savings over aggregation and centralised approaches. Watershed logically organises nodes into energy efficient segments that reduce unecessary data transmissions and improve response accuracy. Compared to standard Watershed segmentation algorithms, the major improvement of our algorithm is that labelling and climbing along the steepest paths are concurrently and locally executed based on the node state, during the entire segmentation process. There are a number of limitations in the work so far that need to be addressed in the future, for example the cost of segmentation.

## References

1. Abdelzaher, T., Blum, B., Cao, Q., Chen, Y., Evans, D., George, J., George, S., Gu, L., He, T., Krishnamurthy, S., Luo, L., Son, S., Stankovic, J., Stoleru, R., Wood, A.: Envirotrack: Towards an environmental computing paradigm for distributed sensor networks. In: Proceedings of the 24th International Conference on Distributed Computing Systems (ICDCS 2004), pp. 582–589 (2004)
2. Bieniek, A., Moga, A.: An efficient watershed algorithm based on connected components. Pattern Recognition 33(6), 907–916 (2000)

3. Bleau, A., Leon, L.J.: Watershed-based segmentation and region merging. Comput. Vis. Image Underst. 77, 317–370 (2000)
4. Chu, M., Liu, J.J.: State-centric programming for sensor and actuator network systems. IEEE Pervasive Computing (2003)
5. Grau, V., Mewes, A.U.J., Alcaniz, M., Kikinis, R., Warfield, S.K.: Improved watershed transform for medical image segmentation using prior information. 23(4), 447–458 (2004)
6. Hammoudeh, M., Newman, R., Dennett, C., Mount, S.: Interpolation techniques for building a continuous map from discrete wireless sensor network data. In: Wireless Communications and Mobile Computing (January 2011)
7. Kuo, C.J., Odeh, S.F., Huang, M.C.: Image segmentation with improved watershed algorithm and its fpga implementation. In: IEEE ISCAS 2001, vol. 2, pp. 753–756 (2001)
8. Lin, C.-H., King, C.-T., Hsiao, H.-C.: Region abstraction for event tracking in wireless sensor networks. In: 8th International Symposium on Parallel Architectures Algorithms and Networks, pp. 274–281 (2005)
9. Meyer, F.: Topographic distance and watershed lines. Signal Process 38, 113–125 (1994)
10. Mottola, L., Picco, G.P.: Using logical neighborhoods to enable scoping in wireless sensor networks. In: Proceedings of the 3rd International Middleware Doctoral Symposium (2006)
11. Mount, S.: Dingo wireless sensor networks simulator (2011), http://code.google.com/p/dingo-wsn/ (accessed March 26, 2011)
12. Osma-Ruiz, V., Godino-Llorente, J.I., Sáenz-Lechón, N., Gómez-Vilda, P.: An improved watershed algorithm based on efficient computation of shortest paths. Pattern Recogn. 40, 1078–1090 (2007)
13. Pottie, G.J., Kaiser, W.J.: Wireless integrated network sensors. Commun. ACM 43(5), 51–58 (2000)
14. Rambabu, C., Rathore, T.S., Chakrabarti, I.: A new watershed algorithm based on hillclimbing technique for image segmentation. 4, 1404–1408 (2003)
15. Roerdink, Meijster: The watershed transform: Definitions, algorithms and parallelization strategies. FUNDINF: Fundamenta Informatica 41 (2000)
16. Shepard, D.: A two-dimensional interpolation function for irregularly-spaced data. In: Proceedings of the 1968 23rd ACM National Conference, pp. 517–524 (1968)
17. Sun, H., Yang, J., Ren, M.: A fast watershed algorithm based on chain code and its application in image segmentation. Pattern Recogn. Lett. 26, 1266–1274 (2005)
18. Sun, P., Seah, W.K.G., Lee, P.W.Q.: Efficient data delivery with packet cloning for underwater sensor networks. In: Symposium on Underwater Technology and Workshop on Scientific Use of Submarine Cables and Related Technologies, pp. 34–41 (April 2007)
19. Świercz, M., Iwanowski, M.: Fast, parallel watershed algorithm based on path tracing. In: Proceedings of the 2010 International Conference on Computer Vision and Graphics: Part II, ICCVG 2010, pp. 317–324 (2010)
20. Vincent, L., Soille, P.: Watersheds in digital spaces: An efficient algorithm based on immersion simulations. IEEE Transactions on Pattern Analysis and Machine Intelligence 13, 583–598 (1991)
21. Wagner, B., Dinges, A., Müller, P., Haase, G.: Parallel volume image segmentation with watershed transformation. In: Salberg, A.-B., Hardeberg, J.Y., Jenssen, R. (eds.) SCIA 2009. LNCS, vol. 5575, pp. 420–429. Springer, Heidelberg (2009)
22. Welsh, M., Mainland, G.: Programming sensor networks using abstract regions. In: Proceedings of the 1st Conference on Symposium on Networked Systems Design and Implementation, vol. 1, p. 3 (2004)

# Complexity Analysis of Adaptive Binary Arithmetic Coding Software Implementations

Evgeny Belyaev[1], Anton Veselov[2], Andrey Turlikov[2], and Liu Kai[3]

[1] Tampere University of Technology, Korkeakoulunkatu 10, 33720 Tampere, Finland
evgeny.belyaev@tut.fi
[2] Saint-Petersburg State University of Aerospace Instrumentation, Bolshaya Morskaya 67, 190000 St. Petersburg, Russia
{felix,turlikov}@vu.spb.ru
[3] School of Computer Science and Technology, Xidian University, NO.2 Taibai South Road, MailBox 161, 710071 Xi'an, China
kailiu@mail.xidian.edu.cn

**Abstract.** This paper is dedicated to the complexity comparison of adaptive binary arithmetic coding integer software implementations. Firstly, for binary memoryless sources with known probability distribution, we prove that encoding time for arithmetic encoder is a linear function of a number of input binary symbols and source entropy. Secondly, we show that the byte-oriented renormalization allows to decrease encoding time up to 40% in comparison with bit-oriented renormalization. Finally, we study influence of probability estimation algorithm for encoding time and show that probability estimation algorithm using "Virtual Sliding Window" has less computation complexity than state machine based probability estimation algorithm from H.264/AVC standard.

**Keywords:** binary arithmetic coder, range coder, probability estimation, complexity analysis.

## Introduction

Adaptive binary arithmetic coding is included in well known image and video compression standards and state of the art codecs like JPEG [1], JPEG2000 [2], H.264/AVC [3], Dirac [4] etc. Arithmetic coders implemented in these codecs are based on Q-coder [5] which is multiplication free adaptive binary arithmetic coder with bit renormalization and look-up tables used for probability estimation. Q-coder was introduced in 1988 and since that time the relative computational complexity of different arithmetical operations changed significantly. For example, table look-up operation takes more CPU clock cycles than a multiplication [7]. Thus, these changes should be considered for designing of a new video compression standards (especially for High Efficiency Video Coding (HEVC) [8]) or state of the art codecs.

In this paper we compare adaptive binary range coder introduced in [6] with arithmetic coder from H.264/AVC standard. At first, we show that the

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 598–607, 2011.

byte-oriented renormalization allows to decrease encoding time up to 40% in comparison with bit-oriented renormalization. Then we investigate the influence of probability estimation algorithm for encoding time and show that using look-up table free "Virtual Sliding Window" (VSW) algorithm [14] allows to decrease the encoding time up to 10% in comparison with probability estimation algorithm from H.264/AVC standard.

Other actual topic for arithmetic encoding is complexity or power consumption modeling which is needed for power-rate-distortion analysis. Previous works [9,10] use assumption that entropy encoder complexity is approximately proportional to the output bit rate. In this paper we prove that in case of binary memoryless sources with known probability distribution encoding time for binary arithmetic encoder is a linear function of a number of input binary symbols and source entropy. If probability is not known, then encoding time is a linear function of a number of input binary symbols and output bit rate.

The rest of this paper is organized as follows. Section 1 describes the main idea of arithmetic encoding and its two integer implementations with different renormalizations. Section 2 is dedicated to integer implementations of probability estimation based on sliding window approximations. Section 3 introduces the linear model for complexity of binary arithmetic encoder and show the comparative results for described adaptive binary arithmetic coding software implementations.

## 1 Integer Implementations of Binary Arithmetic Encoder

Let us consider stationary discrete memoryless binary source with ones probabilities $p$. In binary arithmetic encoding codeword for sequence $\mathbf{x}^N = \{x_1, x_2, ..., x_N\}$, $x_i \in \{0, 1\}$ is represented as $\lceil -\log_2 p(\mathbf{x}^N) + 1 \rceil$ bits of number

$$\sigma(\mathbf{x}^N) = q(\mathbf{x}^N) + p(\mathbf{x}^N)/2, \tag{1}$$

where $p(\mathbf{x}^N)$ and $q(\mathbf{x}^N)$ are probability and cumulative probability of sequence $\mathbf{x}^N$ accordingly which can be calculated by using following recurrence formulas. If $x_i = 0$, then

$$\begin{cases} q(\mathbf{x}^i) \leftarrow q(\mathbf{x}^{i-1}) \\ p(\mathbf{x}^i) \leftarrow p(\mathbf{x}^{i-1}) \cdot (1-p). \end{cases} \tag{2}$$

If $x_i = 1$, then

$$\begin{cases} q(\mathbf{x}^i) \leftarrow q(\mathbf{x}^{i-1}) + p(\mathbf{x}^{i-1}) \cdot (1-p) \\ p(\mathbf{x}^i) \leftarrow p(\mathbf{x}^{i-1}) \cdot p. \end{cases} \tag{3}$$

In practice, integer implementation of arithmetic encoder is based on two $v$-size registers: `low` and `range` (see Algorithm 1). Register `low` corresponds to $q(\mathbf{x}^N)$, register `range` corresponds to $p(\mathbf{x}^N)$. The precision required to represent registers `low` and `range` grows with the increase of $N$. For decreasing coding latency and avoiding registers underflowing the *renormalization* procedure is used for each output symbol (see lines 8–26 in Algorithm 1).

As an alternative to arithmetic coders, *range coders* use bytes as output bit stream element and do byte renormalization at a time [16,17,18] (see lines 8–15 in Algorithm 2). In this paper the binary range coder with Subbotin's [19] renormalization is analyzed.

**Algorithm 1.** Binary symbol $x_i$ encoding procedure in binary arithmetic coder

**Input:** $x_i$, low, range, counter

```
 1: len := range·p
 2: len := max{1,len}
 3: range := range − len
 4: if x_i =1 then
 5:    low := low+range
 6:    range := len
 7: end if
 8: while range < QUARTER do
 9:    if low ≥ HALF then
10:       PUTBIT(1)
11:       for i=1,...,counter do
12:          PUTBIT(0)
13:       end for
14:       low := low − HALF
15:    else if low < QUARTER then
16:       PUTBIT(0)
17:       for i=1,...,counter do
18:          PUTBIT(1)
19:       end for
20:    else
21:       counter := counter + 1
22:       low := low − QUARTER
23:    end if
24:    low := low/2
25:    range := range/2
26: end while
```

## 2 Integer Implementations of Probability Estimation

### 2.1 Sliding Window and Its Approximations

Algorithms of adaptive data encoding based on *sliding window* are widely known. The probability of source symbol is estimated by analysis of special buffer contents [11]. It keeps $W$ previous encoded symbols, where $W$ is the length of the buffer. After encoding of each symbol the buffer's content is shifted by one position, new symbol is written to the free cell and the earliest symbol in buffer is erased. This buffer is called sliding window after the method of buffer content manipulation.

**Algorithm 2.** Binary symbol $x_i$ encoding procedure in binary range coder

**Input:** $x_i$, low, range

```
 1: len := range·p
 2: len := max{1,len}
 3: range := range − len
 4: if x_i =1 then
 5:    low := low+range
 6:    range := len
 7: end if
 8: while (low ⊕ (low+range))<TOP ∨ (range<BOTTOM) do
 9:    if range<BOTTOM ∧ ((low ⊕ (low+range)))≥TOP then
10:       range:= −low ∧ BOTTOM−1
11:    end if
12:    PUTBYTE(low·2^−24)
13:    range:=range·2^−8
14:    low:=low·2^−8
15: end while
```

For binary sources probability of ones is estimated by Krichevsky-Trofimov [12] formula

$$\hat{p}_{t+1} = \frac{S_t + 0.5}{W + 1}, \tag{4}$$

where $S_t$ is the number of ones in the window before encoding symbol with the number $t$.

The advantage of using the sliding window is the opportunity of precise evaluation of source statistics and fast adaptation to changing statistics. However, the window has to be stored in the encoder and decoder memory, which is a serious disadvantage of this algorithm. To avoid it the *Imaginary Sliding Window* technique (ISW) proposed for a binary source [13] and for non-binary source [11]. The ISW technique does not require window content storage and estimates count of symbols from source alphabet stored in the window.

Let us consider the ISW method for a binary source. Define $x_t \in \{0, 1\}$ as source input symbol with number $t$, $y_t \in \{0, 1\}$ as symbol deleted from the window after addition of $x_t$. Suppose at every time instant a symbol in a random position is erased from the window instead of the last one. Then the number of ones in the window is recalculated by the following recurrent randomized procedure.

**Step 1.** Delete a random symbol from the window

$$S_{t+1} = S_t - y_t, \tag{5}$$

where $y_t$ is a random value generated with probabilities

$$\begin{cases} Pr\{y_t = 1\} = \dfrac{S_t}{W}, \\ Pr\{y_t = 0\} = 1 - \dfrac{S_t}{W}. \end{cases} \tag{6}$$

**Step 2.** Add a new symbol from the source

$$S_{t+1} = S_{t+1} + x_t. \tag{7}$$

For implementation of ISW algorithm a random variable must be generated. This random variable should take the same values at the corresponding steps of encoder and decoder. However, there is a way to avoid generating a random variable [14]. At step 1 of the algorithm let us replace random value $y_t$ with its probabilistic average. Then the rule for recalculating number of ones after encoding of each symbol $x_t$ can be presented in two steps.

**Step 1.** Delete an average number of ones from the window

$$S_{t+1} = S_t - \frac{S_t}{W}. \tag{8}$$

**Step 2.** Add a new symbol from the source

$$S_{t+1} = S_{t+1} + x_t. \tag{9}$$

By combining (8) and (9), the final rule for recalculating number of ones can be given as follows:

$$S_{t+1} = \left(1 - \frac{1}{W}\right) \cdot S_t + x_t. \tag{10}$$

### 2.2 Probability Estimation Based on State Machine

One way for implementation of probability estimation can be based on the *state machine* approach. Each state of this machine corresponds to some probability value. Transition from state to state is defined by the value of the input symbol. This approach does not require multiplications or divisions for probability calculation. In addition, the fixed set of states allows to implement the multiplication-free arithmetic encoding [5].

For example, let us consider state machine based probability estimation in H.264/AVC standard [15]. Input symbols are divided into two types: Most Probable Symbols (MPS) and Least Probable Symbols (LPS). State machine contains 64 states and is based on equation (10). Each state defines probability estimation for Least Probable Symbol. Set of probability values $\{\hat{p}_0, \hat{p}_1, ..., \hat{p}_{63}\}$ is defined as:

$$\begin{cases} \hat{p}_i = (1-\gamma)\hat{p}_{i-1}, \text{ where } i = 1, ..., 63, \hat{p}_0 = 0.5, \\ \gamma = 1 - \left(\dfrac{\hat{p}_{min}}{0.5}\right)^{\frac{1}{63}}, \hat{p}_{min} = 0.01875. \end{cases} \tag{11}$$

Probability estimation for symbol $x_{t+1}$ is calculated as

$$\hat{p}_{t+1} = \begin{cases} (1-\gamma)\hat{p}_t + \gamma, \text{ if } x_t = \text{LPS}, \\ \max\{(1-\gamma)\hat{p}_t, \hat{p}_{62}\}, \text{ if } x_t = \text{MPS}. \end{cases} \tag{12}$$

### 2.3 Probability Estimation Based on Virtual Sliding Window

Probability estimation using "Virtual Sliding Window" [14] is also based on equation (10), but it does not use state machine for probability calculation. For this algorithm probability estimation that $x_{i+1}$ is equal to one is defined as

$$\hat{p}_{i+1} = \frac{s_i}{2^{2w}}, \tag{13}$$

where $2^{2w}$ is a window length and $s_i$ is a virtual sliding window state which is recalculated by the following rule:

$$s_{i+1} = \begin{cases} s_i + \left\lfloor \dfrac{2^{2w} - s_t + 2^{w-1}}{2^w} \right\rfloor, \text{ if } x_i = 1 \\ s_i - \left\lfloor \dfrac{s_i + 2^{w-1}}{2^w} \right\rfloor, \text{ if } x_i = 0. \end{cases} \tag{14}$$

For stationary memoryless sources window length expansion increases the probability estimation precision and improves compression rate. For arbitrary source window length expansion may reduce estimation precision. Therefore, optimal window length selection is a complex problem because statistical properties of a binary source are unknown. In [14] the following simple heuristic algorithm of window length selection is proposed. Let us define $L = \{2^{2w_1}, 2^{2w_2}, ..., 2^{2w_l}\}$ as a set of window lengths. The output of the binary source is encoded and then window length is selected from the set $L$. During encoding probability estimations $\hat{p}_i(w_1), \hat{p}_i(w_2), ..., \hat{p}_i(w_l)$ are calculated. After encoding, bit stream length estimation is calculated by equation: $\hat{R}(w_k) = \sum_i \hat{r}_i(w_k)$, where

$$\hat{r}_i(w_k) = \begin{cases} -\log_2 \hat{p}_i(w_k), \text{ if } x_i = 1, \\ -\log_2 (1 - \hat{p}_i(w_k)), \text{ if } x_i = 0, \end{cases} \tag{15}$$

and window length $w^*$ is assigned by equation

$$w^* = \arg\min_k \hat{R}(w_k). \tag{16}$$

Thus, compression gain is reached by assigning specific window length selected by statistical properties of corresponding source. Therefore, Virtual Sliding Window provides better compression efficiency [14] in comparison to adaptation mechanism in H.264/AVC standard [15].

## 3 Computation Complexity Analysis

From Algorithm 1 follows, that lines 1–8 are used for each input binary symbol. On the other hand, amount of using of lines 9–25 is in direct proportion to number of bits in the output bit stream. Let use define $N$ as a number of the

input binary symbols, $R$ as a size of the output bit stream. Therefore encoding time for binary arithmetic coder $T_{arith}$ includes two main parts:

$$T_{arith} = \alpha_{arith} \cdot N + \beta_{arith} \cdot R, \tag{17}$$

where $\alpha_{arith}$ is the computation complexity of lines 1–8 per one input binary symbol, $\beta_{arith}$ is the computation complexity of lines 9–25 per one output binary symbol.

Using the reasoning described above, encoding time for binary range coder (see Algorithm 2) can be written as:

$$T_{range} = \alpha_{range} \cdot N + \beta_{range} \cdot \frac{1}{8} \cdot R, \tag{18}$$

where $\alpha_{range}$ is the computation complexity of lines 1–8 per one input binary symbol, $\beta_{range}$ is the computation complexity of lines 9–14 per one output byte.

It is known [20], that redundancy of integer implementation of arithmetic encoder depends on the number of bits for probabilities representation $\tau$ and bit size $v$ of registers *low* and *range*. Therefore, the size of the output bit stream

$$R \approx N \cdot \left( h(p) + 2 \cdot (\tau + \log e) \cdot 2^{-(v-2)} \right), \tag{19}$$

where $h(p)$ is entropy of binary memoryless source with ones probabilities $p$, $v \geq \tau + 2$.

Equations (17), (18) and (19) show, that if probability of ones is known, then for given arithmetic coder implementation encoding time is the linear function of a number of input binary symbols and source entropy $h(p)$.

Values $\alpha_{arith}$, $\beta_{arith}$, $\alpha_{range}$ and $\beta_{range}$ depend on processor architecture. For simplification let us assume that $\alpha_{arith} \approx \beta_{arith} \approx \alpha_{range} \approx \beta_{range}$ and $v \gg \tau$, then from (17), (18) and (19) follows that

$$\frac{T_{arith} - T_{range}}{T_{arith}} \approx \frac{\frac{7}{8} \cdot h(p)}{1 + h(p)} \in [0, ..., 0.4375]. \tag{20}$$

Figure 1 shows the encoding time for $10^8$ input binary symbols using Processor Intel Core 2 DUO, 3GHz. These results show that byte-oriented renormalization allows to decrease encoding time up to 40% in comparison with bit-oriented renormalization. In addition this figure shows that proposed linear model is fits for encoding time representation for Algorithms 1-2.

In real applications the probability of ones is not known. In this case for input binary symbol $x_i$ the probability estimation of ones $\hat{p}_i$ is calculated and used in line 1 of Algorithms 1-2 instead $p$. In this case, the size of output bit stream can be calculated as

$$R \approx \sum_{i=0}^{N-1} r_i, \tag{21}$$

where

$$r_i = \begin{cases} -\log_2 \hat{p}_i, \text{ if } x_i = 1, \\ -\log_2 (1 - \hat{p}_i), \text{ if } x_i = 0, \end{cases} \tag{22}$$

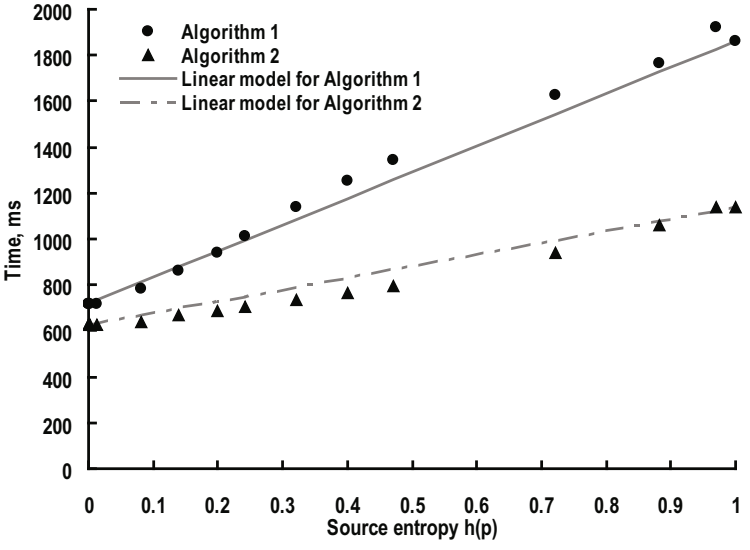



**Fig. 1.** Encoding time for $N = 10^8$ in case when probability is known

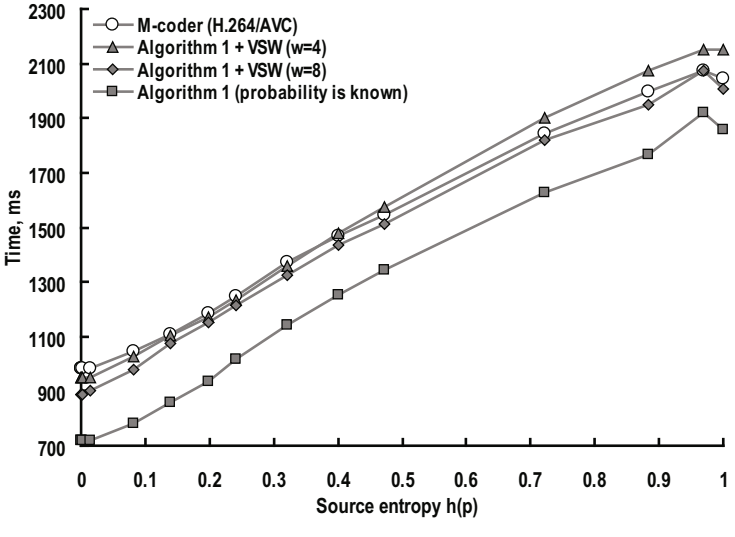



**Fig. 2.** Encoding time for $N = 10^8$ in case of binary arithmetic encoder with probability estimation using state machine and "Virtual Sliding Window"

Equations (17), (18) and (21) show, that the encoding time for adaptive binary arithmetic coder is the linear function of a number of input binary symbols and the output bit rate which depends on precision of the probability estimation $\hat{p}_i$.

Figure 2 shows the encoding time for Algorithm 1 in case of binary arithmetic encoder with probability estimation using state machine (as in H.264/AVC standard) and "Virtual Sliding Window" with parameters $w = 4$ and $w = 8$. This figure shows that both probability estimation algorithms require additional computation complexity. At the same time, "Virtual Sliding Window" allows to decrease the encoding time up to 10% (for $w = 8$) in comparison to probability estimation algorithm from H.264/AVC standard.

## 4 Conclusion

In this paper we have proved that in case of binary memoryless sources with known probability distribution encoding time for binary arithmetic encoder is a linear function of a number of input binary symbols and source entropy. We have shown that the adaptive binary arithmetic encoder implementation based on byte-oriented renormalization and probability estimation using "Virtual Sliding Window" has significantly less computational complexity than binary arithmetic encoder from H.264/AVC standard. Therefore, it is more preferable as an entropy encoding method for future video compression standards or state of the art codecs.

**Acknowledgements.** This work was supported by the Academy of Finland (project no. 213462, Finnish Programme for Centres of Excellence in Research 2006-2011), by the Russian Foundation for Basic Research (project 10-08-01071-a) and by the project of NSFC International Young Scientists.

## References

1. ITU-T and ISO/IEC JTC1, Digital Compression and cod- ing of continuous-tone still images. ISO/IEC 10918-1 ITU-T Recommendation T.81, JPEG (1992)
2. ITU-T and ISO/IEC JTC 1, JPEG 2000 Image Coding System: Core Coding System. ITU-T Recommendation T.800 and ISO/IEC 15444-1 JPEG 2000 Part 1 (2000)
3. Advanced video coding for generic audiovisual services. ITU-T Recommendation H.264 and ISO/IEC 14496-10, AVC (2009)
4. Eeckhaut, H., Schrauwen, B., Christiaens, M., Campenhout, J.: Speeding up Dirac's entropy coder. In: Proc. 5th WSEAS Int. Conf. on Multimedia, Internet and Video Technologies, pp. 120–125 (2005)
5. Pennebaker, W.B., Mitchel, J.L., Langdon, G.G., Arps, R.B.: An overview of the basic principles of the q-coder adaptive binary arithmetic coder. IBM J. Research and Development 32, 717–726 (1988)
6. Belyaev, E.: Low bit rate video coding based on three-dimensional discrete pseudo cosine transform. In: International Conference on Ultra Modern Telecommunications (2010)

7. Said, A.: Comparative analysis of arithmetic coding computational complexity. Hewlett-Packard Laboratories Report, HPL-2004-75 (2004)
8. High Efficiency Video Coding, http://www.h265.net/
9. Lu, X., Wang, Y., Erkip, E.: Power efficient H.263 video transmission over wireless channels. In: International Conference on Image Processing, pp. 533–536 (2002)
10. He, Z., Liang, Y., Chen, L., Ahmad, I., Wu, D.: Power-rate-distortion analysis for wireless video communication under energy constraints. IEEE Transactions on Circuits and Systems for Video Technology 15, 645–658 (2005)
11. Ryabko, B.: Imaginary sliding window as a tool for data compression. Problems of Information Transmission, 156–163 (1996)
12. Krichevski, E., Trofimov, V.: The performance of universal encoding. IEEE Transactions on Information Theory IT-27, 199–207 (1981)
13. Leighton, T., Rivest, R.L.: Estimating a probability using finite memory. IEEE Transactions on Information Theory IT-32, 733–742 (1986)
14. Belyaev, E., Gilmutdinov, M., Turlikov, A.: Binary arithmetic coding system with adaptive probability estimation by Virtual Sliding Window. In: Proceedings of the 10th IEEE International Symposium on Consumer Electronics, pp. 194–198 (2006)
15. Marpe, D., Schwarz, H., Wiegand, T.: Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard. IEEE Transactions on Circuits and Systems for Video Technology 7, 620–636 (2003)
16. Schindler, M.A.: Byte oriented arithmetic coding. In: Proceedings of Data Compression Conference (1998)
17. Vatolin, D.: Data compression methods. Dialog-MIFI Publisher, Moscow (2002) (in Russian)
18. Lindstrom, P., Isenburg, M.: Fast and Efficient Compression of Floating-Point Data. IEEE Transactions on Visualization and Computer Graphics 12(5), 1245–1250 (2006)
19. Subbotin, D.: Carryless Rangecoder (1999), http://search.cpan.org/src/SALVA/Compress-PPMd-0.10/Coder.hpp
20. Ryabko, B.Y., Fionov, A.N.: An efficient method for adaptive arithmetic coding of sources with large alphabets. Problems of Information Transmission 35(4), 95–108 (1999)

# Video Multicasting in an Autonomic Future Internet with Essentially-Perfect Throughput and QoS Guarantees

Ted H. Szymanski

Dept. ECE, McMaster University,
Hamilton, Ont. Canada L8S-4K1
teds@mcmaster.ca
http://www.ece.mcmaster.ca/faculty/teds

**Abstract.** A framework for an Autonomic Future Internet which supports 2 services classes, the *Essentially-Perfect QoS* (QoS) class and the *Best-Effort* (BE) class, is proposed. All provisioned traffic flows in the QoS class can achieve 100% throughput and essentially-perfect QoS guarantees. Bandwidth is provisioned for the QoS class periodically using measured and projected traffic demand matrices. Each router uses an efficient stochastic matrix decomposition scheduling algorithm to to achieve essentially-perfect QoS guarantees, for every competing QoS traffic flow, i.e., all admitted competing QoS traffic flows never experience congestion. (This QoS scheduling problem is a long-standing unsolved theoretical problem.) The multicasting of aggregated self-similar video streams over the proposed network is explored. It is shown that thousands of bursty self-similar video streams can be multicast across the proposed network with essentially-perfect throughput and QoS guarantees. The technology can be incorporated into Internet routers with minimal hardware cost, it is compatible with the existing IETF DiffServ and MPLS service models, it can achieve link efficiencies as high as 100%, and it can reduce Internet router buffer requirements and power requirements significantly.

**Keywords:** Quality of Service, QoS, Future Internet, autonomic, scheduling, low-jitter, buffer sizes, self-similar, video.

## 1 Introduction

The Internet network is an ubiquitous platform for the delivery of new services such as digital video. However, it suffers from many technical challenges including a reliance on significant over-provisioning, poor link efficiencies, poor resource-utilization, poor Quality of Service (QoS) guarantees and high power consumption. The inefficiency of the Internet is estimated to cost hundreds of millions of dollars in excessive energy costs per year and to contribute a noticeable amount of world green-house gasses. To address these problems, the International research community is exploring the *'Future Internet Network'* [1,2,3,4,5], and it is open to both *evolutionary* and *revolutionary* approaches.

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 608–619, 2011.

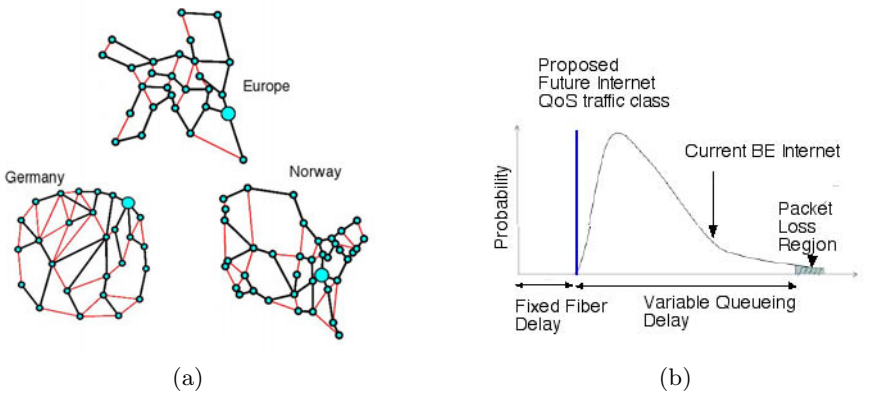



**Fig. 1.** (a) Internet backbone topologies (b) Internet end-to-end delay distribution [10]

Digital video represents a rapidly growing segment of the total Internet traffic [1]. The distribution of broadcast television over the Future Internet may require the distribution of 100s or 1,000s of aggregated high-definition video channels over IP multicast trees. The existing Best-Effort Internet network relies upon significant over-provisioning of bandwidth to reduce congestion [6]. However, this approach is unsustainable; the original designers of the Internet have recently argued that *"the Internet has been grossly overprovisioned. Network operators have deployed mountains of optical communication systems that can handle traffic spikes, but on average these run much below their full capacity..... So although users may not perceive the extent of the problem, things are already dire for many Internet service providers and network operators. Keeping up with bandwidth demand has required huge outlays of cash to build an infrastructure that remains underutilized"* [6]. In summary, new theories for the design of an efficient Future Internet network are needed.

Several challenges and important questions exist in the design and deployment of a Future Internet network [3,7]: *Q1: Should new service classes be introduced? Q2: Will the deployment of a new service class cause incompatibilities or disruption with the existing Internet? Q3: Who will pay for the new service classes, and how will billing be implemented?* In this paper, the framework for an *Autonomic Future Internet* which supports 2 service classes, the existing *'Best-Effort'* (BE) service classes along with a new *'Essentially-Perfect QoS'* (QoS) service class is proposed, building upon a theoretical foundation established in [8,9,10]. Router designs to realize the new QoS class are presented. These designs require only incremental hardware changes to existing router designs, and may solve many outstanding problems of the Internet network.

Fig. 1b illustrates the end-to-end delay distribution for the existing Best-Effort Internet network. The current BE Internet can experience end-to-end delays ranging from 10s to 1,000s of milliseconds. This delay distribution applies to the current Internet, IntServ, DiffServ, MPLS and ATM networks, which all use heuristic Best-Effort scheduling algorithms. Heuristic BE scheduling algorithms are used, since the core theoretical perfect-QoS scheduling problem has remained

unsolved for several decades, i.e., see [8-12]. Fig. 1b also illustrates the end-to-end delay for the proposed QoS service class in the proposed Future Internet. All admissible competing traffic flows in the proposed QoS class never experience congestion (see theorems in [8,9,10]), and receive a small and negligible end-to-end queueing delay in the network routers.

To demonstrate the technology, the multicasting of aggregated H.264 video streams is explored. Three internet topologies from a database of real and proposed networks are shown in Fig. 1a [13]. These 3 topologies are from various EU research projects. A typical IP multicast tree is shown in each topology. It is well-known that digital video traces exhibit significant self-similarity with long-range dependence, which will complicate the dimensioning of buffers/queues in a network [14]. The multicasting of aggregated video in each topology is extensively studied with simulations.

Section 2 presents the proposed Autonomic Future Internet network. Section 3 presents the video traffic models and analysis. Section 4 concludes the paper.

## 2 Framework for an Autonomic Future Internet

A basic Internet router using an Input-Queued (IQ) switch is shown in Fig. 2a. A router of size $M \times M$ consists of M input and output ports $IP(1..M)$ and $OP(1...M)$. Each input port has $M$ Virtual Output Queues (VOQs). Each input port has a input-controller to filter and classify incoming packets, a demultiplexer to forward packets to the appropriate VOQ, a server to select a VOQ for service, and a Best-Effort (BE) scheduler to control the server.

All current Internet routers (and ATM/MPLS switches) use heuristic *Best-Effort* schedulers as shown in Fig. 2a, which cannot achieve rigorous and mathematically provable QoS guarantees. Heuristic BE schedulers can typically achieve $\leq 80\%$ efficiencies through a switch, and require very large packet buffers per input port. Current Internet routers typically use the classic *'Delay-Bandwidth-Product-Rule'* to determine buffer sizes. A router with link speeds of 40 Gbps and a round-trip-time of 250 milliseconds typically requires buffers of $\approx 10$ Gbits per input port, or $\approx 1$ million IP packets [15]. To provide low delays per router given such large buffers, existing BE Internet routers carrying real-time traffic are typically over-provisioned and operate at a small fraction (i.e., 25-33%) of peak capacity and link utilization.

Let the time-axis be divided into *'Scheduling Frames'* consisting of F time-slots each. A time-slot is sufficient to transmit a maximum-size packet (i.e., 1500 bytes) over a link of capacity C. The traffic rates on network links can be expressed as a percentage of link capacity C, or as a number of packet transmissions $\leq$ F within a scheduling frame with F time-slots.

The theory for a *Future Internet* network which can achieve *essentially-perfect* throughput, link-efficiencies and QoS guarantees for all *'smooth'* and *'admissible'* traffic flows within the capacity region of the network, has already been established in [8,9,10]. A *'smooth'* traffic flow is defined as one which exhibits a low burstiness or jitter; an *'admissible'* traffic flow is defined as one which does not

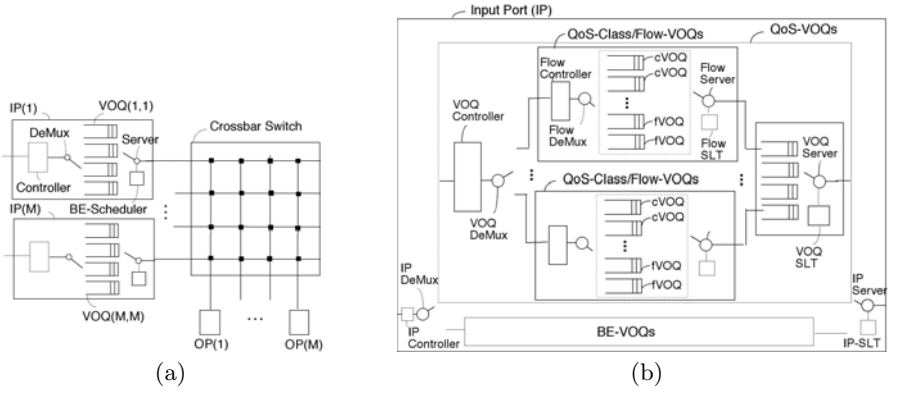



**Fig. 2.** (a) IQ router design. (b) Input Port design with QoS and BE service models.

violate any link and router capacity constraints, and *'essentially-perfect service'* is defined as service which deviates from perfect-service by at most K packets (see ahead for a formal definition). In the proposed Autonomic Future Internet network with 2 service classes, the majority of buffering for the QoS traffic class is moved external to the network, and the end-to-end router queuing delays for the QoS traffic class is negligible, as shown in Fig. 1b. Each source node has a *Traffic Shaper Queue* (TSQ) to shape bursty QoS traffic into a low-jitter stream before transmission. Each destination node has a *Traffic Playback Queue* (TPQ) to regenerate the original bursty streams. The Future Internet routers use minimal buffering, and offer essentially-perfect service to QoS traffic flows with near-minimal end-to-end delays, resource-utilization and energy requirements.

The problem of scheduling of multiple competing traffic flows within a single IQ switch, to achieve perfect throughput and QoS guarantees, is a long-standing unsolved theoretical problem [8,9,10,11,12]. This unsolved theoretical problem applies to the current Internet, IntServ, DiffServ, ATM and MPLS networks. Researchers at Bell Labs. have shown that scheduling multiple competing traffic flows within a single IQ switch to minimize jitter is NP-HARD [12] Recently, the QoS scheduling problem has been mathematically formulated using traffic rate matrices, and a fast polynomial time solution has been presented [8,9,10].

The proposed Autonomic Future Internet supports two service classes, the new *'QoS'* and the existing 'BE' service classes. The new QoS service class contains low-jitter traffic flows which are explicitly provisioned for end-to-end QoS. An input-port design supporting the two service classes, the QoS and BE service classes, is shown in Fig. 2b. The existing VOQs in a router are logically partitioned into the QoS and BE service classes. No new buffers are required, as existing Internet routers already have very large buffers. Incoming packets are filtered by the input controller; BE traffic is forwarded to the BE-VOQs, and QoS traffic is forwarded to the QoS-VOQs .

QoS and BE traffic flows can be identified by their packet headers or their traffic profile; QoS traffic flows have a sufficiently small jitter and service lead/lag.

BE traffic flows, including TCP traffic, typically have very large jitters and service lead/lags. However, BE traffic flows with a sufficiently small jitter can also be promoted up to the QoS service class, if there is sufficient bandwidth capacity provisioned for QoS traffic. The Autonomic-Controller (AC - to be described ahead) will attempt to ensure this condition when it computes its projected QoS traffic demand matrices for provisioning, and the AC can be invoked to dynamically provision more bandwidth when necessary.

Default traffic is treated as BE traffic by the traffic controllers in Fig. 2b, and is assigned to the BE-VOQs. Several servers are shown in Fig. 2b, the flow-server, the VOQ-server, and the IP-server. Each server can be controlled by pre-computed schedule, which is stored in *software-loadable Scheduler Lookup Table* (SLT). The SLT can have the length of a scheduling frame, i.e., $F$=1,024 packet-time-slots, and can have an entry for each time-slot $t$ . The scheduling of multiple competing traffic flows to achieve essentially-perfect QoS guarantees within a single IQ switch requires 2 schedules, which are defined next.

**Definition:** A *'VOQ-transmission-schedule* (VTS) for one router is a sequence of $F$ partial or full permutation matrices $P \in R^{M \times M}$ which define the crossbar switch configurations for the $F$ time-slots within a scheduling frame, i.e., $P \equiv \{P(t)\}, 1 \leq t \leq F$, where $P_{j,k}(t) = 1$ if VOQ(j,k) has a scheduled service opportunity in time-slot t. Each permutation matrix identifies up to M conflict-free VOQs for service per time-slot.

**Definition:** A *Flow-transmission-schedule* (FTS) for one router is a sequence of F matrices $Z \in R^{M \times M}$ which define the flow (or sub-class) to be serviced in each VOQ for the F time-slots within a scheduling frame, given a VOQ-transmission-schedule which identifies the VOQs to be serviced, i.e., $\mathbf{Z} \equiv \{Z(t)\}, 1 \leq t \leq F$ , where $Z_{jk}(t) = f$ if flow $f$ within VOQ(j,k) has a scheduled service opportunity in time-slot $t$.

In Fig. 2b, the SLT entries can be easily computed in software periodically using the scheduling algorithms in [8] in a network processor. The new hardware required in the input port of Fig. 2b can easily fit on a small FPGA. The change in router design can allow for essentially-perfect QoS service class, which can co-exist with regular BE traffic.

### 2.1 End-to-End QoS Guarantees

Let a backbone topology be denoted by a graph model $G(V, E)$, where $V$ is a set of nodes (routers) with labels $1, ..., N$, and $E$ is a set of directed edges, each with capacity $C$ (bits/sec). According to [16], a Best-Effort Internet backbone managed by Global Crossings uses MPLS with *Traffic Engineering* (MPLS-TE) to create a mesh of *Label Switched Paths* (LSPs) between all core routers in the backbone. Each pair of source-destination routers has a bandwidth requirement, and each source router routes its LSPs using a constraint-based shortest path routing algorithm. RSVP with Traffic Engineering (RSVP-TE) is used to provision each LSP across the Best-Effort MPLS-TE network, indicating a bandwidth requirement on each link in the LSP. The MPLS-TE network can offer only Best-Effort service, since all existing Internet routers and MPLS switches use heuristic

Best-Effort scheduling algorithms. Reasonable QoS can be achieved if the network is over-provisioned and if the links are operated at sufficiently light loads. We will use this Best-Effort MPLS-TE network as a starting point for our proposed Autonomic Future Internet. The proposed Future Internet can also use the *DiffServ* or *'DiffServ over MPLS'* service models. The following two definitions will be necessary.

**Definition:** The *Ideal Inter-Departure Time* (IIDT) of cells in a GR flow f with quantized guaranteed traffic rate of $\Theta(f)$ packet transmissions per scheduling frame is given by IIDT(f) = $F/\Theta(f)$ time-slots of duration (C/L) sec.

**Definition:** The *Normalized Service Lead/Lag* of a traffic flow f at time-slot $t$, denoted NSLL(t), with a guaranteed rate of $\Theta(f)$ packet transmission opportunities per scheduling frame is given by $S(t) - t/IIDT(f)$, where $S(t)$ is the received service measured in actual packet transmissions at time-slot t.

Assume any Future Internet backbone topology (i.e., as in Fig. 1a), with packets of size 1500 bytes. Consider any admissible routing of LSPs in the topology. An admissible routing for a traffic flow (or LSP) does not violate any link capacity constraints [10]. A traffic flow f is said to conform to $T(\lambda, \beta, \delta)$, denoted $\sim T(\lambda, \beta, \delta)$, if the average packet arrival rate is $\lambda$ packets/sec, the burst arrival rate is $\leq \beta$ packets/sec, and the maximum NSLL is $\delta$ packets [10]. Let $Af$ and $Df$ denote the arriving and departing traffic flows observed at a queue, and let $Q(t)$ denote the number of packets in the queue at time $t$.

Four theorems which characterize the end-to-end performance over any network topology are briefly summarized for completeness [10]. Theorem 1 states that given a QoS flow f traversing VOQ(j,k), with arrivals $Af \sim T(\theta(f), \beta, K)$ and service $Sf \sim T(\theta(f), \beta, K)$, and $Q(0) \leq 2K$ initially, then $Q(t) \leq O(K)$ for all $t$. In other words, every competing QoS traffic flow buffers at most 2K packets per router, where K is the NSLL bound. Theorem 2 states that the steady-state end-to-end queueing delay of a QoS traffic flow traversing H routers is $O(KH)$ IIDT time-slots. *In other words, all competing admissible QoS traffic flows never experience congestion.* This property is illustrated in Fig. 1b. Theorem 3 states that in the steady-state, the departures of a QoS traffic flow at any router in an end-to-end path of H routers will exhibit a maximum NSLL of K, i.e., $Sf \sim T(\theta(f), \beta, K)$. In other words, the NSLL of a flow is not cumulative when traversing multiple routers, so that arbitrarily large networks can be considered. Theorem 4 states that a QoS traffic flow which traverses an end-to-end path with H IQ routers can be delivered with essentially-zero network-introduced delay jitter, when a small playback buffer of size 4K packets is used.

The above 4 theorems establish that the proposed Future Internet network can deliver smooth traffic flows with essentially-perfect end-to-end QoS. However, real video traffic is often self-similar and extremely bursty [17]. In order to deliver self-similar video traffic with QoS guarantees, the video stream can be shaped at the source node by a *Video Shaper Queue* (VSQ) to have a bounded NSLL, and the bursty video stream can be recovered at the destination(s) using a *Video Playback Queue* (VPQ). These queues are external to the proposed Future Internet network, and they will be examined in section 3.

### 2.2 The Autonomic Controller

The performance of the proposed Future Internet network is characterized by several types of traffic rate matrices. Let $T^{QoS}$ and $T^{BE} \in R^{NxN}$ be *global traffic demand matrices*. Let the symbol $*$ denote either QoS or BE traffic. Element $T^*(i,j)$ denotes the bandwidth requirement for QoS or BE traffic between the pair of source-destination routers $(i,j)$. Let $L^{QoS}$ and $L^{BE} \in R^{NxN}$ be *link load matrices*, where element $L^*(i,j)$ denotes the bandwidth requirement for QoS or BE traffic over link (or edge) $(i,j)$. Let $D_k^{QoS}$ and $D_k^{BE} \in R^{MxM}$ be *router traffic rate matrices*, where element $D_k^*(i,j)$ (for $1 \leq k \leq N$) denotes the bandwidth requirement for QoS or BE traffic over VOQ(i,j) in router/switch (k).

Given the global traffic demand matrices, the link load matrices are determined by a routing algorithm R:

$$L^{QoS} = R\left(T^{QoS}\right); L^{BE} = R\left(T^{BE}\right) \tag{1}$$

The routing algorithm R can be centralized or distributed; it can use single-path or multi-path routing. In the Global Crossing MPLS-TE network, the routing is performed by the source routers using a constraint-based routing algorithm. The proposed Future Internet can use the same routing methodology. When an LSP is routed, RSVP-TE can be used to dynamically provision bandwidth for each LSP. The *'DiffServ over MPLS'* technology can also used, so that bandwidth for DiffServ traffic classes can be dynamically provisioned using RSVP-TE and MPLS-TE.

In the MPLS-TE model, LSP connections are provisioned using RSVP-TE, and each switch k can automatically update its own link load matrices $D_k^{QoS}$ (and optionally $D_k^{BE}$) in response to the RSVP-TE provisioning messages. Given a link load matrix, each switch can schedule its own QoS traffic in software using the scheduling algorithms in [8,10]. Each switch can schedule its own BE traffic using the same heuristic BE schedulers that it currently uses (or it can also use the algorithms in [8,10]).

The global traffic demand matrices $T^{QoS}$ and $T^{BE}$ evolve over time. Let an *'Autonomic-Controller'* (AC) be distributed controller realized in a software overlay layer, which maintains the recent history of these matrices on a periodic basis, i.e., every 15 minutes. For example, $T^{QoS}_{(h,d,y)}$ and $L^{QoS}_{(h,d,y)}$ represent the global traffic demand and link traffic rate matrices for QoS traffic in the interval $h$ of day $d$ in year $y$, and similarly for BE traffic. (To maintain estimates every 15 minutes, $1 \leq h \leq 24*4$). In an MPLS-TE system these matrices can be updated automatically in response to RSVP-TE provisioning messages. However, it is desirable for an AC to record the history of these matrices, to optimize system performance. Given the history of these matrices, the AC can compute projected traffic demands, and perform optimized constraint-based routing algorithms to realize the projected traffic demand in advance.

The AC may compute projected values for these global traffic demand matrices, based upon a function of past history, i.e.,

$$T^{QoS}_{(h+1,d,y)} = f\left(T^{QoS}_{(-,-,y)}, T^{QoS}_{(-,-,y-1)}\right); T^{BE}_{(h+1,d,y)} = f\left(T^{BE}_{(-,-,y)}, T^{BE}_{(-,-,y-1)}\right) \tag{2}$$

**Table 1.** Statistics for Aggregated H.264/AVC Video Streams (based on real KAET Talk Show video)

| Channels | MIn. Bit-Rat | Min. $X^{(m)}$ | E$[X^{(m)}]$ | Max. $X^{(m)}$ | SCV | CV | Hurst Val. |
|---|---|---|---|---|---|---|---|
| 1 | 1.464 Mbps | 0.469 Kb | 50 Kb | 781 Kb | 3.4 | 1.84 | 0.770 |
| 10 | 14.64 Mbps | 66.2 Kb | 500 Kb | 2028 Kb | 0.413 | 0.642 | 0.975 |
| 100 | 146.4 Mbps | 2637 Kb | 4996 Kb | 8278 Kb | 0.0315 | 0.178 | 1.03 |
| 1000 | 1.463 Gbps | 39.16 Mb | 48.79 Mb | 58.9 Mb | 0.0044 | 0.066 | 1.02 |

where $-$ denotes any admissible $h$ or $d$. For example, the AC may use the current values of the QoS traffic rate matrices and partial derivatives over the hour, week and year to compute the projected QoS global traffic demand matrix, and similarly for BE traffic:

$$T^{QoS}_{(h+1,d,y)} = f\left(T^{QoS}_{(h,d,y)}, T^{QoS}_{(h,d-7,y)}, T^{QoS}_{(h,d,y-1)}, \frac{\partial T^{QoS}_{(h,d,y)}}{\partial h}, \frac{\partial T^{QoS}_{(h,d,y)}}{\partial d}, \frac{\partial T^{QoS}_{(h,d,y)}}{\partial y}\right) \tag{3}$$

When the switch traffic rate matrices $D^{QoS}_k$ are updated using RSVP-TE, the algorithms in [8,10] can be used to schedule the packet transmissions through each switch, to meet QoS guarantees. We assume these matrices are updated periodically, i.e., every 10-15 minutes in a core-network, and perhaps every minute in an edge network. The memory requirements for recording the history of these matrices is relatively small. Each matrix entry requires 2 bytes to provide a reasonable resolution on the bandwidth requirement. The history of one 50x50 matrix over 1 year at 15 minute intervals requires only 175 Mbytes (without compression), which is negligible compared to the memory in an iPod music player.

For any backbone topology, the buffer requirements per router can be reduced by factors of 100 - 10,000 if all traffic uses the QoS-enabled service model. Each QoS-enabled traffic flow requires $\leq 2$ packet buffers per router on average [8,9,10]. To achieve resource-efficiency and power-efficiency, service providers can offer the essentially-perfect QoS class *for free to all users*, rather than charging for it [3,7], representing a fundamental paradigm-shift. The QoS-enabled service can be offered for free to all video traffic sources which use a VSQ with a sufficiently small NSLL at the sources. The VSQ and VPQ functions can be incorporated into software modifications of streaming video servers and players.

## 3 Video Traffic Models

In this section, the multicasting of aggregated video over the proposed Future Internet is explored. The H.264 video frames sizes for the *KAET Talk Show* video available at the University of Arizona web-site [17] are used. The Talk Show trace is a 30 minute HD 1920x1080 video trace in the H.264/AVC format, with a mean bit-rate of 1.464 Mbps and a compression ratio of $\approx 150$. The ratio

of peak to minimum frame size is 1,667, indicating a very large burstiness. To aggregate multiple vide streams, copies of this video stream are circularly rotated by a random amount and added together (as in [17]).

A discrete-time stochastic process representing one H.264 *"video stream"* is a sequence of video frame sizes of length L denoted $X = \{X_1, \cdots X_L\}$. Each *'Group of Pictures'* (GOP) is a sequence of $G$ video frames consisting of one I-frame, and several dependent P and B-frames. The Talk-Show stream uses a GOP of length $G = 12$ with the format {I,B,B,P,B,B,P,B,B,P,B,B}. Therefore, each $GOP_i$ represents a sequence of consecutive video frame sizes $\{X_{(i-1)\cdot G+1}, \cdots, X_{i\cdot G}\}$.

Let X denote a stochastic process with a mean value $\mu = E[X_t]$, with a finite variance $\sigma^2 = E[(X_t - \mu)^2]$, and with an autocorrelation function $r(k) = E[(X_t - \mu)(X_{t+k} - \mu)]/E[(X_t - \mu)^2]$ , $(k = 01, 1, 2, ...)$ that depends only on the lag k. For each $m = 1, 2, 3, ...$, let $X^{(m)}$ denote a new time-series (process) obtained from a series $X$ by averaging $X$ over independent blocks of size $m$ [14]:

$$X_{(k)}^{(m)} = (1/m)(X_{(km-m+1)} + ... + X_{(km))}, k = 1, 2, 3, .... \tag{4}$$

Let $r^{(m)}$ denote the autocorrelation function of $X^{(m)}$. The process $X$ is *second-oder self-similar* with Hurst parameter $H = 1 - \beta/2$ if the processes $X^{(m)}$ have the same autocorrelation function [14], i.e.,

$$r^{(m)}(k) = r(k), \forall m = 1, 2, 3, ..., (k = 1, 2, 3, ....) \tag{5}$$

Equivalently, the aggregated processes $X^{(m)}$ have the same second-order properties.

The statistics for several aggregated video streams are shown in Table 1. The *Coefficient of Variation (CV)* should be a good indicator for the amount of *excess bandwidth* needed to accommodate bursts of traffic. According to Table 1, a single video stream should need an excess bandwidth of $\approx 84\%$ to accomodate bursts, while the aggregation of 1,000 video streams should need an excess bandwidth of $\approx 6.6\%$ to accomodate bursts. The Hurst parameter was computed for all aggregated flows and it is well above 0.5, indicating that the aggregated flows are highly self-similar. An exact analysis for the sizes of the VSQ and VPQ as a function of the excess bandwidth is presented.

The VSQ can be modelled as a discrete-time discrete-state $D^x/D^y/1$ queueing system, with a deterministic inter-arrival and inter-departure times D [10]. Batches of video frames arrive and depart from the VSQ at deterministic times (i.e., D=1/30 sec. per batch) with batch size distributions $x$ and $y$ respectively. The arriving batch-size distribution $x$ is determined from the statistics of the arriving video streams, as shown in Table 1. The service batch-size distribution $y$ is determined from the provisioned guaranteed service-rate of the VSQ, and the bound on the NSLL of the VSQ.

Let $Q_{SQ}(t)$ be a random variable denoting the size of the VSQ for an aggregated traffic flow at discrete-time $t$. The size of the shaper queue (in bytes) for an aggregated traffic flow consisting of $A$ H.264 video channels (each denoted by vector X) is given by :

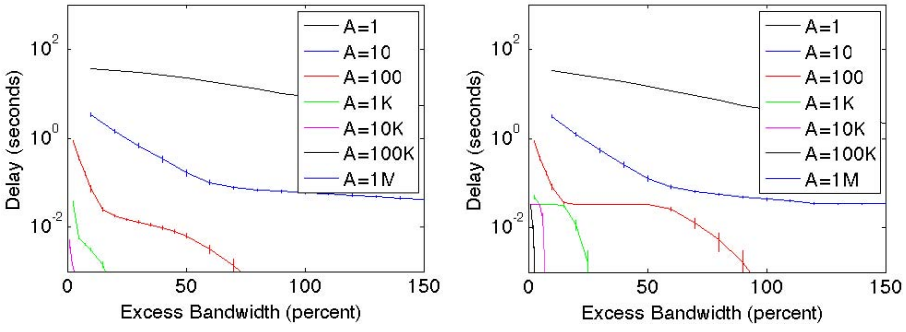



**Fig. 3.** (a) VSQ delay vs. excess bandwidth. (b) VPQ delay vs. excess bandwidth.

**Table 2.** End-to-End Delay Bounds for Aggregated H.264/AVC Video Streams

| Channels | EXBW | Hops | RQ Delay (millisec) | SQ Delay (millisec) | PBQ Delay (millisec) |
|---|---|---|---|---|---|
| 1 | 100% | 20 | 156 | 8.9 sec | 4.9 sec |
| 10 | 50% | 20 | 21 | 168 | 128 |
| 100 | 15% | 20 | 2.7 | 25 | 37 |
| 1,000 | 5% | 20 | 0.3 | 5.7 | 33 |
| 10,000 | 2% | 20 | $\leq 0.1$ | 1.5 | 33 |

$$Q_{SQ}(t+1) = max(0, Q_{SQ}(t) + \sum_{f=1}^{A} X(t,f) - R * \Delta t + NSLL(t)) \qquad (6)$$

where NSLL(t) is a random VSQ NSLL, and where R is the provisioned rate. As summarized in theorems 1-4, competing admissible QoS traffic flows never experience congestion, so that each end-to-end QoS traffic flow can be analysed in isolation from other competing QoS traffic flows, using Eq. (6).

The VSQ was simulated for 50 hours of video traffic, for various choices of excess bandwidth provisioned in the multicast trees. The VPQ was also simulated to recover the aggregated video stream for each level of aggregation. Fig. 3 illustrates the maximum VSQ and VPQ delays for aggregated streams containing 1...1M video streams. The 95% confidence intervals are shown. The VSQ and VPQ delays drop rapidly as the excess bandwidth increases. Additional details on the video simulations and analyses are presented in [9,10].

### 3.1 End-to-End Delays in the Future Internet

Table 2 illustrates the queueing delays for aggregated video streams traversing 20 routers in an end-to-end path, using the proposed QoS technology. The aggregation of 1,000 video streams each requiring 1.464 Mbits per second (Mbps)

will require an aggregate bandwidth of 1.464 Gbps. To achieve a small delay, let an excess bandwidth of 5% be provisioned in the multicast tree, so the guaranteed-rate is 1.05x1.464=1.611 Gbps. The mean time between IP packets in a perfectly-scheduled transmission is 1500 bytes/1.611 Gbps $\leq$ 7.44 $\mu$sec, which equals the IIDT of this aggregated flow. The mean queueing delay for any flow in any router is $\leq$ min(2 IIDT,$d(F)$), where $d(F)$ is the scheduling frame duration [8]. This bound applies for all link loads $\leq$ 100%. Letting $d(F)$= 1 millisec, then the mean queueing delay along an end-to-end path of 20 routers is $\leq$ 20*2*7.44 $\mu$sec $\leq$ 298 $\mu$sec. Given the aggregation of $10^3$ video streams, a provisioning of 5% excess bandwidth will result in a VSQ delay of 5.7 millisec, a VPQ delay of 33 millisec, and a small cumulative queueing delay within the Internet routers of $\leq$ 1 millisec. An aggregation of $10^4$ or more video streams can achieve total end-to-end queueing delays $\leq$ 35 millisec with 2% excess bandwidth, and a link-efficiency of $\geq$ 98%. The same concepts can be applied to local area networks to achieve QoS guarantees.

## 4 Conclusions

A framework for an Autonomic Future Internet network which supports 2 service classes, the *essentially-perfect QoS* class and the *Best-Effort* class, is proposed. Bandwidth is provisioned for QoS traffic using measured and projected traffic demand matrices. The routers use a mathematical packet scheduling algorithm to achieve essentially-perfect throughput and QoS guarantees for all competing admissible QoS traffic flows. We also explore a new paradigm for Internet QoS: *To offer access to the essentially-perfect QoS class for free*, in addition to the traditional *Best-Effort* class. Content providers such as Google's YouTube have a strong incentive to use the new QoS service class, to realize a potentially significant reduction in monthly bandwidth costs (as less excess bandwidth is provisioned in the multicast trees), and to offer essentially-perfect QoS to its users for free. Internet users have a strong incentive to use the new QoS service class, to achieve better QoS and lower delays for free (depending upon their local area networks). Internet service providers have a strong incentive to support the new QoS service class, to realize a potentially significant capacity increase in their infrastructure. Current IP routers and links carrying real-time traffic operate at typically 25-50% of peak capacity, as they are significantly over-provisioned to provide relatively poor QoS guarantees. Using the new QoS service class, IP links can operate at loads as high as 50 or 100% of capacity. A capacity increase of 2-4 times can represent a potential capital cost savings of several billion dollars for a national backbone network. Finally, internet router manufacturers have a strong incentive to support the new QoS service class, as the router buffer sizes required for QoS traffic can be less than 1% of the buffer sizes required for current Best-Effort traffic [8,9,10]. It is estimated that router buffers represent a significant fraction of the cost, size and power dissipation of existing Internet routers. Routers can be made smaller, faster, less-expensive, requiring less power and simultaneously offering a new essentially-perfect QoS service class.

## References


1. Cisco Systems New Release, Approaching the Zettabyte Era
2. European Union Future Internet Portal (2011), http://www.future-internet.eu
3. Shenker, S.: Fundamental Design Issues for the Future Internet. IEEE JSAC, 1176–1188 (September 1995)
4. Giordano, S., Salsano, S., Van den Berghe, S., Ventre, G., Giannakopoulos, D.: Advanced QoS Provisioning in IP networks: The European Premium IP Projects. IEEE Comm. Mag., 30–36 (January 2003)
5. Carapinha, J., Bless, R., Werle, C., Miller, K., Dobrota, V., Rusm, A.B., Grob-Lipski, H., Roessler, H.: Quality of Service in the Future Internet. In: Proc. ITU-K Kaleidoscope, India (December 2010) (Available at IEEEXplore)
6. Roberts, L.G.: A Radical New Router: The Internet is Broken - Lets Fix It. IEEE Spectrum (July 2009)
7. Shetty, N., Schwartz, G., Walrand, J.: Internet QoS and Regulations. IEEE/ACM Trans. Networking 18(6) (December 2010)
8. Szymanski, T.H.: A Low-Jitter Guaranteed-Rate Scheduling Algorithm for Packet-Switched IP Routers. IEEE Trans. Comm. 57(11) (November 2009)
9. Szymanski, T.H., Gilbert, D.: Internet Multicasting of IPTV with Essentially-Zero Delay Jitter. IEEE Trans. Broadcasting 55(1) (March 2009)
10. Szymanski, T.H., Gilbert, D.: Provisioning Mission-Critical Telerobotic Control Systems over Internet Backbone Networks with Essentially-Perfect QoS. IEEE JSAC 28(5) (June 2010)
11. Chen, W.J., Chang, C.-S., Huang, H.-Y.: Birkhoff-von Neumann Input Buffered Crossbar Switches. IEEE Trans. Communications 49(7) (July 2001)
12. Keslassy, I., Kodialam, M., Lakshman, T.V., Stilliadis, D.: On Guaranteed Smooth Scheduling for Input-Queued Switches. IEEE/ACM Trans. Networking 13(6) (December 2005)
13. Orlowski, S., Wessaly, R., Pioro, M., Tomaszewski, A.: SNDlib 1.0 - Survivable Network Design Library. Networks 55(3), 276–286 (2009)
14. Leland, W.E., Taqqu, M.S., Willinger, W., Wilson, D.V.: On the Self-Similar Nature of Ethernet Traffic. IEEE/ACM Trans. Networking 2(1) (February 1994)
15. Iyer, S., Kompella, R., Mckeown, N.: Designing Packet Buffers for Router Linecards. IEEE Trans. Networking 16(3) (June 2008)
16. Gunnar, A., Johansson, M., Telkamp, T.: Traffic Matrix Estimation on a Large IP Backbone - A Comparison on Real Data. In: ACM Int. Meas. Conf., Italy (October 2004)
17. Srinivasan, S.K., Vahabzadeh-Hagh, J., Reisslein, M.: The Effects of Priority Levels and Buffering on the Statistical Multiplexing of Single-Layer H.264/AVC and SVC Encoded Video Streams. IEEE Trans. Broadcasting 56 (September 2010)

# Splitting and Merging for Bandwidth Exploitation in SVC-Based Streaming Networks

Tsang-Ling Sheu and Yi-Chuen Hsieh

Department of Electrical Engineering,
National Sun Yat-Sen University,
Kaohsiung, Taiwan
sheu@ee.nsysu.edu.tw

**Abstract.** This paper presents stream splitting and merging (SSM) mechanisms to effectively exploit network bandwidth in SVC-streaming networks. The multi-layer characteristics of SVC (Scalable Video Coding) are fully utilized to split and merge concurrent video streams. In stream splitting, adequate SVC layers for splitting are calculated from periodical measurement of network bandwidth. In stream merging, a certain number of enhancement layers of one stream are merged to the top of base layer of another stream. The novelties in stream merging are in two aspects: (i) if the stream to be merged is slower than the one to contribute, all the arriving packets of the faster stream are buffered; (ii) if the stream to be merged is faster than the one to contribute, the packets of the faster stream are delayed. For the purpose of validation, the proposed SSM algorithms are implemented on Linux platform. In the experiments of a wired and wireless heterogeneous network, buffering delay and queuing length are analyzed, respectively, to demonstrate the effectiveness of the proposed SSM.

**Keywords:** SVC, splitting, merging, bandwidth exploitation, Linux.

## 1 Introduction

To adapt to variable network bandwidths and different resolutions of terminals, SVC (Scalable Video Coding) compression techniques [1] were introduced recently for multimedia delivery over networks. An SVC-encoded film possesses multiple-layer characteristics, i.e., one base layer and many enhancement layers. In SVC, only base layer can be independently decoded at a receiver. Decoding an enhancement layer will rely on the base layer and other lower enhancement layers. One of the benefits offered by SVC is that it can provide extreme flexibility in changing bit rates of a film dynamically. In other words, if different numbers of enhancement layers are grabbed by a receiver, decoding bit rates (DBR) are varied to match available network bandwidth. Another benefit offered by SVC is that DBR can be adaptively adjusted so that different resolutions of terminals can be satisfied; the more enhancement layers a decoder can receive, the higher resolution a terminal can achieve.

Previous works in adjusting the streaming bit rates for SVC-based multimedia networks include two different aspects: (i) based on priority, and (ii) by intermediate nodes. In the first aspect, L. N. Zhang, *et al.* [2] proposed a priority-based layer switching (PLS) scheme. In PLS, according to the receiver reports of RTCP (RTP

S. Balandin et al. (Eds.): NEW2AN/ruSMART 2011, LNCS 6869, pp. 620–631, 2011.

Control Protocols), partial of enhancement layers are discarded to adapt to available bandwidth. Similarly, J. C. Chiang, *et al.* [3] introduced a method to discard enhancement layers of low-priority users when the burden of BS in a WiMAX network exceeds a threshold. L. N. Zhang, *et al.* [4] improved PSNR by embedding FEC (Forward Error Correction) codes to the packets of different SVC layers in a GPRS/EDGE network. I. Kofler *et al.* [5] proposed a proxy server which divides traffic into two priorities, high for video and low for best effort. By preempting the best effort traffic, the quality of an SVC stream can be improved.

In the second aspect, streaming bit rates are adjusted by intermediate nodes. Y. Zhao, *et al.* [6], C. Griwodz, *et al.* [7], and H. Jin, *et al.* [8] proposed different algorithms to effectively save network bandwidth by merging SVC streams on an intermediate node. H. Wu, *et al.* [9] proposed a feedback mechanism, where an intermediate node can measure and report bandwidth usage so that the sender can pick up an adequate path to transmit video streams. To solve bottleneck problems, S. Mao, *et al.* [10] proposed a stream-splitting method with which every sub-stream may take different paths for transmissions. According to the RTCP reports, E. Kurutepe, *et al.* [11] proposed a method to dynamically pull out different numbers of enhancement layers from an intermediate node. Finally, C. M. Lin, *et al.* [12] suggested an approach in choosing a media relay node (MRN). In their approach, MRN is used for dynamically adjusting the number of enhancement layers along with the available bandwidth.

Unlike the previous works focusing on pure research, in this paper, the proposed stream splitting and merging (SSM) algorithms are implemented on an SVC streaming gateway (SSG). In stream splitting, SSG periodically estimates available bandwidth to determine the adequate number of enhancement layers (ELs) to split from an existing stream. In stream merging, SSG merges a number of ELs from one stream to another stream with fewer layers, when it detects the available bandwidth is increasing. The novelty in stream merging is that the SSG can operate smoothly even when two streams arrive at SSG in two different times. Specifically, when a stream to be merged arrives at SSG a little bit late than the one to contribute, SSG will copy all the arriving packets of the faster stream into the buffer, referred to as copy-to-merge (CM) buffer. On the other hand, when a stream to be merged arrives at SSG earlier than the one to contribute, the SSG simply delays the packets of the faster stream in a buffer, referred to as delay-to-merge (DM) buffer. Experiments are conducted on a wired and wireless heterogeneous network to demonstrate the effectiveness of the proposed SSM. Buffering delay and queuing length are analyzed for the CM and DM buffers, respectively, to study different characteristics incurred.

The remainder of this paper is organized as follows. In Section 2, the proposed SSM algorithms are presented. Section 3 illustrates the differences between CM and DM buffering. In Section 4, implementations of the SSM algorithms on a Linux platform along with the experiments conducted on a heterogeneous network are described. Finally, Section 5 gives the concluding remarks.

## 2 The SSM Algorithms

To fully exploit bandwidth of wireless networks, such as WiMAX and WLAN, in this paper, we design the stream splitting and merging (SSM) algorithms and implement the algorithms on an SVC Streaming Gateway (SSG). The operations of SSM are

illustrated by an example. As shown in Figure 1, in the beginning, client 1 requests a video stream with four SVC layers, one Base Layer (BL) and three Enhancement Layers (EL-1, EL-2, and EL-3), from an IPTV server. An SSG is installed between two networks, a WiMAX and a WLAN ad-hoc network. Since clients 2 and 3 request only three layers (one BL plus two ELs) of the same stream from the server, SSG simply splits the lower three layers (BL, EL-1, and EL-2) from the stream downloaded by client 1. This stream manipulation technique is referred to as stream splitting in this paper. On the other hand, since client 4 requests four layers of the same stream, to effectively exlpoit wireless bandwidth, the upper-most layer (EL-3) from client 1 is now merged direclty with the lower three layers (BL, EL-1, and EL-2) by client 3, which finally forwards the four layers to client 4. This stream manipulation technique is referred to as stream merging in this paper.

## 2.1 Stream Splitting

Prior to introducing the techniques of stream splitting, Figure 2 illustrates the handshaking procedure among a client, the proposed SSG, and a video server. Once TCP 3-way handshaking was completed, Client 1 uses a modified RTSP (Real-Time Streaming Protocol), i.e., RTSP-lite with RR=1, BRM=0, DM=0, and MV_id=3, to set up a connection with the server. As shown in Figure 3, the header structure of RTSP-lite is designed on the top of TCP. In the 2-byte header, RR (1 bit) denotes request/reply, BRM (1 bit) denotes bit-rate map, DM (1 bit) denotes decoder's messages, LN (3 bits) denotes the number of SVC layers requested, and MV_id (8 bits) denotes movie ID.

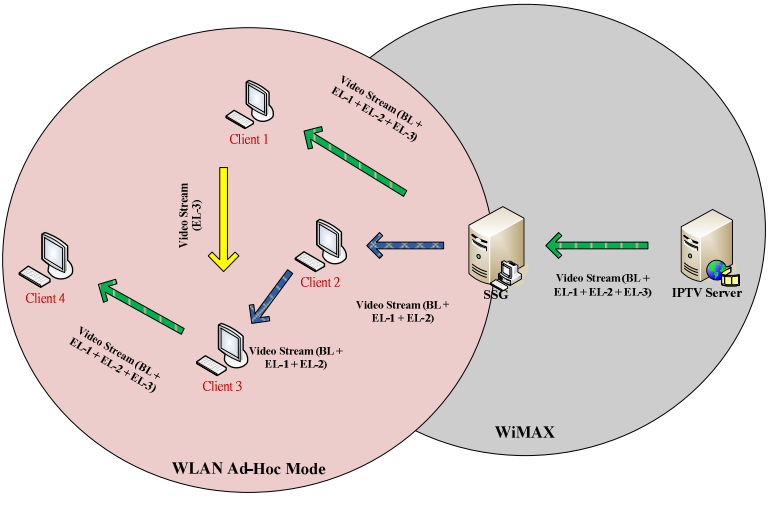



**Fig. 1.** Stream splitting and merging

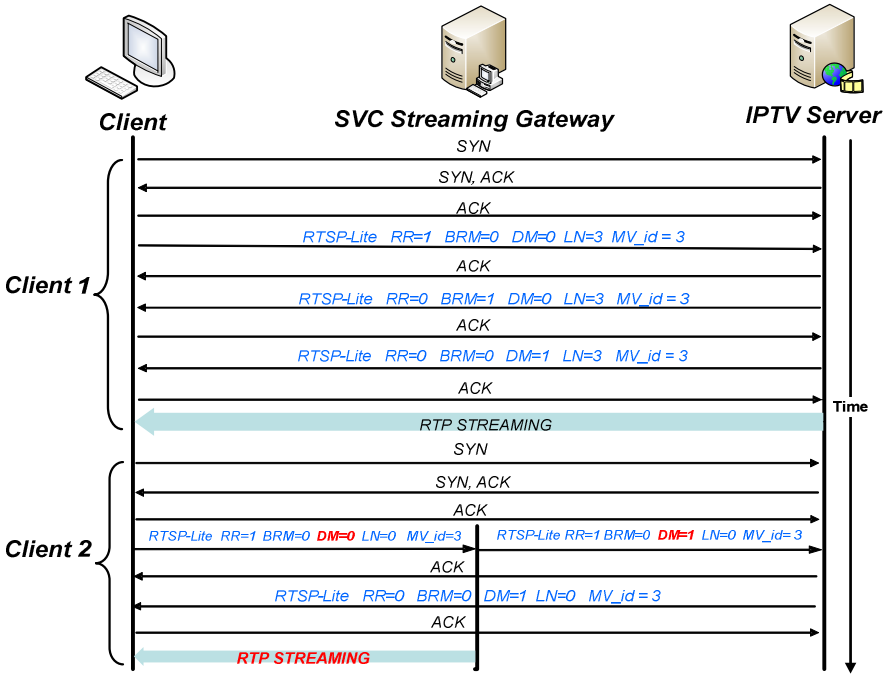



**Fig. 2.** Handshaking of stream splitting

In the scenario, we assume Client 2 issues a request for the same movie (MV_id=3) right after Client 1 receives the stream. The same movie ID in the request message was intercepted by SSG. After SSG modifies DM bit from 0 to 1, the server is triggered to send out decoder's messages. When SSG intercepts decoder's messages, it begins to split a certain number of SVC layers based on the available bandwidth.

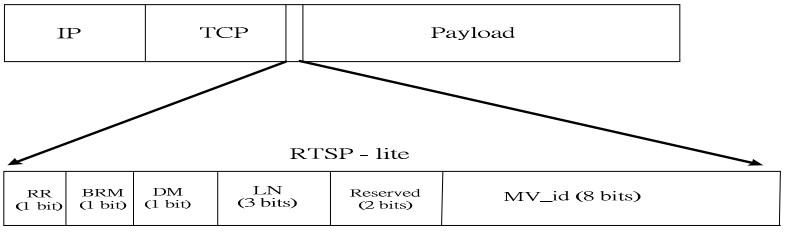



**Fig. 3.** Header design for RTSP-lite

Figure 4 shows the detail operations of stream splitting. When SSG intercepts the request from Client 2 (TCP Port No = 50100 && RR = 1), it will examine whether the movie ID is the same as that of any currently transmitted video streams; if yes, SSG sets DM = 1 to notify the server that decoder's message is required for transmission rather than the bit rate map (BRM). Stream splitting is invoked once decoder's message arrives at SSG. Next, SSG will determine how many SVC layers can be split

to Client 2. To accomplish this, available bandwidth (AB) for Client 2 is calculated as from Eq. (1).

$$AB = BW - BG - SBR1 \tag{1}$$

For simplicity, the stream requested by client 1 is referred to as stream 1 (S1), and that by Client 2 as stream 2 (S2). In Eq. (1), BW (Bandwidth) denotes the maximum bandwidth of a link and BG (Background traffic) denotes the background bit rates by excluding the bit rate of stream 1 (SBR1). By consulting with BRM, SSG can determine the number of SVC layers, denoted as $n$, to split based on Eq. (2).

$$BR\,(Layer_{\,n}) \le AB < BR\,(Layer_{\,n+1}) \tag{2}$$

where $BR(Layer_n)$ and $BR(Layer_{n+1})$ denote the bit rates required up to $n$-layer and ($n$+1)-layer, respectively. When AB increases, the SVC layers split to Client 2 are increased accordingly.

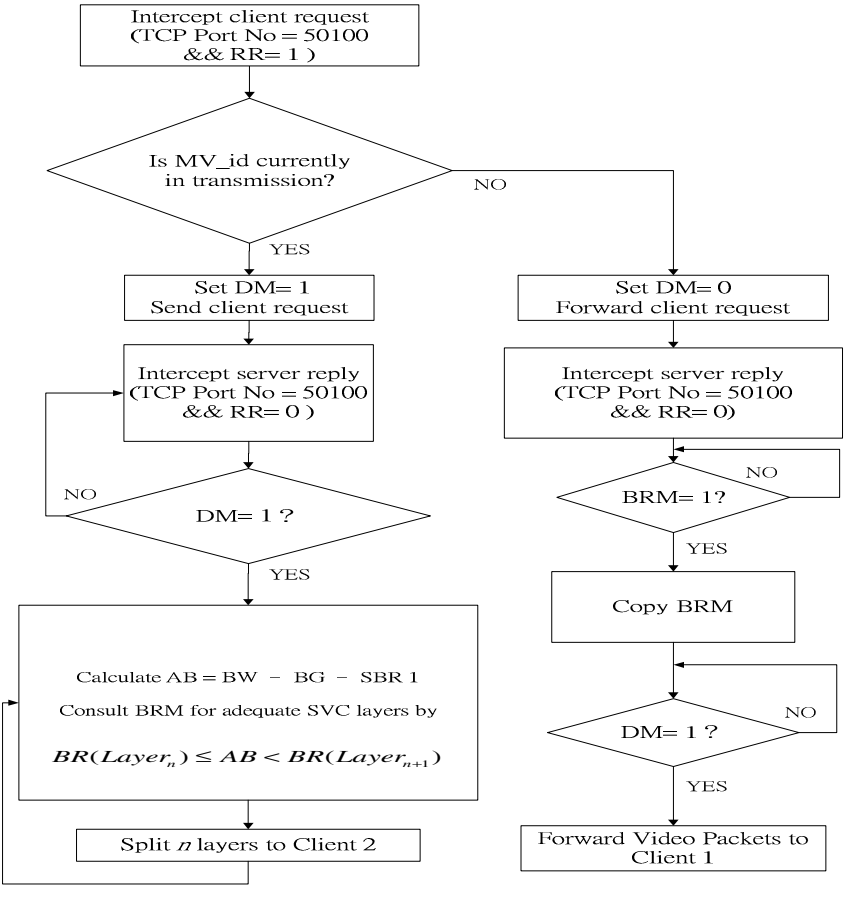



**Fig. 4.** Procedure of stream splitting

### 2.2 Stream Merging

The main goal of stream merging is to merge a certain number of SVC layers from one stream to the top of another stream of the same movie. To calculate the number of SVC layers to merge, Eq. (1) and Eq. (2) are respectively modified to become Eq. (3) and Eq. (4).

$$AB = BW - BG - SBR1 - SBR2 \tag{3}$$

$$BR(Layer_{n-m}) \leq AB < BR(Layer_{n+1-m}) \tag{4}$$

where SBR1 and SBR2 denote the bit rates of S1 and S2, respectively, $m$ denotes the original SVC layers of S2, and $n$ denotes the number of SVC layers to merge from S1 to S2. Due to different delays that two streams may encounter in a network, two scenarios are investigated. The first scenario assumes the stream to be merged arrives at SSG slightly late than the stream that will contribute the SVC layers, while the second scenario assumes the opposite behavior. To deal with different scenarios, we design two buffer mechanisms, CM (copy-to-merge) and DM (delay-to-merge) buffers at SSG, respectively.

## 3 The CM and DM Buffering

### 3.1 CM Buffering

Figure 5 shows the operation of CM buffering. Here, we assume stream 2 (S2), the one to be merged, arrives at SSG slightly late than stream 1 (S1), the one to contribute. It is also asumed that S1 has four layers (BL, EL-1, EL-2, and EL-3), and S2 has two layers (BL and EL-1). Concurrently with packet frowarding, arriving packets of every layer (from BL to EL-3) in S1 are stored into the CM buffer. This is because there is no way to know how many SVC layers to merge before S2 really arrives. When S2 arrives at SSG, the number of SVC layers to merge can be computed from Eq. (4), say two for example. Hence, two layers (EL-2 and EL-3) of packets are merged to S2. This merging process is repeatedly operated on a frame-by-frame basis by SSG; a frame will not be forwarded until it is completely constructed (video frame 4 of S2 in the figure). It is noticed that two streams are not synchronized after this CM buffering.

### 3.2 DM Buffering

Figure 6 shows the operation of DM buffering. Here, we asume S2, the one to be merged, arrives at SSG earlier than S1, the one to contribute. Other similar assumptions in the CM buffering are applied here. However, different from the CM buffering, inside the DM buffer, we observe that SSG does not have to buffer every arriving packet of S1. Instead, SSG stores every arriving packet of S2 into the buffer. Thus, buffer size requirement for the DM is relatively smaller than that for the CM because S2 has less SVC layers than S1. If the number of SVC layers to merge, computed from Eq. (4), is two for example, SSG will copy EL-2 and EL-3 packets of S1 and merge them to S2. Similarly, this merging process is repeatedly operated on a frame-by-frame basis by SSG; a frame will not be forwarded until it is completely

constructed. As it can be observed from Figure 6, video frames 4, 5, and 6 of S2 are constructed and forwarded in a sequence. It is interesting to notice that two streams are automatically synchronized after the DM buffering process.

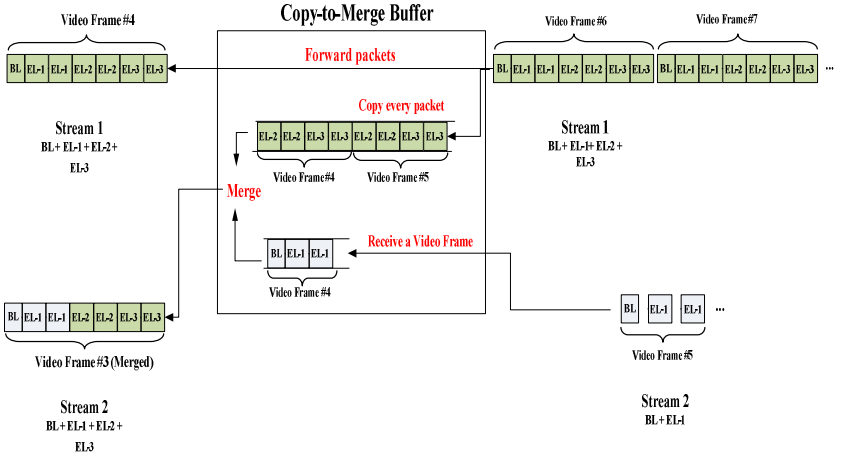



**Fig. 5.** CM buffering at SSG

Finally, Figure 7 shows the operation procedure of stream merging at SSG. After intercepting client requests and bit rate maps (BRM), SSG begins to initiate two processes concurrently; one process to determine which stream is faster and the other process to monitor and compute the available bandwidth (AB). When S2, the one to be merged, is slower than S1, the one to contribute, CM buffering is applied; otherwise DM buffering is applied.

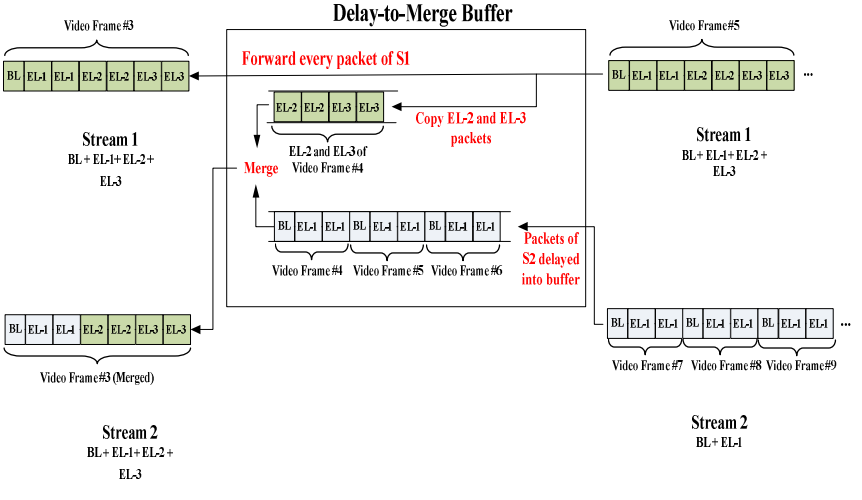



**Fig. 6.** DM buffering at SSG

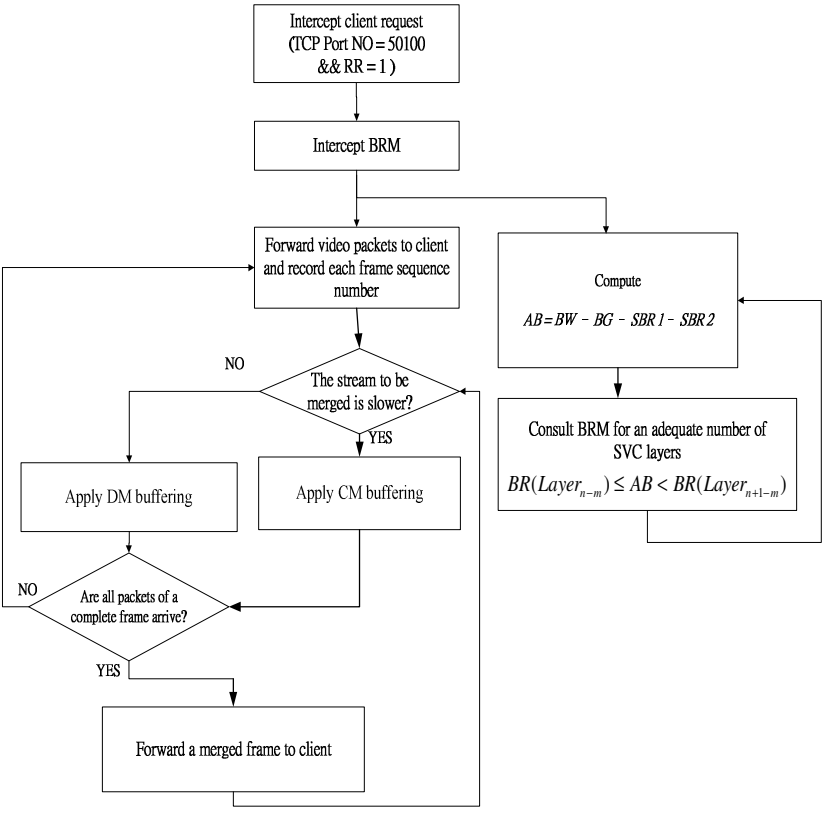



**Fig. 7.** Procedure of stream merging

# 4 Experiments and Analyses

## 4.1 Experiments of Stream Splitting

Figure 8 shows the heterogeneous topology for stream splitting. Here, we assume client 1 requests an SVC film with four layers (BL, EL-1, EL-2, and EL-3) from an IPTV server through intermediate nodes, including an Ethernet switch, two hubs, SSG, two subscriber stations (SS), and a WiMAX basestation (BS). When SSG discovers that client 2 requests the same film, it invokes the stream splitting process. Hence, two SVC layers (BL and EL-1) are split from stream 1 to stream 2. Iperf traffic generator is used to inject different background traffic from a hub on Ethernets. Figure 9 shows the comparison of bandwidth variations on WiMAX with and without stream splitting. In this experiment, we assume client 1 requests an SVC stream with four layers at time=0. At time=100, client 2 requests the same movie stream. As it is observed from Figure 9(a), since stream splitting is not applied, the utilized bandwidth has reached the maximum capacity of WiMAX (8 Mbps). On the other hand, in

Figure 9(b), since stream splitting is applied, the utilized bandwidth remains unchanged (about 604,014 bytes/sec or 5 Mbps).

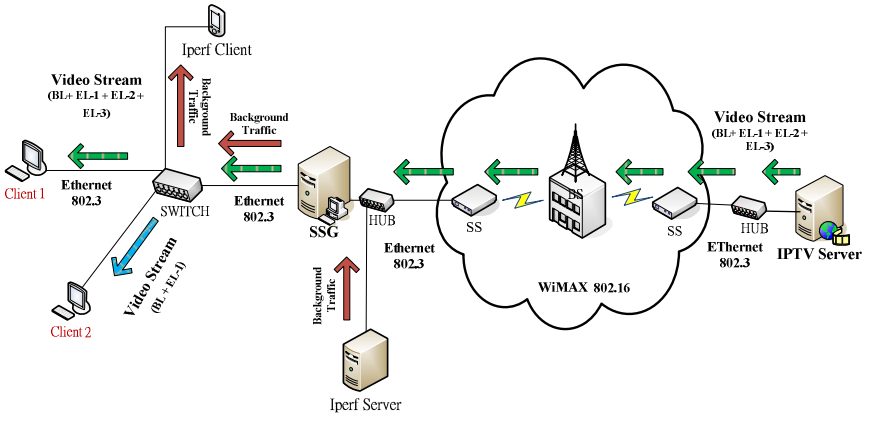
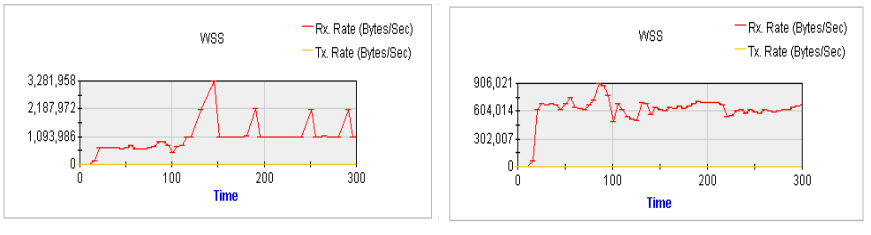



**Fig. 8.** Stream splitting over heterogeneous networks

Next, by increasing the background traffic from 24 Mbps to 26 Mbps, we observe the degradation of video quality in stream 2. As shown in Figures 10(a) and 10(b), video quality of (a) is much better than that of (b), since in the former four SVC layers (BL to EL-3) can be split from stream 1 to stream 2, while in the latter only one layer (BL) can be split.




(a) Without stream splitting (b) With stream splitting

**Fig. 9.** Bandwidth saving on WiMAX

(a) Background traffic=24 Mbps (b) Background traffic=26Mbps

**Fig. 10.** Quality degradation in stream 2

### 4.2 Experiments of Stream Merging

As shown in Figure 11, in the experiment of stream merging, we set up two IPTV servers on the WiMAX network for two clients (clients 1 and 2) to request their streams, respectively. The scenario is purposely designed such that client 1 can receive six layers (BL to EL-5) from server-1 while client 2 receives only one layer (BL) from server 2.

Figure 12 shows the difference of incurred delay between merging one layer and merging five layers for the CM buffering. In the beginning, two curves have slight difference. However, they begin to separate gradually and finally the delay difference reaches about 500 ms. This is because the queuing delay (processing time plus waiting time) increases more quickly in the case of merging five layers than that of merging only one layer. Figure 13 shows the variations of queue length in the CM buffering. Figure 13 can validate the results of Figure 12. In the experiment, queue length is measured at SSG for every minute. It can be observed that the number of packets queued in the CM buffer for the case of merging five layers is about 6 to 7 times bigger than the case of merging just one layer.

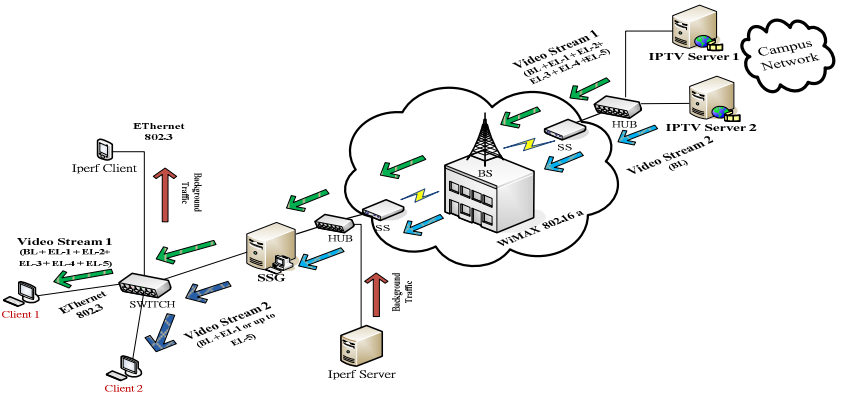



**Fig. 11.** Stream merging over heterogeneous networks

Figure 14 shows the DM buffering delay for the situation when the stream (i.e., S2) to be merged arrives at SSG earlier than the stream (i.e., S1) to contribute. By comparing the DM buffering delay to the CM buffering delay, we observe a significant difference; delay in the former (about 10.5 sec) is relatively smaller than the delay in the latter (about 11.5 sec). This is because in the CM buffering, all the six-layer packets of S1 are required for buffering, which naturally increases the queuing delay. The results shown in Figure 15 can validate the above logical explanation. In the case of merging five layers, we observe that the number of packets queued in the DM buffer is around 300 to 500 packets only, which is significantly smaller than the packets queued in the CM buffer (around 6,000 to 8,000 packets), as shown in Figure 13. Additionally, as we observe from Figure 15, the number of queued packets of merging five layers exhibits only a slight difference than that of merging one layer. This is because in the DM buffering scheme, S2 packets are required for queuing, not S1

packets. Since S2 has only one layer (BL) in this experiment, no matter how many layers of S1 are merged to S2, the number of packets stored in the DM buffer remains nearly no change.

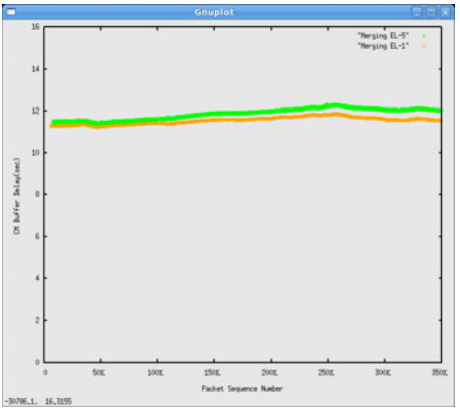



**Fig. 12**. CM buffering delay

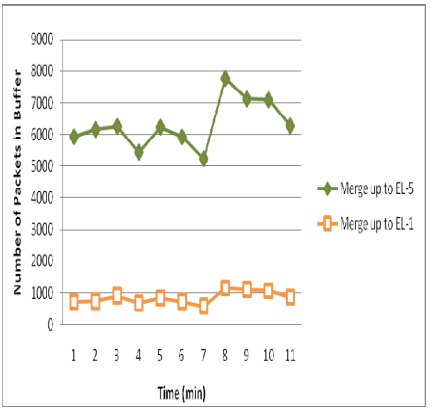



**Fig. 13.** Queue length in CM buffering

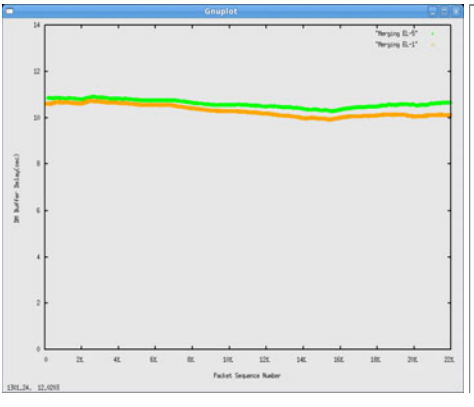



**Fig. 14.** DM buffering delay

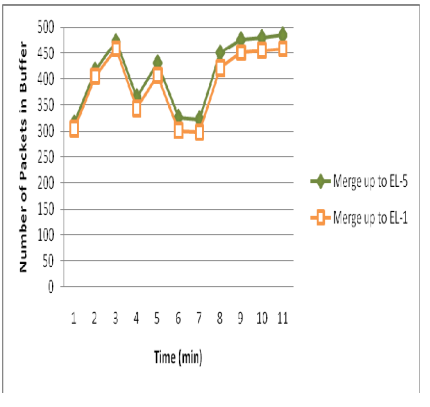



**Fig. 15.** Queue length in DM buffering

## 5 Conclusion

In this paper, we have presented an SSM algorithm by fully utilizing multi-layer characteristics of SVC. The proposed SSM aims at effectively exploiting network bandwidth using two stream manipulation techniques. In stream splitting, adequate SVC layers for splitting can be determined from periodical measurement of network bandwidth. In stream merging, a certain number of enhancement layers of one stream can be merged to the top of base layer of another stream. When two streams arrive at SSG at different speeds, the CM and DM buffering schemes were designed respectively to merge the packets of the same video frame of two different streams. For the purpose

of performance evaluation, the proposed SSM mechanisms were implemented on Linux platform. In the experiments set up on a wired and wireless heterogeneous network, buffering delay and queuing length were analyzed. By comparing the queuing delay between the CM and the DM buffers, we observed that the former is about one second larger than the latter. The difference in queuing delay was validated by the measurement of queuing lengths. The results revealed that the number of packets queued in the CM buffer is about 20 times bigger than the number of packets queued in the DM buffer.

## References


1. Wiegand, T., Sullivan, G.J., Bjøntegaard, G., Luthra, A.: Overview of the H.264/AVC Video Coding Standard. IEEE Transactions on Circuits and Systems for Video Technology 13(7), 560–576 (2003)
2. Zhang, L.-N., Yuan, C., Zhong, Y.-Z.: A Novel SVC VoD System with Rate Adaptation and Error Concealment over GPRS/EDGE Network. In: Congress on Image and Signal Processing, May 27-30 (2008)
3. Chiang, J.-C., Lo, H.-F., Lee, W.-T.: Scalable Video Coding of H.264/AVC Video Streaming with QoS-Based Active Dropping in 802.16e Networks. In: Proceedings of the 22nd International Conference on Advanced Information Networking and Applications, March 25-28 (2008)
4. Zhang, L.-N., Yuan, C., Zhong, Y.-Z.: HandVoD: A Robust and Scalable VoD So-lution with Raptor Codes over GPRS/EDGE Network. In: IEEE International Circuits and Systems for Communications, May 26-28 (2008)
5. Kofler, I., Kuschnig, R., Hellwagner, H.: Improving IPTV Services by H.264/SVC Adaptation and Traffic Control. In: IEEE International Broadband Multimedia Systems and Broadcasting, May 13-15 (2009)
6. Zhao, Y., Eager, D.-L., Vernon, M.-K.: Scalable On-Demand Streaming of Nonli-near Media. IEEE Transactions on Networking 15(5), 1149–1162 (2007)
7. Griwodz, C.: Movie Placement in a Hierarchical CDN with Stream Merging. In: Multimedia Computing and Networking 2004 (MMCN 2004), January 21-22, vol. 5305, pp. 1–15 (2004)
8. Jin, H., Deng, D.: HHMSM: A Hierarchical Hybrid Multicast Stream Merging Scheme for Large-scale Video-on-demand Systems. In: International Conference on Multimedia and Expo, July 6-9, vol. 2, pp. 305–308 (2003)
9. Wu, H., Liu, Y., Zhang, Q., Zhang, Z.-L.: SoftMAC: Layer 2.5 Collaborative MAC for Multimedia Support in Multihop Wireless Networks. IEEE Transactions on Mobile Computing 6(1), 12–25 (2007)
10. Mao, S., Bushmitch, D., Narayanan, S., Panwar, S.-S.: MRTP: a multiflow real-time transport protocol for ad hoc networks. IEEE Transactions on Multimedia 8(2), 356–369 (2006)
11. Kurutepe, E., Civanlar, M.R., Tekalp, A.M.: Client-Driven Selective Streaming of Multiview Video for Interactive 3DTV. IEEE Transactions on Circuits and Systems for Video Technology 17(11), 1558–1565 (2007)
12. Lin, C.-M., Zao, J.-K., Peng, W.-H., Hu, C.-C., Chen, H.-M., Yang, C.-K.: Band-width Efficient Video Streaming Based Upon Multipath SVC Multicasting. In: Wireless Communications and Mobile Computing Conference, August 6-8 (2008)

# Author Index