Yevgeni Koucheryavy
Lefteris Mamatas
Ibrahim Matta
Vassilis Tsaoussidis (Eds.)

Wired/Wireless Internet Communication

LNCS 7277

10th International Conference, WWIC 2012
Santorini, Greece, June 2012
Proceedings

Springer

Lecture Notes in Computer Science 7277

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

Yevgeni Koucheryavy Lefteris Mamatas
Ibrahim Matta Vassilis Tsaoussidis (Eds.)

Wired/Wireless Internet Communication

10th International Conference, WWIC 2012
Santorini, Greece, June 6-8, 2012
Proceedings

Springer

Volume Editors

Yevgeni Koucheryavy
Tampere University of Technology
Department of Communications Engineering
Korkeakoulunkatu 10, 33720 Tampere, Finland
E-mail: yk@cs.tut.fi

Lefteris Mamatas
University College London
Department of Electronic and Electrical Engineering
Torrington Place, London WC1E 7JE, UK
E-mail: lmamatas@ee.ucl.ac.uk

Ibrahim Matta
Boston University
Computer Science Department
111 Cummington Street, MCS-271, Boston, MA 02215, USA
E-mail: matta@bu.edu

Vassilis Tsaoussidis
Democritus University of Thrace
Department of Electrical and Computer Engineering
Building A, Panepistimioupolis Kimmeria, Xanthi 67100, Greece
E-mail: vtsaousi@ee.duth.gr

ISSN 0302-9743 e-ISSN 1611-3349
ISBN 978-3-642-30629-7 e-ISBN 978-3-642-30630-3
DOI 10.1007/978-3-642-30630-3
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2012938230

CR Subject Classification (1998): C.2, H.4, D.4.4, H.3.5, I.2, D.2, H.5, K.6.4

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

The 10^{th} International Conference on Wired / Wireless Internet Communications (WWIC) took place in the magnificent island of Santorini during June 6–8, 2012. Previous events were held in Vilanova i la Geltrú (Spain) in 2011, Lulea (Sweden) in 2010, Twente (The Netherlands) in 2009, Tampere (Finland) in 2008, Coimbra (Portugal) in 2007, Bern (Switzerland) in 2006, Xanthi (Greece) in 2005, Frankfurt-Oder (Germany) in 2004 and Las Vegas (USA) in 2002. This year's event was organized and partially supported by the Space Internetworking Center (SPICE), Democritus University of Thrace in Greece.

The conference attracted 50 submissions from 22 different countries in Asia, Europe, North America and Africa. All submissions were subject to thorough review work by the Technical Program Committee members and additional reviewers. The committee, based on the final paper ranking, accepted 23 papers that were ranked first as regular papers to be presented in different thematic sessions and the next six papers (above threshold) as short papers to be presented in a work-in-progress session. All papers satisfied the criteria of novelty, presentation, relevance, and in addition, demonstrated clear potential for significant impact; therefore, all 29 papers were included in the proceedings, as part of the *Lecture Notes in Computer Science* series.

The papers were organized into six thematically distinct technical sessions, covering the following major topics: Virtual Networks and Clouds, Multimedia Systems, Wireless Sensor Networks and Localization, Delay-Tolerant and Opportunistic Networks, Handover Techniques and Channel Access and Ad hoc Networks. The work-in-progress papers were presented in a dedicated session. Furthermore, an additional session with invited papers in the area of Delay- and Disruption-Tolerant Networks was organized.

In WWIC 2012, we particularly emphasized on areas such as Delay- and Disruption-Tolerant Networks (DTNs) and Opportunistic Networks with a dedicated technical session, an invited session, one keynote speaker and a very interesting panel discussion. All events in the area of DTNs were kindly supported and organized by the Space Internetworking Center (SPICE) project.

WWIC 2012 was pleased to host the joint ERCIM eMobility and MobiSense workshops, which took place on June 9, 2012.

We would like to thank the authors for choosing to submit their results to WWIC 2012, all the members of the Technical Program Committee, the members of the Organizing Committee, as well as all the additional reviewers for their effort to provide detailed and constructive reviews. We are grateful to the two keynote speakers, Eytan Modiano and Jeorg Ott, for accepting our invitation; and to Springer LNCS for their long-term support. Last but not least, we would like to thank the companies Erasmus SA (www.erasmus.gr) and

LiveEvents (www.liveevents.tv) for providing organizational support and live Internet broadcasting services, respectively.

We hope that all participants enjoyed the technical and social conference program, the hospitality of our Greek hosts and the beauty of the conference location. Next year's conference will take place in St. Petersburg, Russia. We hope to see you there again.

June 2012

Vassilis Tsaoussidis
Yevgeni Koucheryavy
Ibrahim Matta
Lefteris Mamatas

Organization

Steering Committee

Torsten Braun	University of Bern, Switzerland
Georg Carle	TU München, Germany
Geert Heijenk	University of Twente, The Netherlands
Yevgeni Koucheryavy	Tampere University of Technology, Finland
Peter Langendorfer	IHP Microelectronics, Germany
Ibrahim Matta	Boston University, USA
Vassilis Tsaoussidis	Democritus University of Thrace, Greece

Conference Chairs

General Co-chairs

Vassilis Tsaoussidis	Democritus University of Thrace, Greece
Yevgeni Koucheryavy	Tampere University of Technology, Finland

TPC Co-chairs

Ibrahim Matta	Boston University, USA
Lefteris Mamatas	University College London, UK

Technical Program Committee

Ozgur B. Akan	Koc University, Turkey
Onur Altintas	Toyota InfoTechnology Center, Japan
Brice Augustin	University of Paris est Creteil (UPEC), France
Khalid Al-Begain	University of Glamorgan, UK
Hans van den Berg	TNO ICT / University of Twente, The Netherlands
Fernando Boavida	University of Coimbra, Portugal
Thomas Michael Bohnert	SAP Research, Switzerland
Torsten Braun	University of Bern, Switzerland
Wojciech Burakowski	Warsaw University of Technology, Poland
Scott Burleigh	NASA JPL, USA
Maria Calderon	Universidad Carlos III de Madrid, Spain
Georg Carle	TU München, Germany
Paulo Martins Carvalho	University of Minho, Portugal
Arsenia Chorti	Princeton University, USA
Stylianos Dimitriou	Democritus University of Thrace, Greece
Vasilis Friderikos	Kings College, UK

Jarmo Harju	Tampere University of Technology, Finland
Sonia Heemstra de Groot	TU Delft, The Netherlands
Geert Heijenk	University of Twente, The Netherlands
Andreas Kassler	Karlstad University, Sweden
Yevgeni Koucheryavy	Tampere University of Technology, Finland
Peter Langendoerfer	IHP Microelectronics, Germany
Remco Litjens	TNO ICT, The Netherlands
Pascal Lorenz	University of Haute Alsace, France
Christian Maihfer	Daimler AG, Germany
Lefteris Mamatas	University College London, UK
Xavi Masip-Bruin	Universitat Politecnica de Catalunya, Spain
Ibrahim Matta	Boston University, USA
Rob van der Mei	Centre for Mathematics and Computer Science, The Netherlands
Abdelhamid Mellouk	University of Paris est Creteil (UPEC), France
Paulo Mendes	University of Lusofona, Portugal
Edmundo Monteiro	University of Coimbra, Portugal
Liam Murphy	University College Dublin, Ireland
Ioanis Nikolaidis	University of Alberta, Canada
Guevara Noubir	Northeastern University, USA
Evgeny Osipov	Lulea University of Technology, Sweden
Panagiotis Papadimitriou	Leibniz University of Hannover, Germany
Giorgos Papastergiou	Space Internetworking Center, Greece
George Pavlou	University College London, UK
Ioannis Psaras	University College London, UK
Guy Pujolle	Pierre et Marie Curie University (Paris 6), France
Dimitrios Serpanos	University of Patras, Greece
Vasilios Siris	Athens University of Economics and Business/FORTH-ICS, Greece
Dirk Staehle	University of Wurzburg, Germany
Burkhard Stiller	University of Zurich and ETH Zurich, Switzerland
Vassilis Tsaoussidis	Democritus University of Thrace, Greece
Ageliki Tsioliaridou	Space Internetworking Center, Greece
Miki Yamamoto	Kansai University, Japan
Marcelo Yannuzzi	Universitat Politecnica de Catalunya, Spain
Eiko Yoneki	Cambridge University, UK
Chi Zhang	Juniper Networks, USA

Organizing Committee

Publicity Chair

Panagiotis Papadimitriou	Leibniz University of Hannover, Germany

Web Chair

Stylianos Dimitriou	Democritus University of Thrace, Greece

Proceedings Chair

Ageliki Tsioliaridou	Democritus University of Thrace, Greece

Members

Nikolaos Bezirgiannidis	Democritus University of Thrace, Greece
Sotirios Diamantopoulos	Democritus University of Thrace, Greece
Giannis Komnios	Democritus University of Thrace, Greece
Sotirios-Angelos Lenas	Democritus University of Thrace, Greece
Spiros Lianos	Erasmus, Greece
Pinelopi Mitrogianni	Erasmus, Greece
Agapi Papakonstantinou	Democritus University of Thrace, Greece
George Papastergiou	Democritus University of Thrace, Greece
Fani Tsapeli	Democritus University of Thrace, Greece
Niki Tsiamaki	Erasmus, Greece

Additional Reviewers

Nidal AlBeiruti
Baris Atakan
Stylianos Basagiannis
A.Ozan Bicen
Diego Borsetti
Wei Koong Chai
Marinos Charalambides
Desislava Dimitrova
Khaled Dridi
Roman Dunaytsev
Jasper Goseling
Stephan Günther
William Ivancic
Vasileios Karyotis
Murat Kocaoglu
Ioannis Komnios
Sotirios-Angelos Lenas
Christos Liaskos
Souzana Makaratzi
Andreas Müller
Michael Müller
Heiko Niedermayer
Cristian Olariu
Vasco Pereira
Andrè Rodrigues
Felix Schmidt-Eisenlohr
Keith Scott
Renè Serral-Gracià
Ricardo Silva
Christina Thorpe
Lloyd Wood
Baoxian Zhang

Table of Contents

Session 1: Virtual Networks and Clouds

Session 2: Multimedia Systems

Session 3: Wireless Sensor Networks and Localization

Session 4: Ad Hoc Networks

Session 5: Handover Techniques and Channel Access

Session 6: Delay-Tolerant and Opportunistic Networks

Work-in-Progress Session

Invited Session: Delay- and Disruption-Tolerant Networks (DTNs): Scenarios, Applications and Protocols

VNEMX: Virtual Network Embedding Test-Bed Using MPLS and Xen

Sarang Bharadwaj Masti[1], Siva P. Meenakshi[2], and Serugudi V. Raghavan[3]

[1] IIT Madras, Chennai - 36, India
sarang@cse.iitm.ac.in
[2] IIT Madras, Chennai - 36, India
spmeena@cse.iitm.ac.in
[3] IIT Madras, Chennai - 36, India
svr@cse.iitm.ac.in

Abstract. Network virtualization has received considerable attention by the network research community in the past few years as a means of overcoming the "Internet ossification" problem. It provides a smooth deployment path for new architectures and allows multiple virtual networks to co-exist on the same substrate network by sharing the substrate network resources. One of the main challenges in network virtualization is the efficient allocation of substrate network resources to the virtual networks, a problem known as Virtual Network Embedding(VNE). A number of algorithms for VNE exist in the literature. In this paper, we propose $VNEMX$, a test-bed for comparing and evaluating VNE algorithms. We demonstrate the viability of using MPLS along with Xen to create a test-bed on which virtual networks can be deployed and tested. We also evaluate the proposed architecture for the test-bed using metrics such as virtual network creation time, transmission capability of the virtual links, isolation between flows and cpu utilization.

1 Introduction

The Internet has now become a vital component of many parts of our life such as work, education, entertainment etc. It has become one of the most important technologies in the recent years. However, this widespread deployment and popularity of Internet has created an ossifying force that precludes the deployment of new architectures and technologies, thus inhibiting further development. This condition, where further developments in the Internet are hindered because of its widespread popularity, is known as "Internet ossification". The Internet that was created several years ago has retained its basic architecture over these years, in spite of there being developments in terms of its size, speed, link technologies and applications. The main reasons for this are (1) widespread deployment (2) existence of multiple-stakeholders, with conflicting goals and policies that make it impossible to modify the basic architecture or to deploy a new architecture. Thus, to overcome this impasse and to foster innovation and development, network virtualization has been proposed as a solution [3].

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 1–12, 2012.

Network virtualization enables multiple network instances to co-exist on the same physical network by allowing the network instances to share the underlying resources [11,15]. It also provides a test-bed for evaluating new architectures [8,1,4] and a smooth deployment path for deploying these new architectures. One of the main challenges in network virtualization is the efficient allocation of substrate network resources to the virtual networks, a problem known as Virtual Network Embedding(VNE). Unfortunately, VNE has been shown to be $NP - Hard$ by a reduction from multiway separator problem [2]. Hence, most of the past research in this field has been directed towards finding heuristics to achieve near optimal solutions [17,13,12,16,7,10]. Up until now, only simulators have been used to compare and evaluate these algorithms. A test-bed would present a much more realistic testing environment, thus lending more credibility to the experimental results.

In this paper, we propose $VNEMX$, a test-bed for comparing and evaluating VNE algorithms. We demonstrate the viability of using MPLS along with Xen to create a test-bed on which virtual networks can be deployed and tested. We also evaluate the proposed architecture for the test-bed using metrics such as virtual network creation time, transmission capability of the virtual links, isolation between flows and cpu utilization. The implementation of the centralized controller with an interface to communicate with VNE algorithms under evaluation, a middle ware to interconnect Xen with MPLS, RSVP like resource reservation protocol that is extended for MPLS-Xen based Virtual Networks and performance analysis of the embedded virtual networks are our contributions.

The rest of the paper is organized as follows. The related work is mentioned in section 2. A brief overview of the proposed architecture for the test-bed is given in section 3. Sections 4 and 5 respectively give the protocols for resources discovery, allocation and deallocation. A description of the implementation details of our test-bed prototype is given in section 6 followed by evaluation results in section 7 and conclusion in section 8.

2 Related Work

Network virtualization can be used to create test-beds to evaluate future internet architectures. Several architectures for test-bed networks have been proposed in the past, and the chief ones among them include Planet Lab, GENI, Trellis and VINI. Planet Lab [8] is a geographically distributed, large scale test-bed designed for deploying and evaluating network services. Each network service runs in isolation and receives a slice of the platform i.e. a portion of the node's resources. VINI [4], a virtual network infrastructure, allows researchers to evaluate their protocols and architectures in a controlled and realistic environment. It allows multiple experiments to be run simultaneously. It also allows the creation of virtual topologies. [5] is also a software platform for running virtual networks on shared hardware. In order to achieve high performance it uses container-based virtualization for nodes and a new tunneling technique, EGRE, for the virtual links. GENI [1] is a large-scale, realistic experiment facility for evaluating

new architectures and test-beds. Like other test-bed architectures, GENI allows multiple experiments to be run in parallel.

CABO [11] proposes an architecture that achieves a clear separation among the infrastructure providers and service providers. They show that combining the roles of the infrastructure provider and service provider impedes the deployment of new technologies and architectures and thus propose an architecture to enable a separation. Cabernet [18] proposes an architecture that enables easy deployment of wide-area network services, by separating the business roles into three layers - connectivity, infrastructure and service layers. It facilitates simplified service deployment by abstracting the negotiations with different infrastructure providers using the connectivity layer. [14] propose a network virtualization architecture based on four main business players - Infrastructure Provider, Virtual Network Provider, Virtual Network Operator and Service Provider thus creating a clear separation between the infrastructure provider and service provider. The architecture was designed to enable resource sharing among various stakeholders. [6] study the various issues and challenges that may arise in a commercial environment using the 4WARD network virtualization architecture [9] as the reference. They look at issues such as scalability, isolation, manageability, on-demand provisioning etc. from the perspective of an infrastructure provider.

In our work, we have developed an MPLS and Xen based architecture for a test-bed that could be used to deploy virtual networks. This testbed serves as a platform to compare and analyze the performance of VNE algorithms as well as to analyze the performance of MPLS-Xen based VNs.

3 Architecture Overview

In this section, we give a brief description of the proposed architecture for the test-bed. Figure 1 shows an example substrate network that would be maintained by the infrastructure provider. As shown in the above figure, the MPLS network consists of core nodes(aka. Label Switch Routers) that are responsible for switching packets depending on the label attached to the packets and the edge nodes that are responsible for attaching/removing the labels. The ingress edge node, classifies the incoming packets and attaches labels to packets before forwarding them to the core nodes. Similarly, the egress edge node, removes the attached label and forwards the packet to the destination. The edge nodes in addition to packet forwarding functions shall also host the virtual machines. Each of these virtual machines correspond to a virtual node belonging to some virtual network embedded on the substrate network. The bandwidth of substrate links is used to support the virtual links. One of the edge nodes on the substrate network also acts as a centralized controller. The centralized controller is responsible for making admission decisions and also managing the resources(node and link) on the substrate network. The main tasks of the centralized controller are as follows - (i) Receive virtual network requests that have to be admitted. (ii) Run the admission algorithm(VNE algorithms) to determine the feasibility of admitting the request. (iii) If the request can be admitted, send messages to

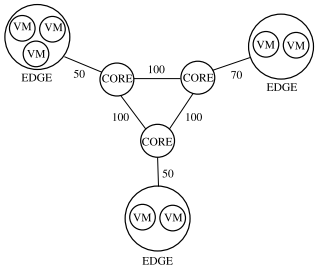


Fig. 1. Substrate network

appropriate substrate nodes to reserve resources for the virtual nodes and virtual links. (iv) When a virtual network request expires, send messages to release resources allocated to the virtual network on the substrate nodes and links.

In order to determine the feasibility of admitting the virtual network request, the centralized controller runs one of the algorithms for Virtual Network Embedding(VNE) [17,13,12,16,7,10]. These algorithms need to be aware of the residual node and link resources available on the substrate network. Thus, it is necessary to have a resource discovery protocol that can be used by centralized controller to determine the resource availability on the substrate network, which is explained in detail in section 4. Also, when the virtual network has to be admitted, resources have to be reserved on the substrate nodes and links for the virtual network. Hence, there is a need for a resource reservation protocol, which is explained in detail in section 5.

4 Resource Discovery

The VNE algorithms that run on the centralized controller need to be aware of the substrate network topology and the resources availability on the substrate nodes and links in order to determine the feasibility of mapping virtual network requests. Messages have to exchanged between the substrate nodes and the centralized controller periodically so that the controller has the knowledge of the entire substrate network. This can be achieved by using the Link State Routing protocol and including the information about resource availability on the substrate nodes and links. Link State Routing protocol is one of the widely used routing protocols in packet-switching networks. The protocol works in two stages: (i) Discovering neighbors - Each node tries to determine all its neighbors by using a simple reachability protocol. (ii) Distributing the information about links - After each node has determined its neighbors, it constructs a link-state advertisement message which contains the identifier of the node producing the message(Eg. IP address), the nodes to which it is connected to directly and a sequence number which is increased each time a new message is constructed. This link-state information is then flooded throughout the network.

Using the link-state information sent by all the other nodes in the network, any node can determine the entire network topology. Thus in our case, to distribute

information about the resource availability, each node when constructing the link-state message will include the following additional information in addition to the ones described previously - (i) The residual CPU capacity of the substrate node. (ii) The residual bandwidth of each of the links connected to the node. With this additional information, the centralized controller can determine the substrate network topology and the available resources on substrate nodes and links.

5 Resource Reservation

The VNE algorithm running on the centralized controller determines the feasibility of mapping the virtual network request. If the requests can be mapped, it gives the substrate nodes that can be used to support the virtual nodes and also the paths on the substrate network that can be used to support the virtual links. In order to inform appropriate substrate nodes to reserve resources for the virtual nodes and to reserve bandwidth along the substrate paths for the virtual links, the centralized controller will need to send resource reservation messages. Similarly, when a virtual network request expires the centralized controller needs to send messages to appropriate substrate nodes so that, the nodes can release the resources allocated to the virtual network that has expired. This can be achieved by using a resource reservation protocol like RSVP.

- **Virtual Node Creation:** The centralized controller, after determining the substrate node that can be used to support the virtual node, sends an NRESV messages to the *noded* daemon running on the corresponding substrate node. The NRESV message contains the following information - (i) SNID - The substrate node to which the message is destined. (ii) VNID - Unique identifier associated with the Virtual Network. (iii) NID - Unique node identifier identifying the virtual node belonging to the virtual network. (iv) CAP - The amount of CPU capacity that has to be reserved on the substrate node for the virtual node. When the *noded* daemon receives the NRESV message, it creates the virtual node and sends back an acknowledgment to the centralized controller.
- **Virtual Link Creation:** In order to reserve bandwidth resources along the substrate paths for the the virtual link, the centralized controller creates a LRESV message. For each virtual link, which is identified by virtual nodes on either ends, the centralized controller creates a LRESV message and send this message to the *linkd* daemon running one of the substrate nodes supporting the corresponding virtual nodes. The format of the LRESV message is as follows - (i) VNID - Unique identifier associated with the Virtual Network. (ii) VLINK - Described by the virtual nodes on either sides of the virtual link in the virtual network. (iii) BW - The amount of bandwidth that needs to be reserved along the path for the virtual link. (iv) PATH - Gives the path along which the resources have to be reserved for the virtual link. It is a list of substrate nodes with the first and the last substrate node corresponding to the substrate nodes supporting the virtual nodes on either

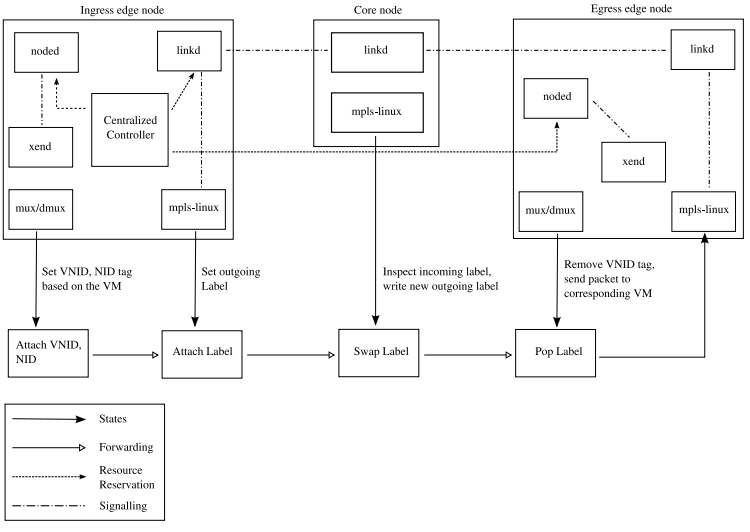


Fig. 2. Block diagram of the system

ends of the virtual link. When the substrate node gets the LRESV message, it updates the forwarding table to indicate the interface along which the traffic belonging to the virtual link has to forwarded. It then removes the first node from the list of nodes in the PATH field and forwards the LRESV message to the next node on the path. Once the path has been established, the *linkd* daemon send an acknowledgment back to the centralized controller. In a similar manner, when resources allocated to a virtual network have to be released, messages are sent to the appropriate substrate nodes to release these resources.

6 Implementation Details

In this section, we give a brief description of the prototype that we implemented. Figure 2 gives a block diagram showing the various components of the system. We used Xen for the creation of virtual machines and mpls-linux kernel module for enabling MPLS on the substrate nodes. A brief description of the various modules is given below:

- **noded** - The *noded* daemon, as explained earlier in section 5, is responsible for creating virtual nodes. Upon receiving the NRESV message form the centralized controller, the daemon interacts with *xend* and ensures that a virtual machine is created.
- **linkd** - The *linkd* daemon is responsible for the creation of virtual links. Upon receiving the LRESV message, the *linkd* daemon interacts with *mpls-linux* to update the MPLS related forwarding tables and forwards the LRESV message to the next substrate node along the path as explained in section 5.

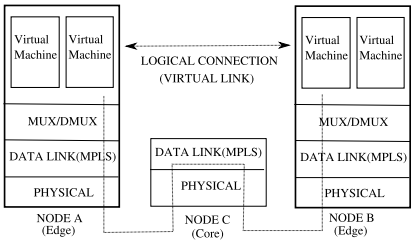


Fig. 3. Routing of packets between VMs

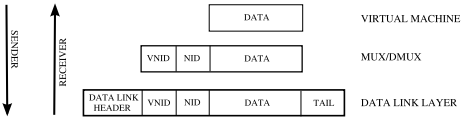


Fig. 4. Headers added to the packet as they move through the protocol stack

- **xend** - This acts as an interface between *noded* and the Xen hypervisor, receiving commands for the creation and destruction of virtual machines.
- **mpls-linux** - This is the kernel module providing the MPLS packet forwarding capability.
- **mux/dmux** - This module is responsible for multiplexing the packets from various virtual machines(VMs) on the sender side and de-multiplexing the packets and delivering them to the appropriate VMs on the receiver side.

6.1 Data Transfer between Virtual Machines

We consider the following scenario to describe the communication taking place between the VMs. A VM, i.e., vm1 (sender) on the edge node A is communicating with a VM i.e., vm2 (receiver) on the edge node B as shown in figure 3. The virtual link between vm1 and vm2 has been allocated along the path A-C-B. When vm1 sends a packet to vm2 the *mux/dmux* module on node A captures the packets and attaches a tag that gives both the VNID(virtual network ID) and the NID(virtual node ID) and passes it on to the *mpls-linux module*. The *mpls-linux* module, encapsulates this packet within an MPLS packet with an MPLS label that identifies the virtual link between vm1 and vm2. The MPLS packet is then routed along the path A-C-B by the intermediate nodes until it finally reaches node B. Now, at node B the *mpls-linux* module strips off the MPLS header and hands over the packet to the *mux/dmux* module that determines the virtual network to which the packet belongs to by looking at the VNID tag attached. Using the VNID information the *mux/dmux* module can determine the VM to which the packet is destined, which is vm2 in this example. After stripping off the VNID and NID tag, the *mux/dmux* module forwards the packet to the appropriate VM, thus completing the transfer of packet from vm1 to vm2.

7 Evaluation Results

In order to evaluate the proposed architecture and protocols, we set up a test-bed as shown in figure 5. Six linux boxes were used as substrate nodes. The edge nodes were configured with Xen to create virtual machines. To enable MPLS support, mpls-linux kernel module was installed in all of the machines. The configuration

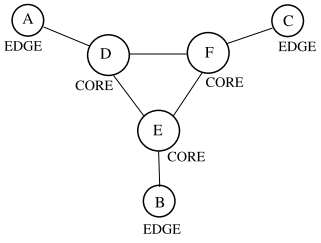


Fig. 5. Test-bed topology

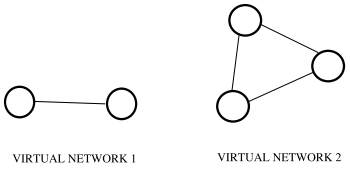


Fig. 6. Virtual networks embedded on the substrate

of the machines used is as follows. All the substrate nodes used Intel Pentium 4 dual core with 3 Ghz speed, as the processor. Edge substrate nodes (A, B and C) had 4 GB main memory where as the core substrate nodes (D, E and F) had a main memory memory of size 2 GB. The hardisk capacity was 60 GB for both core and edge substrate nodes. Fedora core 8 operating system with mpls patched kernel(2.6.26.6-49.fc8.mpls.1.962) was installed on the core nodes. Debian distribution with Xen virtual machine patched kernel(2.6.26-2-xen-686) was installed on the edge nodes. It is important to note that the prototype was developed only as a demonstration of the concept and the evaluation results are specific to our implementation.

7.1 Virtual Network Creation

In order to determine the time required to create virtual networks, we tried to embed virtual networks having the topology as shown in figure 6. Table 1 shows the time required for creating the virtual nodes [1] and virtual links. We observe from the table that the average time required for virtual node creation is around 5-6 seconds. The time required to set up MPLS paths for supporting virtual links is dependent on the length of the path since it involves sending reservation message along the path. Thus, as the length of the path increases the time required to set up the path increases. However, from the above table we observe that the node creation is a much more time consuming process than link creation since link creation just involves updating the tables along the path which can be done relatively fast. The time required to bring up the entire network is approximately equal to the maximum of the virtual node creation times plus maximum of the link creation times, since the creation of virtual network takes place in two phases - virtual node creation and virtual link creation. The centralized controller first sends NRESV messages to all the concerned substrate nodes, each of which try to create the virtual machines in parallel. After receiving the acknowledgments from all the substrate nodes, it sends LRESV messages to set up paths for the links.

[1] We use the term virtual node and virtual machine(VM) interchangeably.

Table 1. Virtual Network Creation Time Measurements

No of Nodes in VN	Virtual Nodes Avg. Creation Time (Sec)	MPLS Paths Avg. Creation Time (Sec)
2	5.5	.0622
3	6.33	.118

7.2 Performance Measurement

In this section, we evaluate the performance of the virtual links as well as the network by transferring data between the virtual machines using the file transfer protocol(FTP). We considered a two node virtual network with one of the virtual nodes configured as the sender and the other configured as the receiver. The first experiment was designed to observe the transfer rates that could be achieved by the configured VMs and MPLS tunnels. The second experiment was used to observe the cpu load on the substrate node and the degree of isolation between flows when multiple flows are active.

Experiment 1. In this experiment we considered a two node embedded virtual network with the virtual nodes on the edge nodes A and C. The virtual link between these virtual nodes was established via the core nodes D and F. The virtual nodes were configured with 128 MB of primary memory and 4 GB of disk-space. The interface configuration of the substrate nodes on the path is as follows.

- Node A - Incoming interface: 100 Mbps; Outgoing interface: 1000 Mbps.
- Node D - Incoming interface: 100 Mbps; Outgoing interface: 1000 Mbps.
- Node F - Incoming interface: 1000 Mbps; Outgoing interface: 1000 Mbps.
- Node C - Incoming interface: 100 Mbps; Outgoing interface: 100 Mbps.

The bandwidth of the virtual link belonging to the virtual network was varied from 1Mbps to 30Mbps. To set the bandwidth of the virtual link, we used the linux command *tc*. Using a token bucket we shaped the traffic coming out of the virtual machine thus ensuring that the traffic doesn't exceed the allotted bandwidth. The token bucket filter was used since it provides accurate shaping and is also scalable with higher bandwidths. Initially, we set the token bucket size to a constant value of 5000 bytes and observed the achieved transfer rate for different bandwidth settings of the virtual link. We transferred a file of size 1.2 Gbits between the virtual nodes on A and C, varied the bandwidth of the virtual link, and recorded the transfer rate and transfer time. For the constant token bucket size of 5000 bytes, the achieved transfer rate is almost equal to the configured rate upto 10 Mbps. After that the achieved transfer rate started to decrease when compared to the configured rate. One of the reasons for this is the insufficient token bucket size for higher bandwidths. In order to achieve equivalent transfer rates at higher configured bandwidth rates, we varied the token bucket size depending on the required bandwidth as suggested in the linux *tc* manual

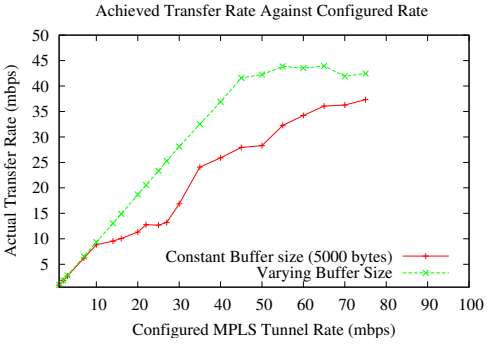


Fig. 7. Comparison of Achieved Transfer Rate Against Configured MPLS Tunnel Rate for FTP traffic

i.e., $MinimumBucketsize = Rate/(HZ\ value\ in\ the\ kernel)$. The comparison of transfer rates and transfer time for constant bucket size and varying bucket sizes is given in figure 7. We can still observe a difference of 0.1 to 1.5Mbps between achieved transfer rate and configured tunnel rate. This is not a limitation of the architecture but a limitation of our prototype implementation. The mux/dmux module was implemented as an user space process that frequently make switches to the kernel mode to transfer packets, thus resulting in decreased performance. Implementing the mux/dmux module as a kernel module would make it much more efficient and would result in transfer rates that is almost equal to the configured rate.

Experiment 2. The second experiment was aimed to measure the cpu load and observe the isolation provided to the flows when multiple virtual machines are transmitting data. We had seven virtual machines running on each of the edge nodes A and C. The virtual links between the virtual machines was established through the code nodes D and F. The bandwidth of the virtual links between the virtual machines was configured as follows:

- Link between vm1 on A and vm1 on C - 1Mbps
- Link between vm2 on A and vm2 on C - 1Mbps
- Link between vm3 on A and vm3 on C - 2Mbps
- Link between vm4 on A and vm4 on C - 2Mbps
- Link between vm5 on A and vm5 on C - 4Mbps
- Link between vm6 on A and vm6 on C - 4Mbps
- Link between vm7 on A and vm7 on C - 8Mbps

A file of size 1.2Gbits was transferred between the virtual machines simultaneously. The cpu load and transfer rate achieved on each VM's interface was monitored using the linux command *sar*, which is a system activity measurement utility. The bandwidth isolation provided for each of the flows is clearly seen in figure 8. The cpu system load variation with time is shown in figure 9.

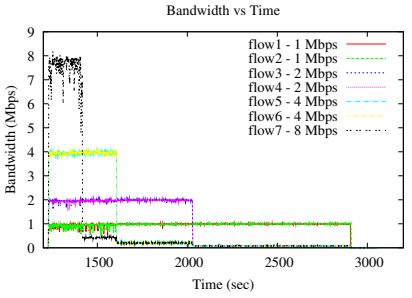


Fig. 8. Isolation of flows in the embedded VNs

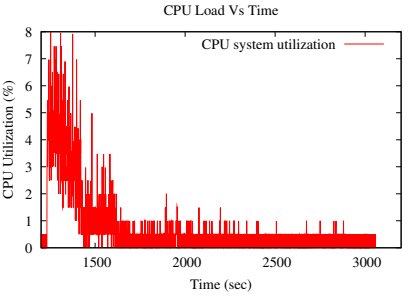


Fig. 9. CPU system load during simultaneous transmission of flows

When all 7 flows are active, the cpu system load peaks to 8 %. The cpu system peak load drops to 3.5 %, 1 % and .5 % as the transfer gets completed.

8 Conclusion

In this paper, we proposed $VNEMX$, a test-bed for comparing and evaluating VNE algorithms. We demonstrated the feasibility of using MPLS and Xen to create a test-bed on which virtual networks can be deployed and tested. The proposed architecture for the test-bed was evaluated using metrics such as virtual network creation time, virtual link transmission capability, cpu load etc. We embedded two node and three node virtual network topologies and measured the time required for the creation of virtual nodes and links. As expected, we observed an increasing trend in creation times when the number of nodes and links increased in the virtual network. We also evaluated the performance of the virtual network and the transmission capability of the virtual links using FTP. We observed that, for bandwidths upto 30Mbps the achieved transmission rate was almost equal to the configured rate. We also showed that in spite of having many flows active at the same time good isolation was maintained between the flows.

References

1. Geni: Global environment for network innovations, http://www.geni.net/
2. Andersen, D.G.: Theoretical approaches to node assignment (2002), http://www.cs.cmu.edu/~dga/papers/andersen-assign.ps
3. Anderson, T., Peterson, L., Shenker, S., Turner, J.: Overcoming the internet impasse through virtualization. Computer 38(4), 34–41 (2005)
4. Bavier, A., Feamster, N., Huang, M., Peterson, L., Rexford, J.: In vini veritas: realistic and controlled network experimentation. SIGCOMM Comput. Commun. Rev. 36(4), 3–14 (2006)
5. Bhatia, S., Motiwala, M., Mühlbauer, W., Valancius, V., Bavier, A., Feamster, N., Peterson, L., Rexford, J.: Hosting virtual networks on commodity hardware (2008)

6. Carapinha, J., Jiménez, J.: Network virtualization: a view from the bottom. In: Proceedings of the 1st ACM Workshop on Virtualized Infrastructure Systems and Architectures, VISA 2009, pp. 73–80. ACM, New York (2009), http://doi.acm.org/10.1145/1592648.1592660
7. Chowdhury, N., Rahman, M., Boutaba, R.: Virtual network embedding with coordinated node and link mapping. In: INFOCOM 2009, pp. 783–791. IEEE (2009)
8. Chun, B., Culler, D., Roscoe, T., Bavier, A., Peterson, L., Wawrzoniak, M., Bowman, M.: Planetlab: an overlay testbed for broad-coverage services. SIGCOMM Comput. Commun. Rev. 33(3), 3–12 (2003)
9. Bauke, S., et al.: Virtualization approach: Concept, 4ward project deliverable 3.1.0 (2009)
10. Farooq Butt, N., Chowdhury, M., Boutaba, R.: Topology-Awareness and Reoptimization Mechanism for Virtual Network Embedding. In: Crovella, M., Feeney, L.M., Rubenstein, D., Raghavan, S.V. (eds.) NETWORKING 2010. LNCS, vol. 6091, pp. 27–39. Springer, Heidelberg (2010), http://dx.doi.org/10.1007/978-3-642-12963-6_3, doi:10.1007/978-3-642-12963-6_3
11. Feamster, N., Gao, L., Rexford, J.: How to lease the internet in your spare time. SIGCOMM Comput. Commun. Rev. 37(1), 61–64 (2007)
12. Lischka, J., Karl, H.: A virtual network mapping algorithm based on subgraph isomorphism detection. In: VISA 2009: Proceedings of the 1st ACM Workshop on Virtualized Infrastructure Systems and Architectures, pp. 81–88. ACM, New York (2009)
13. Lu, J., Turner, J.: Efficient mapping of virtual networks onto a shared substrate. Tech. Rep. WUCSE-2006-35, Washington University (2006)
14. Schaffrath, G., Werle, C., Papadimitriou, P., Feldmann, A., Bless, R., Greenhalgh, A., Wundsam, A., Kind, M., Maennel, O., Mathy, L.: Network virtualization architecture: proposal and initial prototype. In: Proceedings of the 1st ACM Workshop on Virtualized Infrastructure Systems and Architectures, VISA 2009, pp. 63–72. ACM, New York (2009), http://doi.acm.org/10.1145/1592648.1592659
15. Turner, J., Taylor, D.: Diversifying the internet. In: Global Telecommunications Conference, GLOBECOM 2005, vol. 2, pp. 755–760. IEEE (2005)
16. Yu, M., Yi, Y., Rexford, J., Chiang, M.: Rethinking virtual network embedding: substrate support for path splitting and migration. SIGCOMM Comput. Commun. Rev. 38(2), 17–29 (2008)
17. Zhu, Y., Ammar, M.: Algorithms for assigning substrate network resources to virtual network components. In: Proceedings of the 25th IEEE International Conference on Computer Communications, INFOCOM 2006, pp. 1–12 (2006)
18. Zhu, Y., Zhang-Shen, R., Rangarajan, S., Rexford, J.: Cabernet: connectivity architecture for better network services. In: Proceedings of the 2008 ACM CoNEXT Conference, CoNEXT 2008, pp. 64:1–64:6. ACM, New York (2008), http://doi.acm.org/10.1145/1544012.1544076

Towards Large-Scale Network Virtualization

Panagiotis Papadimitriou[1], Ines Houidi[2], Wajdi Louati[2],
Djamal Zeghlache[2], Christoph Werle[3], Roland Bless[3], and Laurent Mathy[4]

[1] Institute of Communications Technology, Leibniz University of Hannover, Germany
panagiotis.papadimitriou@ikt.uni-hannover.de
[2] Institut Telecom, Telecom SudParis, France
{ines.houidi,wajdi.louati,djamal.zeghlache}@it-sudparis.eu
[3] Karlsruhe Institute of Technology, Germany
{werle,bless}@tm.uka.de
[4] Computing Department, Lancaster University, UK
laurent@comp.lancs.ac.uk

Abstract. Most existing virtual network (VN) provisioning approaches assume a single administrative domain and therefore, VN deployments are limited to the geographic footprint of the substrate provider. To enable wide-area VN provisioning, network virtualization architectures need to address the intricacies of inter-domain aspects, i.e., how to provision VNs with limited control and knowledge of any aspect of the physical infrastructure.

To this end, we present a framework for large-scale VN provisioning. We decompose VN provisioning into multiple steps to overcome the implications of limited information on resource discovery and allocation. We present a new resource selection algorithm with simultaneous node and link mapping to assign resources within each domain. We use a signaling protocol that integrates resource reservations for virtual link setup with Quality-of-Service guarantees. Our experimental results show that small VNs can be provisioned within a few seconds.

1 Introduction

The Internet has been experiencing a remarkable emergence of new applications and network services. This is due to the flexibility of the Internet's original design and the resulting ability to act as a carrier for nearly arbitrary services. While the services on top of the Internet evolved, the core of the Internet remained unchanged over the past decades with the exception of small patches. Thereby, the current Internet infrastructure remains sub-optimal for many network applications that require high performance, reliability and/or security.

To overcome this impasse, numerous Future Internet Initiatives [7], [18], [1] leverage on network virtualization to concurrently deploy and operate different network architectures within virtual networks (VNs). Recently, technology has evolved to satisfy the various needs for full network virtualization across multiple administrative domains. Although large-scale VN setup may be technically feasible, VN provisioning and management requires the separation between the

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 13–25, 2012.

network operations and the physical infrastructure. A newly envisioned level of indirection already exists in GENI [7], 4WARD [1], and Cabernet [18].

In this context, *VN Providers* (VNPs) that act as brokers for virtual resources between *VN Operators* (VNOs) and *Physical Infrastructure Providers* (InPs) will have to provision VNs without having control or even knowledge of any aspect of the physical infrastructure (see [15] for VNP and VNO definitions). This entails serious implications on resource discovery and allocation, since InPs will not be willing to disclose their resource and topology information to third parties (i.e., VNPs). Currently, most existing VN embedding approaches are limited to a single administrative domain, assuming complete knowledge of physical resources and the underlying substrate topology. Thereby, such VN deployments are limited to the geographic footprint of the substrate provider. Recent work [6] presents a multi-domain VN embedding framework, where VN requests are relayed across InPs till the embedding is completed. However, this work lacks the required embedding algorithms and their evaluation and therefore, it is unclear how fast it converges to a full embedding.

In this paper, we present a VN provisioning framework that allows VNPs to discover and allocate resources across multiple physical infrastructures with limited information disclosure. Furthermore, we propose a new algorithm with simultaneous node and link mapping for resource assignment within InPs. To provide the required interoperability across InPs for virtual link setup, we use a signaling protocol based on the Next Steps in Signaling (NSIS) framework [8]. Using a prototype implementation, we evaluate the performance of VN provisioning with a diverse size of virtual and substrate networks.

The remainder of the paper is organized as follows. Section 2 gives an overview of the provisioning framework. In Section 3, we present our resource assignment algorithm with simultaneous node and link mapping within InPs. Section 4 discusses our signaling protocol for virtual link setup. Section 5 provides experimental results on VN provisioning and performance analysis of virtual link setup. Finally, in Section 6, we highlight our conclusions and discuss future work.

2 Virtual Network Provisioning Overview

In this section, we provide an overview of our multi-domain VN provisioning framework. In contrast to VNs deployed on top of a single shared substrate, multi-domain VN provisioning is coordinated by VN Providers, which have limited resource and topology information of the underlying substrates. Hence, virtual resources should be initially discovered and matched at a rather high level of abstraction (i.e., VNP) to identify candidate resources from which the most appropriate resources will be selected based on detailed information only available to InPs. Thereby, our approach consists in decomposing VN provisioning into the following steps:

Resource Advertisement: To facilitate resource discovery, resource/service advertisement from the InPs is required. An InP will not expose any information on the substrate topology nor the number of virtual resource instances it

is willing to provide for the construction of VNs. Separate node and link specifications will essentially describe different types of virtual resources that can be offered by the InPs. Hence, the VNP will be in position to match requested with offered virtual resources across InPs. Thereby, we consider the InPs disclosing a set of offered virtual nodes and links accompanied with static attributes (e.g., node/link type, operating system, number of network interfaces, geographic location) and the associated cost. Virtual resource descriptions are formed by extracting the static attributes from resource repositories at the InPs. Upon receiving any advertised information, the VNP registers it into a local repository which is subsequently used for the resource matching step.

Resource Matching: Resource matching is the identification of a set of resources offered by InPs that fulfill the requirements for each individual resource request. We consider a VN request as a weighted undirected graph $G_v = (N_v, L_v)$, where N_v is the set of virtual nodes n_v and L_v is the set of virtual links l_v between nodes of the set N_v. For a given G_v request, VNP determines for each $n_v \in N_v$ the corresponding $Match(n_v)$ set among resources advertised from InPs. This set essentially includes all possible candidates for each requested virtual node.

VN Splitting: Since a $Match(n_v)$ set may include resources from multiple InPs, the VNP has to decide to which InP each requested virtual node n_v should be assigned. Essentially, the VNP should split the VN graph into sub-graphs that will compose the requests for each target InP, while minimizing the cost for allocating all nodes and links across InPs. Previous work [10] provides heuristic and exact methods for VN splitting.

Resource Assignment: Resource assignment relies on a selection process within each InP, since full knowledge of the physical resources and topology is required. Each InP assigns its partial VN to the substrate network using a VN mapping algorithm. Unlike most existing work [17], [16], [12], where node and link mapping is achieved sequentially, we develop a new heuristic resource assignment algorithm with simultaneous node and link mapping. The main objective of this algorithm is to optimize load balancing over substrate nodes. Furthermore, our heuristic algorithm yields faster execution than multi-commodity flow algorithms (e.g., [5]), which is critical for embedding VNs onto large physical infrastructures.

VN Instantiation: Upon resource assignment, the selected substrate resources are allocated by the InPs in order to instantiate the requested VN. Within each InP, VN instantiation is coordinated by a dedicated management node, which signals requests to substrate nodes for virtual node and link setup. Our prototype implementation uses Xen [3] for node/router virtualization and Click Modular Router [11] (running in the Linux kernel) for packet encapsulation/decapsulation and packet forwarding. Each virtual node request (within each InP) is handled by a separate thread, speeding up VN instantiation. Similarly, separate threads allow VN setup to proceed in parallel across InPs. We use NSIS (and particularly the implementation available in [14]) to carry the required information for

virtual link setup across inter-domain paths. Besides providing interoperability, our signaling protocol improves performance since it couples virtual link setup with resource reservation via the QoS Signaling Layer Protocol (NSLP) [13].

3 Resource Assignment

Assigning VN graphs to shared substrate networks is known as an NP-hard problem. Hereby, we propose a heuristic resource assignment algorithm where node and link mapping phases are simultaneously executed in one stage.

3.1 Embedding Model and Problem Formulation

Substrate Network Model: The substrate network can be represented by a weighted undirected graph $G_s = (N_s, L_s)$, where N_s is the set of substrate nodes and L_s is the set of substrate links between nodes of the set N_s. Each substrate node $n_s \in N_s$ is associated with the capacity weight value $C(n_s)$ which denotes the *available* capacity of the physical node n_s. Each substrate link $l_s(i,j) \in L_s$ between two substrate nodes i and j is associated with the *available* bandwidth capacity $C(l_s(i,j))$.

Let ψ be a set of substrate paths in the substrate network G_s. The available bandwidth capacity $C(P)$ associated to a substrate path $P \in \psi$ between two substrate nodes can be evaluated as the minimal residual bandwidth of the links along the substrate path:

$$C(P) = \min_{l_s(i,j) \in P} C(l_s(i,j)) \tag{1}$$

Let V_s and M_s denote a node capacity vector and a link capacity matrix, respectively, associated to the graph G_s such that:

- V_s=$[C(n_s^i)]$ is the available capacity vector for substrate nodes n_v^i, where $1 \leq i \leq |N_s|$.
- M_s=$[C(l_s(i,j))]$ is the available bandwidth capacity matrix for substrate links $l_v \in L_s$ between nodes n_s^i and n_s^j, where $1 \leq i,j \leq |N_s|$.

Virtual Network Model: An InP receives requests to set up on-demand VN topologies with different capacity parameters over the shared substrate. Each virtual node $n_v \in N_v$ is associated with a minimum required capacity denoted by $C(n_v)$. Each virtual link $l_v \in L_v$ between two virtual nodes is associated with a capacity weight value $C(l_v)$ which denotes the minimum required bandwidth capacity of the virtual link l_v.

We represent a VN request with the quadruple Req $= (Reqid, G_v, V_v, M_v)$, where *Reqid* represents the unique identifier for the request *Req*. V_v and M_v denote a node capacity vector and a link capacity matrix, respectively, associated to the graph G_v, so that:

- V_v=$[C(n_v^i)]$ is the minimum required capacity vector for virtual nodes n_v^i, where $1 \leq i \leq |N_v|$.
- M_v=$[C(l_v(i,j))]$ is the minimum required bandwidth capacity matrix for virtual links $l_v \in L_v$ between nodes n_v^i and n_v^j, where $1 \leq i,j \leq |N_v|$.

VN Mapping Problem Formulation: Based on the substrate and VN models, the challenge is to find the best mapping between the virtual graph G_v and the substrate graph G_s given specific objectives. Our goal is to provide a mapping, denoted by *MAP*, that optimizes load balancing across the substrate resources with respect to the capacity constraints. Finding the optimal VN mapping solution that satisfies multiple objectives and constraints can be formulated as an NP-hard problem, as follows:

Node Mapping: Let $MAP_N : N_v \rightarrow N_s^{Reqid} \subseteq N_s$ denote a mapping function between virtual nodes and substrate nodes, where N_s^{Reqid} represents the set of substrate nodes capable of supporting at least one virtual node of a request *Reqid*, i.e., $N_s^{Reqid} = \{n_s \in N_s \mid C(n_s) \geq \min_{n_v \in N_v}\{C(n_v)\}\}$.

Link Mapping: Let $MAP_L : L_v \rightarrow \phi \subseteq \psi$ denote a mapping function between virtual links and substrate paths, where $\phi = \{P \in \psi \mid C(P) \geq C(l_v), \forall\ l_v \in L_v\}$.

3.2 Resource Assignment Algorithm

We propose a centralized and heuristic resource assignment algorithm (Algorithm 1) with simultaneous node and link mapping. Since our objective is to optimize load balancing over substrate nodes, we use a greedy node mapping algorithm to assign virtual nodes to the substrate nodes with the maximum substrate resources.

Once a virtual node n_v is assigned, all virtual links directly connected to this node as well as the set of its neighborhood nodes $Nei(n_v)$ are assigned. Virtual link assignment is based on the shortest-path (SPT) algorithm. To accomplish that, two predefined functions are used in this algorithm: *SORT* and *HEAD* function. *SORT* is a sorting function that sorts a vector of nodes (e.g., N_s) based on their capacities by ordering them from higher to lower capacity. The *HEAD* function returns the first element (node identifier) of the vector. The SPT algorithm computes a path from node n_s to each node k ($n_s \neq k$) so that the weight between node n_s and all other nodes is minimum. Let P_k denotes the shortest path between node n_s and a substrate node k. The substrate path P_k is associated with the minimum path capacity $C(P_k)$ between k and n_s (see Equation 1). Consequently, the SPT algorithm returns a set T_{n_s} associated to the node n_s such that: $T_{n_s} = \{(P_k, C(P_k))\forall\ k \in N_s^{Reqid}\}$.

For each virtual node $j \in Nei(N_v)$ (starting with the largest required capacity) the algorithm will select the substrate node k such that $C(P_k)$ is minimal. The node k should also satisfy virtual node and link constraints so that $C(k) > C(j)$ and $C(P_k) > M_v[n_v][j]$. Therefore, the virtual node j is assigned to the substrate node k and the virtual link $l_v(n_v, j)$ is assigned to the substrate path P_k at one

stage (i.e., simultaneous node and link mapping). The same process is repeated for the residual VN graph until all virtual nodes and links are assigned. Once the entire request is mapped successfully onto the substrate, the algorithm execution is terminated.

In order to satisfy the node and link constraints of incoming VN requests, updated resource information is needed. To this end, the substrate nodes monitor CPU load and link bandwidth which are subsequently communicated to the InP management node before embedding a requested VN.

Algorithm 1. Resource Assignment

Inputs: $G_s = (N_s^{Reqid}, L_s), V_s, M_s$

$G_v = (N_v, L_v), V_v, M_v$

SORT(V_v)

SORT(V_s)

$n_v \leftarrow$ HEAD(V_v)

$n_s \leftarrow$ HEAD(V_s)

MAP$_N(n_v) \leftarrow n_s$ // Virtual node with largest required capacity is mapped to the substrate node with maximum available capacity

SORT($Nei(n_v)$)

for each $l_v(n_v, j) \in L_v$; $j \in Nei(n_v)$ // Mapping all virtual links directly connected to the virtual node n_v and its adjacency list $Nei(n_v)$ **do**

a. $T_{n_s} \leftarrow$ SPT(n_s)

b. $k \leftarrow \{k \in N_s^{Reqid} \setminus n_s;\ C(P_k)$ isminimal$\}$ such that:

$C(P_k) > M_v[n_v][j]$ and $V_s[k] > V_v[j]$ // substrate node k should satisfy the requested node and link constraints

i. MAP$_N(j) \leftarrow k$

ii. MAP$_L(l_v(n_v, j)) \leftarrow P_k$

iii. $N_v \leftarrow N_v \setminus j$ and $L_v \leftarrow L_v \setminus l_v(n_v, j)$ // Removing the already assigned virtual nodes and links from the VN request graph

if $N_v = \emptyset$ **then**

STOP

else

GOTO 1 with the residual graph of the VN request G_v

end if

end for

4 Virtual Link Setup

The setup of virtual links across multiple domains with QoS guarantees requires a resource reservation protocol that supports inter-domain signaling. Recently, IETF approved the Next Steps in Signaling Protocol (NSIS) suite which provides a resource reservation protocol for inter-domain signaling (i.e., QoS NSLP [13]). One major enabler in this context is the generic QoS parameter specification

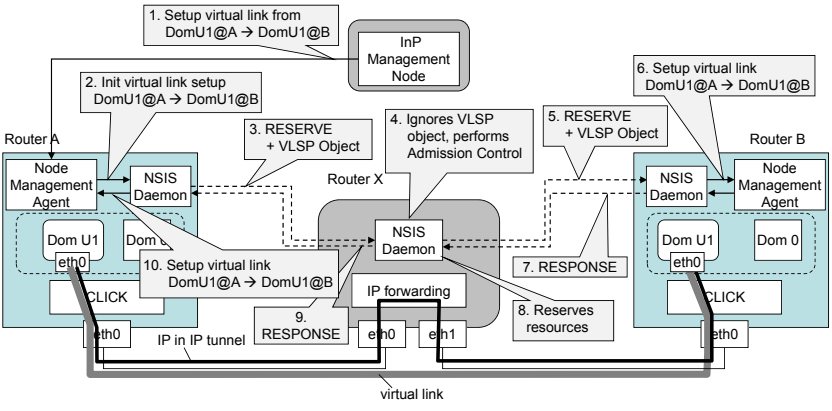


Fig. 1. Virtual link setup

using the QSPEC template [2], which can be mapped to domain specific QoS mechanisms. Irrespective of the herein proposed NSIS-based solution, however, InPs must agree on a common method and signaling protocol to set up virtual links between their domains a priori.

We use the NSIS protocol suite (the NSIS-ka implementation [14]) to combine the virtual link setup with the resource reservation signaling via the QoS NSLP. This integration reduces the setup time of a virtual link, as the resource reservation at the same time conveys the necessary address information of the virtual link. A QoS NSLP extension mechanism for carrying new objects is used to convey the newly created *virtual link setup protocol* (VLSP) object. Only the substrate nodes hosting the virtual nodes at the edges of the virtual link need to support the VLSP object and act on it accordingly by installing any state required for virtual link setup. Intermediate substrate nodes may be involved in guaranteeing QoS properties of the virtual link or an aggregate of virtual links and therefore need to process the QoS NSLP content. They can, however, simply ignore and forward the contained VLSP object, which is ensured by the NSLP object extensions flags in the VLSP object header. The path-coupled signaling approach of NSIS ensures that a viable substrate path with enough resources exists to accommodate the new virtual link. The VLSP object was also used in [4] to carry the required information for virtual link setup combined with authentication.

Fig. 1 shows the sequence of events during the setup of a unidirectional virtual link: Router A and Router X belong to InP1, whereas Router B resides in InP2's infrastructure. For simplicity, we consider Router B acting as gateway border router of InP2. Both routers A and B, which support virtualization, provide a control interface to their corresponding InP management node and run an NSIS daemon that interprets and processes the VLSP object. The intermediate Router X merely needs to perform general QoS tasks, such as admission control, resource reservation, and policing (as a border router), and hence it can use NSIS without any modifications.

The setup of a unidirectional virtual link between two virtual nodes (from DomU1@A to DomU1@B, both part of the same VN) is triggered by a request of the InP management node to the substrate node management agent (*Step 1*). This request contains the specification of the substrate link endpoints and the virtual link endpoints. A virtual link endpoint description at least includes identifiers for the VN, the virtual node, and the respective interface but can be easily extended. In *Step 2*, the request is passed via inter-process communication (IPC) to the NSIS daemon, which puts the QoS requirements (QSPEC object) and the virtual link description (VLSP object) into a QoS NSLP RESERVE message. In *Step 3*, a signaling connection with the NSIS instance on Router B is established and the QoS NLSP RESERVE message is sent. As mentioned before, the intermediate Router X only interprets the contained QSPEC object and performs admission control for the virtual link while the VLSP object is ignored (*Step 4*) and forwarded to Router B in the RESERVE message (*Step 5*). On arrival of the RESERVE message at Router B, admission control is performed and on success, resources are reserved and the local end of the virtual link is set up (Step 6). A RESPONSE message is sent back towards Router A (*Step 7*), which causes Router X to reserve the required resources for the virtual link (*Step 8*) and to forward the RESPONSE to Router A (*Step 9*). Router A then reserves local resources and installs the virtual link (*Step 10*), thereby establishing the data plane from DomU1@A to DomU1@B.

In Section 5.2, we subject our inter-domain solution to virtual link setup to a detailed performance analysis for the scenario given in Fig. 1.

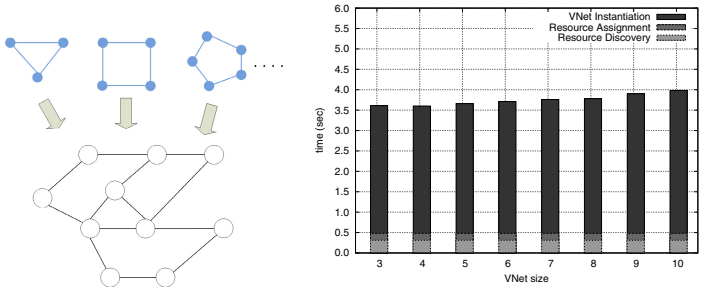
5 Evaluation

In this section, we use our prototype implementations to evaluate the performance with VN provisioning and provide insights into inter-domain virtual link setup.

5.1 Virtual Network Provisioning

VN provisioning experiments are carried out in the *Heterogeneous Experimental Network* (HEN) [9] using Dell PowerEdge 2950 systems with two Intel quad-core CPUs, 8 GB of DDR2 667MHz memory and 8 or 12 Gigabit ports. A fixed number of HEN nodes compose the substrate. To explore VN provisioning with multiple InPs, these nodes are split into multiple logical clusters, each one serving as an independent InP. Separate HEN nodes undertake the role of VNP and VNO.

First, we evaluate VN provisioning with a single domain. We measure the time required to provision VNs with varying size (3–10 nodes/links). Fig. 2(a) shows the substrate topology which is composed of 10 nodes. In Fig. 2(b), we demonstrate the time required for each provisioning step, including resource discovery, assignment, and VN instantiation. Our experimental results indicate that a VN can be provisioned just in a few seconds, with most time being spent within



(a) Experimental scenario (b) Time required for VN provisioning

Fig. 2. VN provisioning with single InP

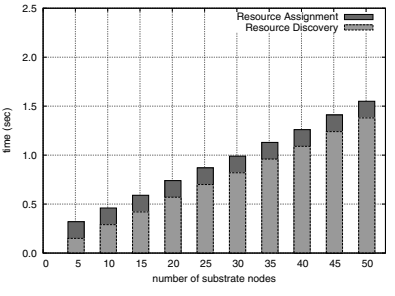
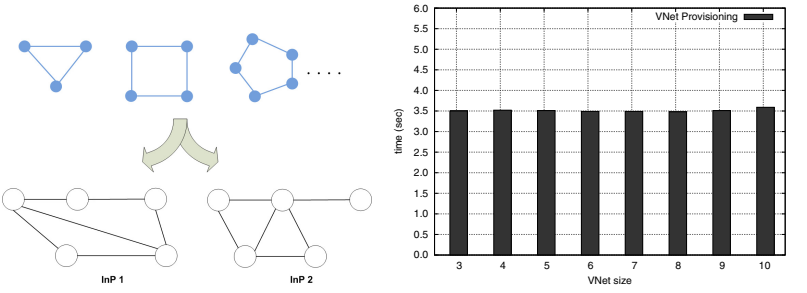


Fig. 3. Resource discovery and assignment with a diverse number of substrate nodes

the InP for virtual node and link setup. More precisely, it takes 3.61 seconds on average across 20 runs with a small standard deviation to provision the VN composed of 3 nodes/links. Resource discovery and assignment are concluded in 0.31 and 0.17 seconds, respectively.

Fig. 2(b) also shows that VN provisioning scales within our experimental infrastructure. Varying the size of the requested VN has no noticeable impact on resource discovery and assignment. According to Fig. 2(b), instantiation times increase only slightly for larger VNs, due to the parallelism during virtual node and link setup. Tests with no parallelization during VN instantiation show significantly higher delays, even for VNs with small size.

In Fig. 3, we provide more insights into resource discovery and assignment. In particular, we measure the delay incurred for these steps while provisioning a VN with 5 nodes/links on top of a substrate network with a varying number of nodes (5–50). Fig. 3 shows that the time required to embed the VN is not increased, validating the efficiency of the proposed resource assignment algorithm in terms of execution time. We further assess the performance of our algorithm with larger substrate nodes and requested VNs. To this end, we implemented a tool for constructing network topologies with CPU and bandwidth specifications. Our tests show that the delay incurred during resource assignment is less than 1 sec for substrates with a number of nodes as high as 200.



(a) Experimental scenario (b) Time required for VN provisioning

Fig. 4. VN provisioning with two InPs

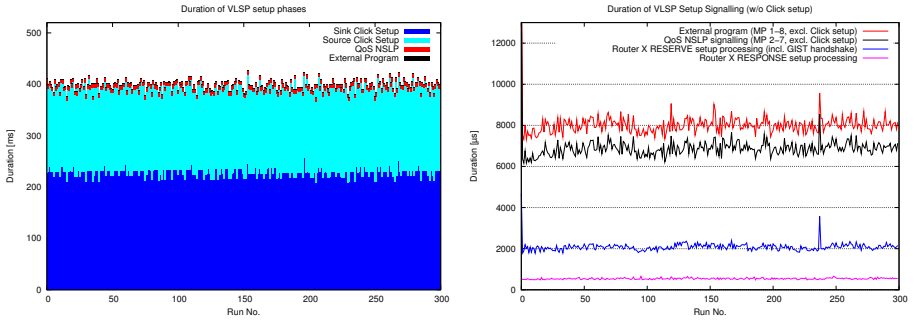
As depicted in Fig. 3, resource discovery scales linearly with the number of substrate nodes, as each substrate node communicates updated resource information to the InP management node, upon an incoming VN request. For large infrastructures, delegating configuration management across multiple nodes can provide more efficiency and lower delays during VN provisioning.

We also assess the performance with VN provisioning and two InPs. To this end, we use the experimental scenario of Fig. 4(a) where VNs with varying size (3–10 nodes/links) are embedded onto two InPs, each one composed of 5 substrate nodes. Fig. 4(b) validates the high performance and the nice scalability properties within our experimental infrastructure. Regardless of its size, each VN can be provisioned within a few seconds. This shows that efficient VN splitting and high parallelization (within each InP and across all participating InPs) during VN instantiation are essential for fast and scalable provisioning of VNs. Similar performance numbers are obtained when we use our prototype implementation to assess VN provisioning in another medium-scale experimental infrastructure (composed of 60 nodes with a single quad-core CPU at 2.26 GHz and 6 GB of DDR3 memory).

5.2 Virtual Link Setup

We conduct a performance analysis of virtual link setup in a controllable environment, based on the setup of Fig. 1. Each node consists of a Pentium IV PC running at 2.8 GHz, 2 GB RAM, and 4 Gigabit Ethernet network interfaces (Intel(R) PRO/1000), interconnected by a Cisco Catalyst Switch 6500 running CatOS. Nodes A and B use a Debian Linux (Squeeze) running kernel version 2.6.18.8, patched so that Click can run with Xen. Node X uses the same Linux distribution but with the Debian distribution kernel 2.6.32-3-686.

The signaling for virtual link setup follows the steps depicted in Fig. 1 and the signaling procedure is repeated 300 times (using packet dumps and in-code timestamps). VLSP objects are processed on the endpoints of the virtual link and the NSIS daemon calls a setup script for Click, passing to it the arguments extracted from the VLSP object in order to establish the virtual link.



(a) with Click setup (b) without Click setup

Fig. 5. Time spent during virtual link setup

Fig. 5(a) shows the results of the VLSP setup for each of the 300 measurements. The measured time for virtual link setup includes the time required for inter-process communication, signaling via NSIS (including admission control and resource reservation), and the execution of ruby scripts on Router A and Router B, which determine and install the required Click scripts for the virtual link. On average, this takes $399ms$ overall, a large part of which is consumed for the execution of the two ruby scripts including Click script installation. At the sink, an average of $228ms$ is spent to determine and install the Click script while the corresponding delay for the source is $162ms$ on average. With a standard deviation of only $9ms$, the setup of virtual links behaves very deterministically.

Fig. 5(b) shows the duration of the overall signaling exchange *excluding* execution time of the Click setup scripts. The two lines at the top show the virtual link setup time as measured at Router A with and without IPC. The QoS NSLP is triggered by an external program running in the node management agent, and ends with returning a result of the request to the external program. Fig. 5(b) shows the overall duration for the signalling message exchange without Click scripts. This sums up to 8 ms on average in our setup with one intermediate substrate node and to 7 ms when omitting the IPC between the node management agent and the NSIS daemon. The two curves at the bottom show the processing time of the RESERVE and RESPONSE respectively in the intermediate router X during setup. The RESPONSE processing takes only $541\,\mu s$ on average, whereas the RESERVE processing requires more time since it includes each time a GIST three-way handshake phase. It becomes already apparent in this simple setup that the combined QoS/VLSP signaling is responsible only for a minor part of the overall time spent in the setup of the virtual link. Even if the substrate path across two domains comprises several hops, we expect that the resource reservation processing will not dominate the virtual link setup time.

6 Conclusions

In this paper, we discussed VN provisioning with emphasis on techniques and embedding algorithms that comply with the restrictions imposed by multiple

domains, such as limited information disclosure. We also provided the required interoperability for inter-domain virtual link setup with QoS guarantees. Despite the increased complexity for resource discovery and allocation, we showed that small VNs can be provisioned within a few seconds. Our results indicate that new business models can create commercial products that setup VNs in large infrastructures in the order of minutes. The proposed VN provisioning framework essentially lowers the barrier for large-scale service deployment by offering the capability to lease network slices from multiple substrate providers.

Due to the limitations of our experimental infrastructure, we were not able to investigate issues of large scale. In the future, we plan to implement a simulator optimized for large scale and thereby, examine scalability issues with VN embedding onto multiple large infrastructures.

References

1. 4WARD Project, http://www.4ward-project.eu
2. Ash, G., Bader, A., Kappler, C., Oran, D.: QSPEC Template for the Quality-of-Service NSIS Signaling Layer Protocol (NSLP), RFC 5975 (October 2010)
3. Barham, P., Dragovic, B., Fraser, K., Hand, S., Harris, T., Ho, A., Neugebauer, R., Pratt, I., Warfield, A.: Xen and the Art of Virtualization. In: Proc. 19th ACM Symposium on OS Principles, Bolton Landing, NY, USA (October 2003)
4. Bless, R., Röhricht, M., Werle, C.: Authenticated Setup of Virtual Links with Quality-of-Service Guarantees. In: Proc. IEEE ICCCN 2011, Hawaii, USA (July 2011)
5. Chowdhury, M., Rahman, M., Boutaba, R.: Virtual Network Embedding with Coordinated Node and Link Mapping. In: Proc. IEEE Infocom 2009, Rio de Janeiro, Brazil (April 2009)
6. Chowdhury, M., Samuel, F., Boutaba, R.: PolyViNE: Policy-based Virtual Network Embedding Across Multiple Domains. In: Proc. ACM SIGCOMM VISA, New Delhi, India (September 2010)
7. GENI: Global Environment for Network Innovations, http://www.geni.net
8. Hancock, R., Karagiannis, G., Loughney, J., Van den Bosch, S.: Next Steps in Signaling (NSIS) Framework, RFC 4080 (June 2005)
9. Heterogeneous Experimental Network, http://hen.cs.ucl.ac.uk
10. Houidi, I., Louati, W., Bean-Ameur, W., Zeghlache, D.: Virtual Network Provisioning Across Multiple Substrate Networks. Computer Networks 55(4) (March 2011)
11. Kohler, E., Morris, R., Chen, B., Jahnotti, J., Kasshoek, M.F.: The Click Modular Router. ACM Transaction on Computer Systems 18(3) (2000)
12. Lu, J., Turner, J.: Efficient Mapping of Virtual Networks onto a Shared Substrate. Washington University. Technical Report WUCSE-2006-35 (2006)
13. Manner, J., Karagiannis, G., McDonald, A.: NSIS Signaling Layer Protocol (NSLP) for Quality-of-Service Signaling, RFC 5974 (October 2010)
14. NSIS-ka, A free C++ implementation of NSIS protocols, KIT, https://svn.tm.kit.edu/trac/NSIS
15. Schaffrath, G., Werle, C., Papadimitriou, P., Feldmann, A., Bless, R., Greenhalgh, A., Wundsam, A., Kind, M., Maennel, O., Mathy, L.: Network Virtualization Architecture: Proposal and Initial Prototype. In: Proc. ACM SIGCOMM VISA, Barcelona, Spain (August 2009)

16. Yu, M., Yi, Y., Rexford, J., Chiang, M.: Rethinking Virtual Network Embedding: Substrate Support for Path Splitting and Migration. ACM SIGCOMM Computer Communications Review 38(2), 17–29 (2008)
17. Zhu, Y., Ammar, M.: Algorithms for Assigning Substrate Network Resources to Virtual Network Components. In: Proc. IEEE Infocom, Barcelona, Spain (April 2006)
18. Zu, Y., Zhang-Shen, R., Rangarajan, S., Rexford, J.: Cabernet: Connectivity Architecture for Better Network Services. In: Proc. ACM ReArch 2008, Madrid, Spain (December 2008)

Prometheus: A Wirelessly Interconnected, Pico-Datacenter Framework for the Developing World

Vasileios Lakafosis[1], Sreenivas Addagatla[2], Christian Belady[2], and Suyash Sinha[2]

[1] School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA
Vasileios@gatech.edu
[2] Microsoft Research, Redmond, WA 98052, USA
{Sreenivas.Addagatla,Christian.Belady,Suyash.Sinha}@microsoft.com

Abstract. The promise of cloud computing is, nowadays, mostly limited to the developed regions of the world where, approximately, only one half of the world's population lives. In this paper, we present an attempt to bring the cloud to the majority populations of the developing world, with the help of long-distance, wirelessly connected and renewable-energy-powered pico-datacenters, the Prometheus nodes. Along with the physical layer and ad-hoc network routing characteristics of the prototype nodes, the challenges and potential solutions in designing such a network with constraints on renewable energy availability, bandwidth and connectivity to the Internet are discussed. With this multifaceted theoretical and experimental analysis, we believe that not only does the pico-DC framework constitute a highly viable solution for the developing world to share computational resources and storage services over wireless links but also that Prometheus can significantly help to improve multiple socioeconomic aspects of the populations of the developing world.

Keywords: Pico-datacenter, pico-DC, Wireless Mesh Networks, Ad Hoc Communication, Rural Connectivity, low-power; Internet.

1 Introduction

The promise of cloud computing is largely limited to the developed world where, according to a United Nations report, only one-half of the world's population lives [1]. In fact, based on the same study, the urban population proportion is merely expected to exceed 60% by 2035 and, thus, a major part of the world will still be living sparsely in rural areas of the developing countries with a very low per capita income. Currently, out of this rural part of the population, one billion people live in shanty towns [2].

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 26–39, 2012.

Table 1. The Indian Telecom Penetration (July - Sept. 2010)

		Percentage of Population	Absolute number of Subscribers
Wireline Subscribers	Urban	2.2%	26.44 Million
	Rural	0.8%	9.13 Million
Wireless Subscribers	Urban	38.1%	460.63 Million
	Rural	18.8%	227.08 Million
	GSM	47.8%	578.49 Million
	CDMA	9.0%	109.22 Million
Internet & Broadband Subscribers	Total	1.5%	17.90 Million
	Broadband	0.9%	10.31 Million

Given this situation, it comes as no surprise that telecom companies are creating networks only in densely populated areas of the developing countries and these networks are still mainly voice-centric, with little or no interest on data services, due to the lack of promising returns on investments. An example that illustrates well the digital divide problem is given in Table 1; this includes data about the telecom penetration in India as reported by the Telecom Regulatory Authority of India in their report entitled *"The Indian Telecom Services Performance Indicators (July - September* 2010*)"* issued in January 2011 [3]. It is worth to first note that only 1.5% of the Indian population have access to the Internet with 40% of these connections not being broadband. Moreover, less than one fifth of the Indian population that correspond to rural citizens are wireless subscribers, due to the sparse cellular coverage, and the vast majority of all Indian mobile phone holders rely on old, low-end GSM mobile phone with practically no data connections. Even not the majority of the 9% of the CDMA users, who live at the dense urban areas, are expected to have enabled data plans on their connections. Needless to mention that less than 1% of the Indian population that correspond to rural citizens are wireline subscribers.

The past, however, has shown that the rural and suburban areas are the ones where socioeconomic changes, brought by cloud computing, can be the most dramatic. Moreover, as discussed in [4], the poor not only need digital services, but they are willing and able to pay for them to offset the much higher costs of unfair pricing, corruption and poor transportation.

This paper presents preliminary design aspects and implementation results of Prometheus, the first pico-DC framework that promises to bring the cloud down to the majority populations of the developing world; i.e. lower barriers to enter in a space where computing and storage are delivered as a service rather than a product in a dynamically scalable and virtualized fashion. Prometheus consists of long-distance wireless-based, renewable-energy-powered and extremely small datacenter (*pico-DC*) nodes with enough compute and storage capability to handle the throughput and computational needs of a medium-sized village of a few hundreds of people. As opposed to the containerized modular micro-DCs

[5] that can house a thousand servers and draw around $500KW$, our proposed framework consists of "green" pico datacenter nodes that are small and light enough to be mounted on a pole or placed on a rooftop and consume a peak power of $150W$; allowing them to operate unobtrusively for a few days under unexpected harsh weather and light conditions.

2 Cloud Services for Rural Areas

The usefulness of a local cloud infrastructure, which can operate autonomously, both power- and network-wise without necessarily being continuously connected to the Internet but only intermittently, and which is mainly accessed through "thin" kiosk clients, is better reflected through example scenarios, as the ones identified below. These example scenarios attempt to illustrate the importance of timely and zero operation-cost computing for decision making and commercial interaction of developing parts of the world relying on their own local computing infrastructure rather than unreliable, generally low-speed, Internet connections or the cloud of their developed world counterparts a lot before even access becomes available to the latter information resources or the central government.

- Real-time, thermal or otherwise, images collected directly by the spatially separated pico-DC nodes or nearby local users contain information relevant to emergency weather conditions or other public safety provisions. While each individual image data is incomplete from a large-scale perspective, extended provincial situational awareness and decision making can be enabled if all these images are collectively processed by multiple pico-DCs.
- An illiterate villager wants to send a voice or text message relevant to a personal or business matter to another person in India. Just by capturing his voice in his own local language dialect with a common use telephone or personal networked device, a sophisticated[1] *natural language processing* cloud application will translate, if necessary, the message to the intended recipient's dialect or Hindi and send it.
- In most rural places where neither doctors nor tele-medicine applications are instantly available, ever popular and computationally heavy medical software can analyze a diverse set of symptoms and health condition parameters and provide a, potentially saving, first diagnosis. In addition, timely information pertinent to a looming epidemic may be gained and help save hundreds of lives in rural areas.
- A villager or local entrepreneur, in the fields, for example, of agriculture and livestock, buys or offers goods for sale at the most beneficial price through the pico-DC-enabled local version of "Craigslist" and "eBay". Providing means to having access to a wider array of merchandisers or customers can enable growth opportunities in various local market segments.

[1] Such as those offered by Nuance (www.nuance.com) that typically have a size of 1GB and cannot run independently on commodity machines or smart phones.

3 Related Work

The common characteristic of all the efforts to deploy wireless networking topologies in rural parts of the world, including India [6,7] and Africa [8], has been that they do not span more than a few hops away from cities that provide Internet gateways. Only if a rural user is located within only a few wireless hops away from a limited-capacity satellite link or a wired Internet gateway, is it possible for him to have access to the Internet. For these highly-partitioned networks, message ferries [9], such as buses and boats, have been proposed to provide intermittent connectivity. This paradigm, named Delay Tolerant Networking, has seen interesting non-real time, i.e. asynchronous, communication applications [9,10,11,12,13]. Nevertheless, none of the above research efforts has suggested providing, involves any kind of or possesses the necessary hardware infrastructure to provide computational services to the low per capita populations.

An ever-increasing number of desktop or mobile applications nowadays relies on the processing power and storage capabilities offered by datacenters to provide a better experience to their users. In the recent past, there have been only a few proposals to harness the power of *grid computing* from mobile devices [14,15,16]. All of these efforts only involve the use of the mobile smart phones as interfaces to remote grid computing infrastructures that reside remotely in large datacenters, where the computational jobs are actually outsourced. That means that no computation takes place locally, which can be prohibitive for a large set of types of applications that require transferring data that need to be processed to a remote and powerful computational site when the aggregate bandwidth of even a few wireless hops away from an Internet gateway cannot exceed very few $Mbps$.

All the above related work does not cover the concept of a *local cloud* that we envision and present in this paper: a computing service that is totally run locally by the Prometheus pico-DC nodes and can fully operate autonomously in an off-grid fashion. To the best of our knowledge, Prometheus is the first of this kind of proposals, targeted to bring computing services from the "cloud" down to the developing and under-developed regions of the world.

4 Prometheus Architecture Overview

4.1 Design Aspects

The Prometheus architecture is drawn on the following constraints:

- A pico-DC instance's peak power budget is limited to $150W$, which, as shown in Section 5.2, is enough to simultaneously cover the real-time operational power needs and continuously charge the batteries for unobtrusive operation even under unexpected harsh weather and light conditions. Moreover, the nodes operate totally off the electricity grid, exploiting the renewable solar and/or wind power. This is important not only for achieving the zero-operation cost target but also does not require from local communities to make special budget provisioning of their limited power available.

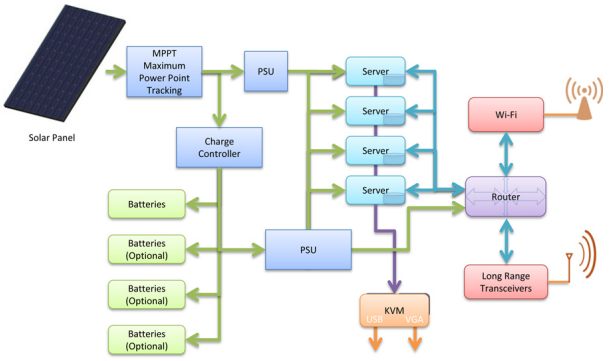
- The pico-DC instances are able to operate in open weather, subject to wind and variances of temperature and humidity across various regions of the world.
- The pico-DC instances can connect to other peer nodes over distances extending up to $20km$. Partially-overlapping wireless areas with a radius of up to $20km$ are exceptionally suitable for not only collectively covering villages of a few square kilometers with only a few pico-DCs nodes and given the 802.11 wireless propagation restrictions but also for inter-connecting such clusters (see Fig. 2).
- A network of such pico-DC nodes may include a few gateways (sink points) to deliver content from the Internet.
- The computing resources and, optionally content from the Internet, offered by the pico-DC instance can be accessed by the community via local Wi-Fi connections. In particular, access to the Prometheus services is possible through common-use, fixed, "thin" kiosks placed at central points of the villages or personal, wireless mobile terminals, if available.
- A pico-DC instance is built and deployed within an overall cost of $1,700$ U.S. dollars (including all the components and our prototype enclosure presented in the next subsection). Note that, in addition to any potential maintenance needed during its lifetime, this is the only one-time cost fee involved, since there are zero operational costs for powering the pico-DCs and the *Software-as-a-Service* is run and offered locally rather than "rented" from a third Internet party. This estimated cost of purchase does not account for the economies of scale that will push it down significantly and make it even more affordable by low-income communities of a few hundred people [4].

4.2 Components of a Prometheus Node

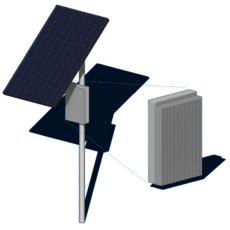
Figure 1a shows a block diagram of the components of a pico-DC node that address the aforementioned design objectives. It should be highlighted that the design of all modules, namely power, computing and communication circuitry modules, allows them to be housed by the same board with an aim to increase performance, decrease power leakage and miniaturize the size, so that they can fit inside our novel prototype enclosure, which assists heat dissipation.

Prometheus Power Module. The $1m \times 1.5m$ solar cell array consisting of two $120W$ solar panels is the main power source to the pico-DC instance, charging a set of four $12V$ $6.8Ah$ Lithium-Ion batteries. These batteries are meant to be available as a secondary source when solar power is not available.

A *Maximum Power Point Tracking* (*MPPT*) device monitors the solar panel's I - V (current - voltage) curve and provides the maximum available variable power output to both a *Power Supply Unit* (*PSU*), which is directly interfaced to the servers and the wireless interfaces, and the battery array. A charge controller monitors and controls the battery charge level to prevent from over-charging or deep discharging of the battery array. A secondary PSU that can draw power from the battery unit is activated only when the solar power is unavailable.



(a) Block Diagram of the pico-datacenter (pico-DC) node.



(b) The Prometheus pico-DC node (inset: the *compute module*).

Fig. 1.

Prometheus Computing Module. The *computing module* contains a set of four inexpensive low-power dual-core processors, the operation of which is dependent on the load request, as well as the power availability. Each processor has a dedicated memory module of $2GB$.

All the servers within a node use the same storage device mounted as a file share, and this device is configured as a network-attached storage. For this, we considered both the options of regular high-capacity hard disk drives, and solid-state disks that, despite their still higher price, are more reliable and are less energy consuming.

On top of this hardware layer, a hypervisor-based virtualization system, such as Microsoft Hyper-V (see `www.microsoft.com/hyper-v-server/`), is designed to run that provides processor consolidation and can support a wide variety of guest operating systems depending on the type and number of end applications. These applications are to be provided both under the *client-server* model, when one Prometheus node acts as a "server", and, more importantly, under the *grid-computing* model, whereby the existing topology of networked Prometheus nodes acts in concert to carry out large tasks.

Prometheus Networking Module The *wireless routing module* provides long-distance (point-to-point, directional) IEEE 802.11 Wi-Fi connectivity to other Prometheus nodes, and local (broadcast) Wi-Fi connectivity to community client devices (Wi-Fi enabled "thin" kisosks or other mobile devices). In particular, the PCEngines Alix 3d3 board (see `http://pcengines.ch/`) houses three different types of wireless high-transmit-power transceivers (Ubiquiti (see `http://ubnt.com/`) XR2, XR5, and R52hn) that connect via mini-PCI interfaces to highly-directional reflective grid antennas, which exhibit a minimum air resistance. Dual-band (2.4 and $5GHz$) omni-directional antennas are used for the long- and short- distance links. The *communication module* possesses its own compact flash card from which it runs Windows Embedded OS Standard

7 with a disk footprint below $600MB$ and the average memory requirements remaining below $170MB$. The performance analysis experimentation, described in Section 5, involved maximum range with high-data-traffic reliability tests, as well as stress performance testing of the Alix board for peak traffic and CPU load and have demonstrated the robustness of our solution.

The *router module's* processor, which is separate from the *computing module's* set of processors destined for regular DC operation tasks, is also intended to run an intelligent power control system that is based on a solar-power-prediction algorithm. This power shaping is an invaluable tool for peak provisioning and smoothing power demand.

From a mechanical design aspect, the low-power processor configuration allows their cooling to be completely fan-less, which enhances the node's lifetime and reliability. Specifically, the passive cooling of the circuit boards and processors is achieved by allowing air to flow either through the large-area heat sinks of our prototype water- and air-sealed enclosure, shown in the inset of Fig. 1b, or through the waterproof air vents (bottom to top) of a second enclosure placed tangentially right under the solar panel.

The design of the Prometheus node provides scalability, primarily, in terms of a multiple server configuration that can adjust to the offered demand and available energy reservoir and, secondarily, in terms of multiple batteries, the number of which depends on the expected annual average solar illumination of the location where the node is installed. The dimensions and the weight of the weather-proof node that houses all the electronic parts, along with a solar panel, are small enough so that it is possible to mount the node on metallic poles, or on existing antenna towers or place it on rooftops.

4.3 The Prometheus Network

A Prometheus wireless network is drawn in Fig. 2. This topology results from the combination of long-distance (10 - 20 kilometers), point-to-point, high-bandwidth (5 - 30$Mbps$) backhaul links, connecting villages and towns, and medium-range broadcast or point-to-multipoint access links to individual users and other public organizations or small businesses. Conceptually, the same network consists of local cloud clusters of a few hundreds of users, the head of which is a pico-DC node that also serves as a wireless mesh node of the Prometheus network. The cloud clusters are shown on the same topology of Fig. 2 and the pico-DC nodes are mounted on each of the towers of the clusters. In such networks there may or may not exist Internet gateways, as the ones shown in the upper left corner (town's satellite link) and the right side (large city's optical fiber) of Fig. 2. As demonstrated with real, on-field link setups presented in Section 5.1, successfully setting up such Prometheus networks is totally feasible.

Network Routing Connectivity As opposed to typical *Mobile Ad Hoc Networks* (MANETs), the dynamic nature in the routing of our wireless topology does not stem from the mobility of the nodes, since the pico-DC nodes are static. Instead, the routing dynamics are derived from the intermittent network

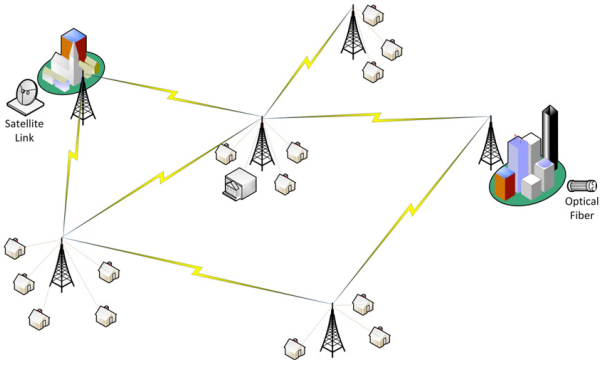


Fig. 2. The Prometheus wireless network topology

presence of the Prometheus nodes due to the possible power outages (see Section 5.2) or even wireless propagation issues (see Section 5.1). The resulting routing overhead, i.e. exchange of routing update maintenance packets and/or route discovery latency, necessitate the deployment of a multi-hop and multi-path wireless mesh routing protocol. We have chosen the *Optimized Link State Routing protocol with link quality sensing* (*OLSRd*) [17] to successfully establish data communication between our prototype pico-DC *networking modules*. The reason for opting for a proactive protocol, rather than reactive (or on-demand) ones such as *AODV* [18], *DSR* [19] and *LQSR* [20], has been our observation that the increased data overhead for maintenance and the potential slow reaction on relatively unlikely restructuring and failures is less prominent than the "reactive" effects of high latency in route finding and excessive flooding.

5 On-Field Performance Analysis

5.1 Wireless Propagation Analysis

The results on the expected maximum distance achieved for certain fade margins, different terrain scenarios, different frequency bands and different antenna gains based on the Longley-Rice Irregular Terrain Model [21] are compiled in Table 2. The different terrain scenarios involve different heights above the mean sea level of the transmitter and receiver, namely hill-to-valley ($400m + 5m$), hill-to-hill ($400m+400m$) and flat terrain ($5m+5m$). As expected, the lower the frequencies used the lower the free space attenuation and, thus, the longer the range that is achieved despite the stricter requirements for a larger ellipsoidal Fresnel zone to be kept clear. Additionally, the higher the radios are placed above the ground, the larger the line-of-sight distance and, as a result, the effective communication distance. The fact that the achieved distance does not exceed $19km$, in case both antennas are elevated $5m$ above the ground, comes as no surprise given the earth curvature effect.

After having theoretically verified the feasibility of two long-distance point-to-point wireless links, we went ahead to experiment with two real link setups in

Table 2. Expected Maximum Distance (Fade Margin)

	Hill-to-Valley (400 m + 5 m)			*Hill-to-Hill (400 m + 400 m)*			*Flat terrain (5 m + 5 m)*		
Antenna Gain (dBi)	*915 MHz*	*2.42 GHz*	*5.8 GHz*	*915 MHz*	*2.42 GHz*	*5.8 GHz*	*915 MHz*	*2.42 GHz*	*5.8 GHz*
30	76 km (8 dB)	74 km (8 dB)	72 km (8 dB)	156 km (8 dB)	156 km (8 dB)	127 km (7.2 dB)	19 km (8 dB)	15.5 km (8 dB)	13.3 km (8 dB)
27	72.5 km (8 dB)	70 km (8 dB)	68 km (8 dB)	153 km (8 dB)	153 km (8 dB)	82 km (7.7 dB)	19 km (4.9 dB)	15.5 km (5.1 dB)	13.3 km (5 dB)
20	57.5 km (8 dB)	58 km (8 dB)	39 km (7.5 dB)	148 km (8 dB)	96 km (7.3 dB)	36 km (7.8 dB)	13 km (5 dB)	12 km (5 dB)	9.6 km (5 dB)
15	49 km (8 dB)	47.5 km (8 dB)	27.3 km (5.3 dB)	138 km (8 dB)	62 km (5.8 dB)	20 km (6.3 dB)	9.7 km (5.7 dB)	8.5 km (5.7 dB)	7.2 km (5.7 dB)
8	38 km (8 dB)	21 km (7 dB)	10.5 km (6.2 dB)	52 km (7.5 dB)	22 km (7.5 dB)	7.5 km (7 dB)	6.1 km (5.7 dB)	4.8 km (5.7 dB)	4.5 km (5.7 dB)

the eastern part of Ellensburg valley in Washington State, U.S.A. . The southern point, denoted as *Viewpoint 1* was located at a viewpoint on the southern side of interstate highway 82, shown in Fig. 3, and the two other points of the links were situated inside the *Wild Horse Puget Sound Energy Renewable Center*, as shown in Fig. 3. Regarding the wireless equipment, we relied on $27dBi$ antennas, each having a $5GHz$ 802.11a/n Airgrid M5 radio incorporated in its feed horn. Both links exceeded not only $21.5km$ of effective range but also the set throughput goals, finally achieving $5Mbps$ of sustained TCP traffic and $30Mbps$ of sustained UDP traffic using the TTCP benchmark tool, which measures TCP throughput between two endpoints. It is interesting to note that the received signal strength for the first point was not very good (around $-86dBm$), whereas for the second point $300m$ away it was significantly higher (around $-73dBm$) exceeding all wireless cards' sensitivity values required to achieve the maximum data rate supported. The main reason for the mediocre quality of the signal at the first trial was that the terrain at the first location, where the antenna was placed, was not as steep as the terrain at the second location inside the renewable energy center, thus partially obstructing the first Fresnel zone. It is also noteworthy that for both highly-directional links the fluctuations of the signal over the time did not exceed $\pm 3dBm$, which is indicative of the good work done with the pointing toward each other of the two antennas.

5.2 Power Budget Analysis

The fact that the operation of the pico-DC nodes relies on renewable energy sources, such as solar and wind energy, renders the energy characterization of the suggested prototypes imperative; especially the consumption profiling of the *communication module* that is the most power-hungry component of the pico-DC node.

Toward that end, we have measured the real-time power consumption of the *wireless router module* with the wireless interface cards attached to it for different wireless traffic and CPU load states; ranging from the idle state to relaying traffic from one wireless interface to the other and adding traffic and computational load by accessing this router and server board with the Microsoft Remote Desktop application through its Ethernet interface. The results are summarized

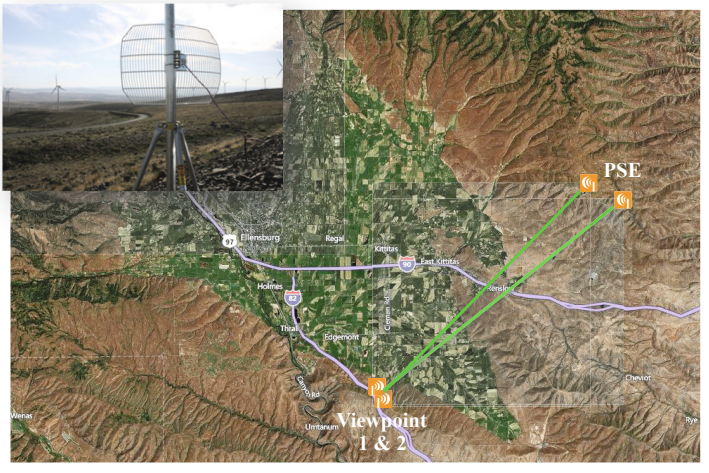


Fig. 3. The two long distance links of the wireless experimentation (extracted from Bing Maps). A photo of the PSE side is shown in the upper left corner.

in Table 3. Ultimately, the peak power consumption never exceeds $8W$, even for the highest CPU and traffic load. Although the idle power consumption of this wireless router prototype is a fraction (3.5%) of the node's overall peak power consumption ($150W$), we do wish to even eliminate this when no communication with the outer world is needed.

Table 3.

Alix 3D3 board	Ethernet	802.11a/n (R52hn) @ max Tx power	802.11b/g (XR2) @ max Tx power	Power Consumption
	Remote Desktop traffic + CPU @ 350MHz	14 Mbps TCP traffic (64K buffer)	14 Mbps TCP traffic (64K buffer)	
√				5.12 W
√	√			5.47 W
√		√		6.27 W
√		√	√	6.98 W
√	√	√	√	7.15 W

Totally eliminating the router's idle power can only be achieved by making the wireless board entirely shut down as if the power switch was to the OFF state and having the board powered on again only whenever the pico-DC node has to transmit to or receive data from neighboring point-to-point nodes or local wireless clients. The decision on waking up the router board can easily be made locally by having its processor, which works as the "supervisor" of the pico-DC

node, trigger the power switch of the board. However, in the case of receiving data this task is more challenging. The idea of complementing an energy-wise costly device with a very low-power micro-controller unit or transceiver, as those used in wireless sensor and personal networks, is not new [22,23]. Within the same notion, we propose the implementation of a *Wake-on-WLAN* technique, shown in Fig. 4, according to which a $2.4GHz$ wireless sensor mote, such as the very widely used Crossbow MicaZ with an overall power consumption of less than $20mW$, is connected to the same directional antenna as the corresponding wireless card interface through a single-pole-double-throw RF switch. The mote's incorporated TI transceiver's highest sensitivity ($-89dBm$) is comparable to that of a typical 802.11 interface and, as a result, this transceiver can be used to sense the same wireless link for "Wake-Up Req" probe packets emitted by its point-to-point neighbor that wishes to establish a connection. In this latter case, the mote makes use of one of its own digital output pins to trigger the router's power switch.

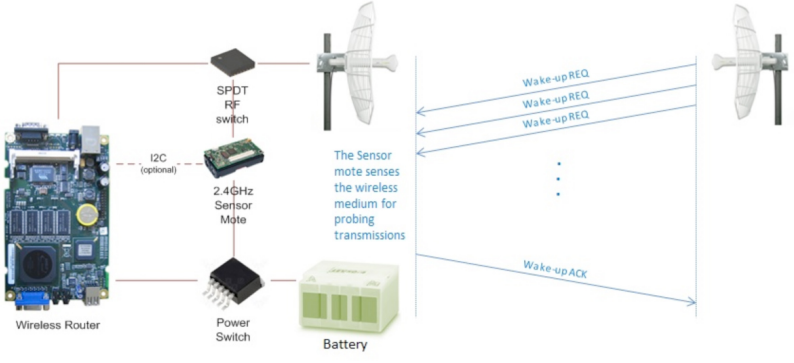


Fig. 4. A *Wake-on-WLAN* implementation

It is important to note that the dependence of the operation of the pico-DC on the intermittent solar power means that power outages throughout a single day are possible. In fact, this exact effect is shown in Fig. 5, which includes four different solar panel power generation curves during 6 am and 8 pm of four different summer 2010 days in Tukwila, Washington. Although one would expect that the solar illumination is at high levels throughout the whole late July or early August days even in Washington State, Fig. 5 shows that this is not the case. With the exception of only the *07/31/10* curve, for which the whole day was cloudy, the remaining curves actually witness severe sporadic and random drops of the solar illumination throughout the day; reflected by the steep drops of the power generated even below $50W$. This irregularity is the result of either sporadic local clouds or leafs covering just some of the solar cells but rending that corresponding whole line of cells useless.

The above results pronounce the need for smart energy-saving mechanisms incorporated into the node's software architecture that occupy a fraction of the

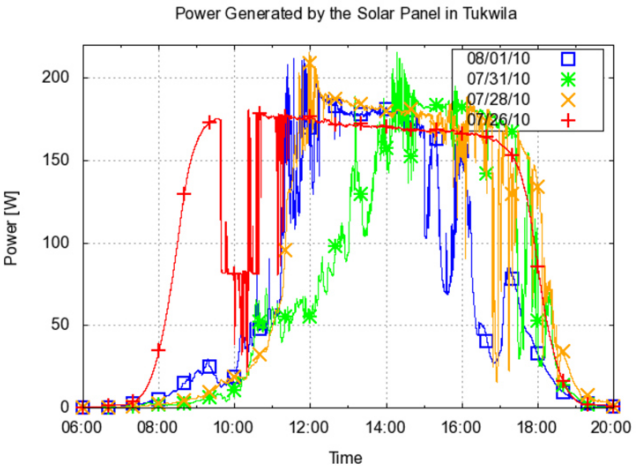


Fig. 5. Graph showing outages during daily operation

computing resources of the *router module's* processor. The proactive part of these mechanisms involves running coarse-grained power availability prediction algorithms based on weather forecast information. On the reactive side, the node is capable of adapting its real-time power consumption based on the contemporary environmental conditions, energy available in its charge tanks, i.e. the batteries, and the expected computational workload assigned or soon to be assigned. Specifically, the pico-DC node can perform a compute load movement across the Prometheus network, decrease excessive communication with peer Prometheus nodes, cease all links to the public Internet (in case of gateway nodes) or just take a full snapshot of its state, store it locally and transit to a deep sleep mode, i.e. check-point, consuming less than $5W$.

6 Conclusion

In this paper, we introduce Prometheus; a framework of long-distance wireless-based, renewable-energy-powered and extremely small datacenter (pico-DC) nodes. We have presented the high-level architecture of the Prometheus network, as well as the major design aspects and implementation of the pico-DC nodes. Novel ways for the masses to have access to the offered computing services have been discussed. Our analysis has demonstrated the feasibility for a low-cost cloud enabler for rural areas in the developing nations.

Prometheus is the first solution, targeted to bring computing services from the "cloud" down to the developing and under-developed regions of the world. Its co-design of hardware and software leverages the realization of very compelling and attractive applications that will allow for significant improvements in the life quality of the other half of the world population.

The computational operations of the pico-DC can be assisted on a voluntarily basis by local users, such as schools, business offices and inactive kiosks that

are willing to offer their idle CPU cycles, as long as no technical knowledge is required for the setup of the client software. This collaborative approach does not only provide a more complete and longer sustained operation of our pico-DC framework, but also can add large amounts of computing resources.

Acknowledgments. The authors would like to thank Eric Peterson (Microsoft Research) for his insightful help with the pico-DC node design and Gabriel Kliot (Microsoft Research) and Chia-Chi Lin (Univ. of Illinois at Urbana-Champaign) for their help with the solar power experimentation.

References

1. World Urbanization Prospects: The 2009 Revision, POP/DB/WUP/Rev.2009/1/F2, United Nations, Department of Economic and Social Affairs, Population Division (2009)
2. State of World Population 2007 - Unleashing the Potential of Urban Growth, United Nations Population Fund, New York (2007)
3. Bhawan, M.D.: The Indian Telecom Services Performance Indicators (July-September 2010), Telecom Regulatory Authority of India (2011)
4. Kumar, R.: e-Governance: Drishtee's Soochana Kendras in Haryana, India. In: Proc. South Asia Conference: Trends in Computing for Human Development in India (2005)
5. Greenberg, A., Hamilton, J., Maltz, D.A., et al.: The cost of a cloud: research problems in data center networks. SIGCOMM Comput. Commun. Rev. 39(1), 68–73 (2008)
6. Bhagwat, P., Raman, B., Sanghi, D.: Turning 802.11 inside-out. SIGCOMM Comput. Commun. Rev. 34(1), 33–38 (2004)
7. Sen, S., Raman, B.: Long distance wireless mesh network planning: problem formulation and solution. In: Proc. of the 16th International Conference on World Wide Web, Banff, Alberta, Canada, pp. 893–902 (2007)
8. Matthee, K.W., Mweemba, G., Pais, A.V., et al.: Bringing Internet connectivity to rural Zambia using a collaborative approach. In: International Conference on Information and Communication Technologies and Development, pp. 1–12 (2007)
9. Zhao, W., Ammar, M., Zegura, E.: A message ferrying approach for data delivery in sparse mobile ad hoc networks. In: Proc. of the 5th ACM International Symposium on Mobile Ad Hoc Networking and Computing, Tokyo, Japan, pp. 187–198 (2004)
10. Balasubramanian, A., Zhou, Y., Croft, W.B., et al.: Web search from a bus. In: Proc. of the Second ACM Workshop on Challenged Networks, Montreal, Quebec, Canada, pp. 59–66 (2007)
11. Chen, J., Subramanian, L., Li, J.: RuralCafe: web searching the rural developing world. In: Proc. of the 18th International Conference on World Wide Web, Madrid, Spain, pp. 411–420 (2009)
12. Guo, S., Falaki, M.H., Oliver, E.A., et al.: Very low-cost internet access using KioskNet. SIGCOMM Comput. Commun. Rev. 37(5), 95–100 (2007)
13. Pentland, A., Fletcher, R., Hasson, A.: DakNet: rethinking connectivity in developing nations. Computer 37(1), 78–83 (2004)
14. Chu, D.C., Humphrey, M.: Mobile OGSI.NET: grid computing on mobile devices. In: Proceedings of Fifth IEEE/ACM International Workshop on Grid Computing 2004, pp. 182–191 (2004)

15. Palmer, N., Kemp, R., Kielmann, T., et al.: Ibis for mobility: solving challenges of mobile computing using grid techniques. In: Proc. of the 10th Workshop on Mobile Computing Systems and Applications, Santa Cruz, California, pp. 1–6 (2009)
16. Phan, T., Huang, L., Dulan, C.: Challenge: integrating mobile wireless devices into the computational grid. In: Proc. of the 8th Annual International Conference on Mobile Computing and Networking, Atlanta, Georgia, USA, pp. 271–278 (2002)
17. Clausen, T., Jacquet, P.: RFC 3626 Optimized Link State Routing Protocol, OLSR (2003)
18. Perkins, C., Royer, E., Das, S.: RFC 3561 Ad hoc On- Demand Distance Vector (AODV) Routing (2003)
19. Johnson, D.B., Maltz, D.A.: Dynamic Source Routing in Ad Hoc Wireless Networks. In: Imielinski, T., Korth, H.F. (eds.) Mobile Computing. The Kluwer International Series in Engineering and Computer Science, pp. 153–181. Springer, US (1996)
20. Draves, R., Padhye, J., Zill, B.: Routing in multi-radio, multi-hop wireless mesh networks. In: Proc. of the 10th Annual International Conference on Mobile Computing and Networking, Philadelphia, USA, pp. 114–128 (2004)
21. Longley, A.G., Rice, P.L.: Prediction of tropospheric radio transmission loss over irregular terrain, ESSA Rep.ERL- 79-ITS-67, U.S. Department of Commerce (1968)
22. Mishra, N., Chebrolu, K., Raman, B., et al.: Wake-on-WLAN. In: Proc. of the 15th International Conference on World Wide Web, Edinburgh, Scotland, pp. 761–769 (2006)
23. Shih, E., Bahl, P., Sinclair, M.J.: Wake on wireless: an event driven energy saving strategy for battery operated devices. In: Proc. of the 8th International Conference on Mobile Computing and Networking, Atlanta, Georgia, USA, pp. 160–171 (2002)

Performance Analysis of Client Relay Cloud in Wireless Cellular Networks

Olga Galinina, Sergey Andreev, and Yevgeni Koucheryavy

Tampere University of Technology (TUT), Finland
{olga.galinina,sergey.andreev}@tut.fi,
yk@cs.tut.fi

Abstract. Cooperative communication is a promising concept to mitigate the effect of fading in a wireless channel and is expected to improve performance of next-generation cellular networks in terms of client throughput and energy efficiency. With recent proliferation of smart phones and machine-to-machine communication, so-called 'client relay' cooperative techniques are becoming more important. As such, a mobile client with poor channel quality may take advantage of other neighboring clients, who would relay data on its behalf. In the extreme, the aggregate set of available client relays may form a relay cloud, and members of the cloud may opportunistically cooperate with the data originator to improve its uplink channel quality. The key idea behind the relay cloud is to provide flexible and distributed control over cooperative communication by the wireless clients themselves. By contrast to centralized control, this will minimize extra protocol signaling involved and ensure simpler implementation. In this work, we build an originator-centric model and study the performance of a relay cloud with respect to the main performance metrics: throughput, packet delay, and energy efficiency. We obtain closed-form analytical expressions for the sought metrics and verify our results via extensive simulations.

Keywords: client relay cloud, cellular networks, performance analysis, throughput, energy efficiency.

1 Introduction and Related Work

Various diversity techniques aim at mitigating the negative effects of multipath channel fading in order to improve the reliability of wireless communication link. In particular, one of the most promising techniques for next-generation mobile systems (3GPP LTE-Advanced, IEEE 802.16m) is *spatial transmit diversity* exploiting two or more transmit antennas to enhance the link quality [1]. However, mobile terminals with multiple transmit antennas may be costly due to their size or hardware limitations. For that reason, a concept of *cooperative communication* has been introduced allowing single-antenna mobiles to take advantage of spatial diversity gain and provide so-called *cooperative diversity*.

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 40–51, 2012.

Historically, the core ideas behind cooperative communication were firstly introduced in the fundamental work [2], where a simplified three-terminal system model containing a *sender*, a *receiver*, and a *relay* was studied within the context of mutual information. More thorough capacity analysis of the relay channel was conducted later in [3]. These pioneering efforts focused on the similar three-node case and suggested a number of relaying strategies. They also established achievable regions and upper bounds on the capacity of what we now call the 'classical' relay channel.

With recent proliferation of smart phones and machine-to-machine communication, wireless technology is rapidly evolving toward 4G mobile systems. As such, a renewed surge of interest has come with rapidly expanding literature on cooperation. For example, [4] addressed some further information-theoretic aspects of the relay channel bringing new important insights.

More specifically, cooperative diversity was described in [5] as a relatively new class of spatial diversity techniques that is enabled by relaying and cooperative communication. In [6], authors proposed an efficient cooperation strategy and also explored the concept of cooperation together with some practical issues of its implementation. A good tutorial on cooperative communication may be found in [7].

Some recent works have also addressed a more complicated usage model with multiple wireless clients that may be selected as relays. The problem of relay selection (when data originator may practically have more than one relay to partner with) has been elaborated upon in [8], where the availability of a centralized cooperation-aware controller was assumed. Thus, it brings the concept of cooperation into the scope of wireless cellular networks with a *base station* controlling the activity of its clients.

Further, in [9] several efficient protocols for the relay selection were proposed to recover the multiplexing loss in relay networks, while requiring additional feedback. Evidently, most recent works study cooperation from the perspective of centralized control, which increases extra protocol signaling involved and results in more difficult implementation for the existing systems. By contrast, we concentrate on a more practical scenario with flexible and distributed control over cooperative communication by the wireless clients themselves.

In our previous work [10], we tailored the 'classical' three-node cooperative model to contemporary wireless networks and analyzed primary QoS parameters together with the most important energy-related metrics. In this paper, we continue our efforts by considering a system with multiple clients. The data originator may opportunistically partner with some of those to improve its uplink channel quality. In the extreme, the aggregate set of available relays may form a *relay cloud*. Thus, the goal of our research is to investigate the benefits of the relay cloud and to develop and assess algorithms that will maximize the impact of cooperative communication.

The rest of the paper is organized as follows. Section 2 describes the system model, while giving the main notations and assumptions. In Section 3, we provide theoretical analysis of the client relay cloud and establish the expressions for the

main performance metrics. In Section 4, we consider some numerical results verified by simulation. Finally, Section 5 concludes the paper.

2 System Model

In this section, we model a wireless cellular network consisting of the base station B and several mobile clients (see Figure 1 for the topology and the Table 1 for the notations).

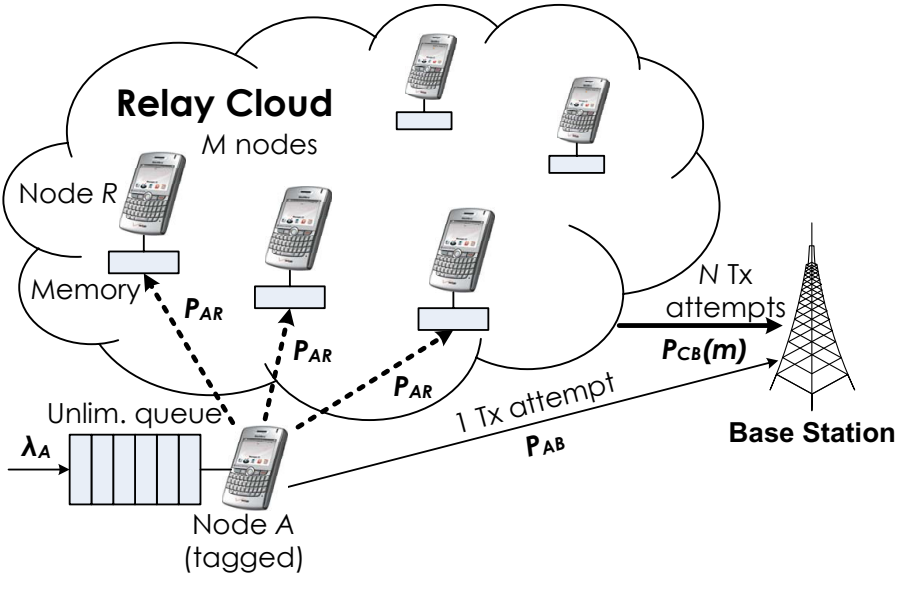


Fig. 1. Illustration of the relay cloud system topology

We define the *cooperative system* as follows. The wireless clients acting as relay nodes are allowed to eavesdrop on the data packets from the originator. As mentioned previously, the aggregate set of available client relays forms a relay cloud. After successful eavesdropping, the members of the cloud may opportunistically transmit on behalf of the data originator to improve its uplink performance. The base station only provides time resources (slots) for such cooperative transmission, whereas opportunistic control resides at the client side. If cooperation is not possible, we term the system *non-cooperative*.

Further on, for the sake of simplicity and without loss of generality we consider the performance of the tagged node A (the data originator). It is assumed, for example, that A is a cell-edge mobile user and thus suffers from the low quality of its uplink channel to the base station. The rest of M neighboring wireless clients (the relay cloud) may potentially perform cooperation acting as relays.

While the originator transmits its initial data packet, each relay node in the cloud may eavesdrop on this packet and store it for subsequent retransmission.

Table 1. Analytical model notations

Notation	Parameter description
λ_A	Mean arrival rate of packets to node A
N	Maximum number of attempts provided by B for cloud transmissions
M	Number of relay nodes in the relay cloud
m	Number of relay nodes in the relay group
p_{AB}	Probability of successful reception at B when A transmits
p_{AR}	Probability of successful reception at R when A transmits
p_{tx}	Opportunistic cooperation probability
$p_{CB}(m)$	Probability of successful reception at B when cloud cooperates
τ_A	Mean service time of a packet from the node A
ρ_A	Queue load coefficient
P_{lossA}	Loss probability of the packet from A
δ_A	Mean packet delay of the packet from A
η_A	Mean throughput of node A
ϵ_A	Mean energy expenditure of node A
ϵ_R	Mean energy expenditure of relay group
ϕ	Mean energy efficiency of the system
P_{TX}	Power level for the transmitting node
P_{RX}	Power level for the eavesdropping node
P_I	Power level for the idle node

The size of extra memory location at each relay is assumed to equal one for every relay session, whereas the size of the outgoing originator buffer is unlimited. In case the originator fails its initial transmission and if eavesdropping is successful, the relay node R decides probabilistically whether to cooperate or not.

The successful relay nodes which decide to cooperate form a so-called *relay group*. We emphasize that the proposed scheme does not require explicit centralized control by the base station and thus minimizes the necessary signaling. The base station may be completely unaware of which nodes belong to the relay group at a particular time instant. Below we detail the system model.

Traffic assumptions. We consider a simple stochastic traffic model to assess the performance of the system and preserve the analytical tractability. As the first step of this research, we assume i.i.d. exponentially-distributed inter-arrival times at the originator (node A). We concentrate on the originator traffic only and abstract out the analysis of own traffic in the relay cloud, which may be more complex. Base station also has no outgoing traffic.

Scheduler assumptions. The system time is slotted. We assume that the packet size equals one and that the transmission of each packet takes exactly one time slot. Scheduling information is immediately available to all the clients (e.g., via a dedicated downlink control channel).

We consider the scheduler operation as follows. As the channel between the node A and the destination is poor, it is very likely that several packet retransmissions may not lead to success. As such, we assume only one attempt to transmit a packet by the originator to save some of its power. If the originator A

has packets, the next time slot is given to A. Upon the transmission, a potential relay may intercept the packet from the originator and store it.

In case node A fails its initial packet transmission, the base station assigns the following slot to the relay cloud so that it could assist the originator. Such assignment repeats until successful delivery or until the number of consecutive cloud retransmission attempts exceeds some maximum number N (a parameter controlled by the base station). In the latter case, all the members of the cloud may empty their memory location and the system considers the current packet as lost.

Channel assumptions. Throughout this paper, we assume immediate feedback over a reliable separate channel (e.g., in the downlink). We also account for the following probabilities of successful delivery p_{AB}, $p_{CB}(m)$ and the symmetric probability for each relay node p_{AR}:

- $p_{AB} = Pr\{\textit{packet from A is received at B}|\textit{only A transmits}\}$,
- $p_{CB}(m) = Pr\{\textit{packet from A is received at B}|\textit{exactly m relays transmit}\}$,
- $p_{AR} = Pr\{\textit{packet from A is received at a given relay}|\textit{only A transmits}\}$.

Let us now illustrate the discussed scheduler operation by an example for $N = 2$ as shown in Figure 2. Firstly, the transmission of packet no. 0 is successful and the system becomes idle. Then, the originator acquires a new packet no. 1 and attempts its uplink transmission, which fails with probability $1 - p_{AB}$. The members of the relay cloud eavesdrop on this transmission and each of them is successful with probability p_{AR} independently. In the following slot, the successful relays make a decision whether or not to help with probability p_{tx}. Those who have decided positively form a relay group that retransmits the eavesdropped packet to the base station simultaneously. As such, a 'virtual MIMO' link with better quality is created due to spatial transmit diversity [11] and the packet is transmitted successfully with probability $p_{CB}(m)$.

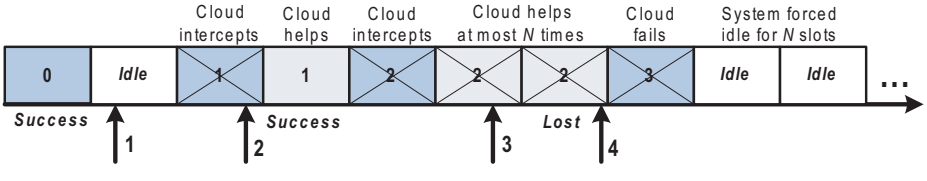


Fig. 2. Example time diagram for the relay cloud system

We generally note that due to the diversity gain the probability $p_{CB}(m)$ is expected to be a nondecreasing function of the relay group size m. Here, m depends on the probabilities p_{AR} and p_{tx}. We implicitly assume that the quality of the "originator-to-base station" channel is low, whereas the quality of the "originator-to-relay cloud" channel is quite good due to many neighboring clients available.

The following slot is given back to the originator A (packet no. 2) and during its unsuccessful transmission the relays intercept the packet again. However, this time the relay cloud is unsuccessful to transmit the packet for $N = 2$ times consecutively. As such, the base station considers the current packet as lost and assigns the next slot to the originator. Further, if the interception fails (packet no. 3) or all the successful relays decide not to transmit twice, $N = 2$ slots are assigned to the relay cloud anyway, but the system stays idle. This is a negative consequence of the distributed control over client relays.

In what follows, we study the mean packet delay, the throughput, and the packet loss probability. In particular, we are interested in the derivation of simple and exact closed-form expressions.

3 Performance Evaluation

This section presents analysis of the relay cloud system with respect to the main performance metrics, such as the mean *number of retransmissions*, the *throughput*, the *packet loss probability*, and the mean *packet delay*.

Firstly, we introduce the following definitions:

Definition 1. The service time is defined as the period of time between the beginning of the first transmission attempt and the moment packet reaches its destination. In case of packet loss, the service time is assumed to be equal N, so that the mean service time could account for the lost packets.

Definition 2. The saturation throughput is defined as the limit reached by the system throughput as the offered load increases [12].

Definition 3. The delay of a packet is defined as the time it takes the packet to reach the destination after it arrives in the system (includes both queueing time and service time).

Definition 4. The energy efficiency is defined as the amount of energy required to successfully transmit one data packet.

Our analytical approach is based on the notion of the service time. We define a stochastic variable T_A, which is the service time of a packet from A. We treat the considered system as an $M/G/1$ system due to the properties of the incoming traffic. Initially, we establish the service discipline and then continue by obtaining the closed-form expressions for the first and the second moments. It should be noted that the first moment is the mean number of the packet transmission attempts.

Knowing both moments, we derive the mean packet delay using the Pollacek-Khinchin formula and the Little's law. The other metrics of interest, such as the throughput, the packet loss probability, the energy expenditure, and the energy efficiency can also be derived from the obtained expressions.

After thorough analysis of all the possibilities for a packet transmission, we formulate the service discipline for the node A as follows:

$$Pr\{T_A = 1\} = p_{AB},$$

$$Pr\{T_A = n\} = (1 - p_{AB}) \sum_{m=1}^{M} \binom{M}{m} p_{AR}^{m} (1 - p_{AR})^{(M-m)} \times$$
$$\times \left\{ \sum_{j=1}^{m} \binom{m}{j} p_{tx}^{j} (1 - p_{tx})^{(m-j)} \left(1 - p_{CB}(j)\right)^{(n-1)} p_{CB}(j) \right\}, n \leq N,$$

$$Pr\{T_A = N + 1\} = (1 - p_{AB}) \sum_{m=1}^{M} \binom{M}{m} p_{AR}^{m} (1 - p_{AR})^{(M-m)} (1 - p_{tx})^{m} +$$
$$+ (1 - p_{AB}) \sum_{m=1}^{M} \binom{M}{m} p_{AR}^{m} (1 - p_{AR})^{(M-m)} \times$$
$$\times \left\{ \sum_{j=1}^{m} \binom{m}{j} p_{tx}^{j} (1 - p_{tx})^{(m-j)} \left(1 - p_{CB}(j)\right)^{N} \right\} +$$
$$+ (1 - p_{AB}) (1 - p_{AR})^{M}.$$

Further, we omit massive transformations and give only the expressions for the first and the second moments of the stochastic variable T_A:

$$\tau_A = E[T_A] = p_{AB} + (N + 1)(1 - p_{AB})(1 - p_{AR})^{M} + \\ + (N + 1)(1 - p_{AB}) \cdot S_1 + (1 - p_{AB}) \cdot S_2, \tag{1}$$

$$E[T_A^2] = p_{AB} + (N + 1)^2 (1 - p_{AB})((1 - p_{AR})^{M} + S_1) + (1 - p_{AB}) \cdot S_3, \tag{2}$$

where components S_1, S_2 and S_3 are given in the Appendix.

The mean load for the queue of the considered tagged node A can be established as:

$$\rho_A = \lambda_A \tau_A. \tag{3}$$

The mean throughput of A may thus be calculated as:

$$\eta_A = \lambda_A (p_{AB} + (1 - p_{AB}) S_4), \tag{4}$$

where component S_4 is also given in the Appendix.

Given the first and second moments, we use the Pollacek-Khinchin formula to obtain the accurate value for the mean packet delay:

$$\delta_A = \tau_A + \frac{E[T_A^2] \lambda_A}{2(1 - \rho_A)}. \tag{5}$$

With the basic formulae for the two moments, we can also find other important metrics. In particular, the packet loss probability is given by:

$$P_{lossA} = 1 - \eta_A \tau_A. \tag{6}$$

Furthermore, let us establish the expressions for the energy consumption. If power level P_i corresponds to a particular power state i, then the normalized energy expenditure per time slot equals $P_i\pi_i$. Here, $\boldsymbol{\pi}$ is the stationary distribution over power states and $i \subset G$, where G is the set of possible states. In the considered model, the three states are accounted for from the power perspective:

- the node is transmitting data with the power P_{TX};
- the node is receiving data with the power P_{RX};
- the node is idle with the power P_I.

Thus, energy expenditures of the tagged node A and of the relay group are calculated as:

$$\epsilon_A = P_{TX}\lambda_A + P_I(1 - \lambda_A\tau_A), \tag{7}$$

$$\begin{aligned}\epsilon_R = P_{RX}M\lambda_A + P_{TX}p_{AR}p_{tx}\lambda_A(\tau_A - 1)M + \\ +P_iM(1 - p_{AR}p_{tx}\lambda_A(\tau_A - 1) - \lambda_A),\end{aligned} \tag{8}$$

Therefore, the total system energy expenditure is given by:

$$\epsilon = \epsilon_A + \epsilon_R. \tag{9}$$

As mentioned above, we define energy efficiency as:

$$\phi = \frac{\eta_A}{\epsilon}. \tag{10}$$

4 Numerical Results

In this section, we use the extended system-level simulator described in our previous works [10] and [11] in order to verify the obtained analytical results. We borrow power consumption values from [13] as: $P_{TX} = 1.65$ W, $P_{RX} = 1.40$ W, and $P_I = 1.15$ W. We also assume that the size of the each slot equals 5 ms.

The main simulation parameters are set as $p_{AB} = 0.3$, $p_{AR} = 0.7$. For the vector of successful delivery probabilities p_{CB}, as an example, we consider a random nondecreasing linear function. However, in reality this function might be much more complicated and surely has a nonlinear structure. Note that solid curves stand for the analytical results, whereas symbols represent simulated data.

In Figure 3, we explore the behavior of the saturation throughput for different number of relay nodes in the cloud. As expected, we observe a monotonically increasing function of M. It is easy to see that beyond the point of $M = 5$ the curve becomes almost linear. Therefore, we set $M = 5$ in what follows.

Further, we explore the mean packet delay at the originator for different numbers relay of nodes available. For this purpose, we vary the arrival rate λ_A. In Figure 4, different curves for the various values of M (with appropriate asymptotes) are compared. Naturally, the delay drops significantly as the number of available relays grows.

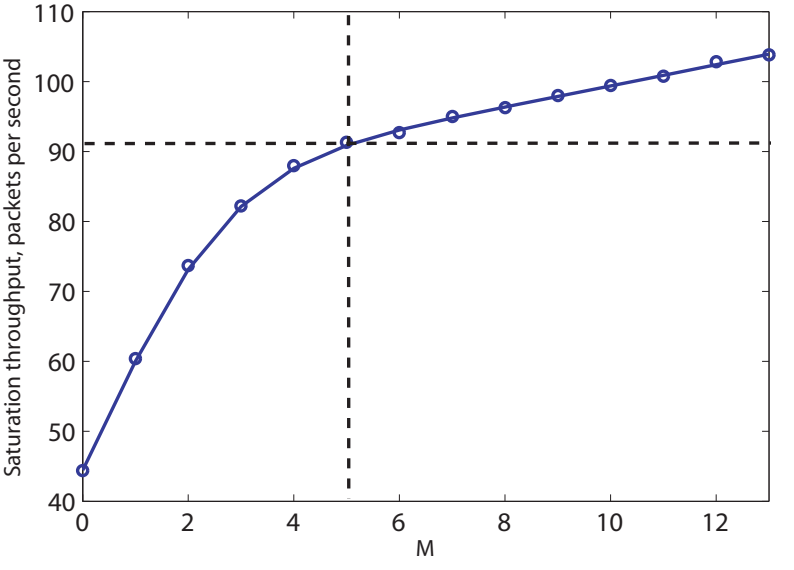


Fig. 3. Saturation throughput vs. number of nodes in relay cloud

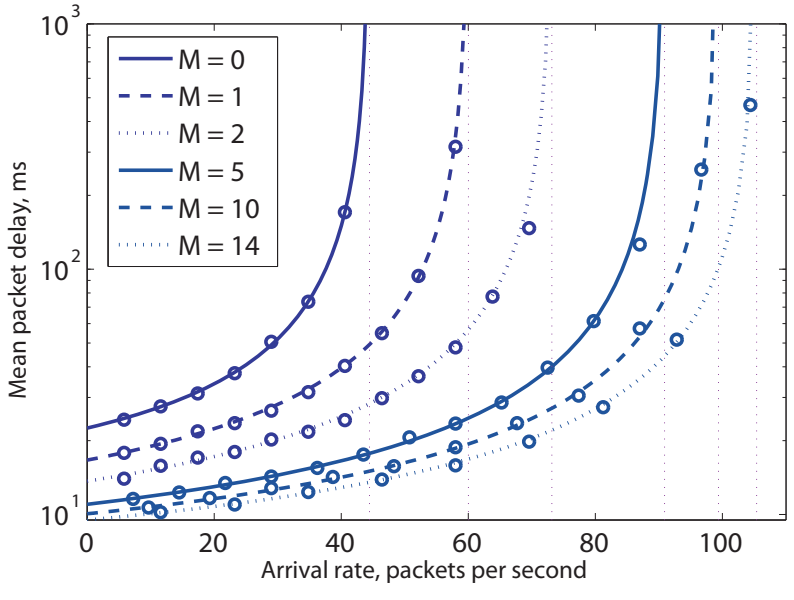


Fig. 4. Mean packet delay vs. arrival rate λ_A for different values of M

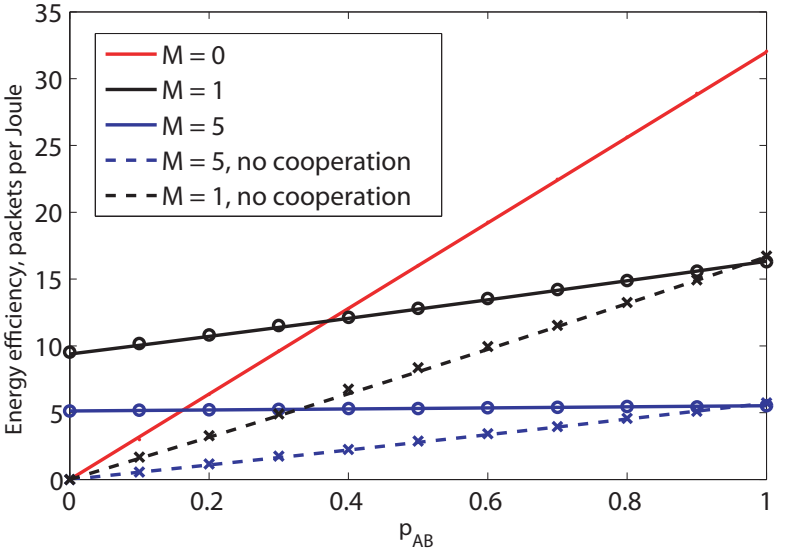


Fig. 5. Energy efficiency vs. probability of successful packet transmission

Also we study the energy efficiency dependence on e.g. the probability p_{AB} in Figure 5. Here, we contrast the non-cooperative mode (when there are no relay nodes) against the systems with $M = 1$ and $M = 5$. Evidently, the assistance of the relay cloud results in slightly higher energy expenditure, which is the cost of the increased originator performance.

5 Conclusion

In this paper, we considered a wireless cellular network that enables the distributed control over cooperative communication via a client relay cloud. The primary aim of such cloud is to enhance system performance through the support of cell-edge mobile clients with poor communication links. The main performance metrics were studied, including throughput, mean packet delay, and energy efficiency. Accurate closed-form analytical expressions have been derived and verified by extensive system-level simulations. The results indicate significant promise of the relay cloud, which is able to recover the performance of the mobile clients with degraded wireless links. As a future extension of this model, it would be reasonable to examine a more realistic arrival process and propose efficient decision-making algorithms on when to co-operate. Also it is important to establish practical scenarios where client relay cloud operation is benefiting the wireless system performance and where it is not.

References

1. LTE Release 10 & beyond (LTE-Advanced)
2. Van Der Meulen, E.C.: Three-terminal communication channels. Advances in Applied Probability 3, 120–154 (1971)
3. Cover, T.M., El Gamal, A.A.: Capacity theorems for the relay channel. IEEE Transactions on Information Theory 25(5), 572–584 (1979)
4. Kramer, G., Gastpar, M., Gupta, P.: Cooperative strategies and capacity theorems for relay networks. IEEE Transactions on Information Theory 51(9), 3037–3063 (2005)
5. Laneman, J., Tse, D., Wornell, G.: Cooperative diversity in wireless networks: Efficient protocols and outage behavior. IEEE Transactions on Information Theory 50(12), 3062–3080 (2004)
6. Sendonaris, A., Erkip, E., Aazhang, B.: User cooperation diversity. Part I, II. IEEE Transactions on Communications 51(11), 1927–1938 (2003)
7. Nosratinia, A., Hunter, T., Hedayat, A.: Cooperative communication in wireless networks. IEEE Communications Magazine 42(10), 74–80 (2004)
8. Nosratinia, A., Hunter, T.: Grouping and partner selection in cooperative wireless networks. IEEE Journal on Selected Areas in Communications 25(2), 369–378 (2007)
9. Tannious, R., Nosratinia, A.: Spectrally efficient relay selection with limited feedback. IEEE Journal on Selected Areas in Communications 26(8), 1419–1428 (2008)
10. Andreev, S., Galinina, O., Lokhanova, A., Koucheryavy, Y.: Analysis of Client Relay Network with Opportunistic Cooperation. In: Masip-Bruin, X., Verchere, D., Tsaoussidis, V., Yannuzzi, M. (eds.) WWIC 2011. LNCS, vol. 6649, pp. 247–258. Springer, Heidelberg (2011)
11. Andreev, S., Galinina, O., Vinel, A.: Performance evaluation of a three node client relay system. International Journal of Wireless Networks and Broadband Technologies, IJWNBT 1(1), 73–84 (2011)
12. Bianchi, G.: Performance analysis of the IEEE 802.11 distributed coordination function. IEEE Journal on Selected Areas on Communications 18, 535–547 (2000)
13. Andreev, S., Galinina, O., Koucheryavy, Y.: Energy-efficient client relay scheme for machine-to-machine communication. In: Proceedings of GLOBECOM (2011)

Appendix 1: Auxiliary Variables

We introduce the following auxiliary variables in order to simplify the expressions above.

$$S_1 = \sum_{m=0}^{M} \binom{M}{m} p_{AR}^{m} (1 - p_{AR})^{(M-m)} \times \tag{11}$$
$$\times \left\{ \sum_{j=0}^{m} \binom{m}{j} p_{tx}^{j} (1 - p_{tx})^{(m-j)} (1 - p_{CB}(j)) + (1 - p_{tx})^{m} \right\}^{N}.$$

$$S_2 = \sum_{m=0}^{M} \binom{M}{m} p_{AR}^{m} (1 - p_{AR})^{(M-m)} \cdot X \cdot b, \tag{12}$$

where

$$X = a^N - \frac{a^{(N+1)}}{(1-a)^2} - \frac{(a-2)}{(1-a)^2} - a^N \frac{(N+2)}{(1-a)}. \tag{13}$$

$$S_3 = \sum_{m=0}^{M} \binom{M}{m} p_{AR}^{m} (1 - p_{AR})^{(M-m)} \cdot Z \cdot b, \tag{14}$$

where

$$Z = X + \frac{2(2a - a^{(N+1)}(N+2))}{(1-a)^2} + \frac{2(a^2 - a^{(N+2)})}{(1-a)^3} + \frac{2 - a^N (N+1)(N+2)}{(1-a)}. \tag{15}$$

The following variable is the probability of the successful transmission by the relays:

$$S_4 = \sum_{m=0}^{M} \binom{M}{m} p_{AR}^{m} (1 - p_{AR})^{(M-m)} \cdot Y \cdot b, \tag{16}$$

where

$$Y = \frac{(1 - a^N)}{(1-a)}. \tag{17}$$

Here, also for the sake of brevity the enlarged variables a and b are defined as:

$$a = \sum_{j=0}^{m} \binom{m}{j} p_{tx}^{j} (1 - p_{tx})^{(m-j)} (1 - p_{CB}(j)) + (1 - p_{tx})^{m}, \tag{18}$$

$$b = \sum_{j=0}^{m} \binom{m}{j} p_{tx}^{j} (1 - p_{tx})^{(m-j)} p_{CB}(j). \tag{19}$$

Periodic Scheduling with Costs Revisited
A Novel Approach for Wireless Broadcasting

Christos Liaskos[1], Andreas Xeros[2], Georgios I. Papadimitriou[1], Marios Lestas[3], and Andreas Pitsillides[2]

[1] Dept. of Informatics, Aristotle University of Thessaloniki, 54124 Greece
{cliaskos,gp}@csd.auth.gr
[2] Dept.of Comp. Science, CY University, P.O. Box 20537 1678, Nicosia Cyprus
{csp6xa1,andreas.pitsillides}@cs.ucy.ac.cy
[3] Department of Electrical Engineering at Frederick University, Nicosia, Cyprus
eng.lm@frederick.ac.cy

Abstract. Periodic broadcast scheduling typically considers a set of discrete data items, characterized by their popularity, size and scheduling cost. A classic goal is the definition of an infinite, periodic schedule that yields minimum mean client serving time and minimum mean scheduling cost at the same time. This task has been proven to be NP-Hard and more recent works have discarded the scheduling cost attribute, focusing only on the minimization of the mean client serving time. In the context of the present work the scheduling cost is reinstated. An analysis-based scheduling technique is presented, which can practically minimize the mean client serving time and the mean scheduling cost concurrently. Comparison with related approaches yields superior performance in all test cases.

Keywords: periodic scheduling, broadcast cost, wireless transmission.

1 Introduction

Broadcasting is an efficient means of bandwidth preservation and information advertising in both wired and wireless environments [1, 2]. In the latter case, which is examined in the present work, broadcasting is much more vital, since there exists only one wireless medium. Thus, overcoming bandwidth limitations by adding more physical paths in not an option. The importance of broadcasting has highlighted the need for efficient broadcast scheduling, i.e. the proper serialization of data item transmission in order to ensure minimal client serving time and efficient quality of service.

The evaluation of wireless broadcast scheduling techniques takes place in a widely approved system setup [2–17]. A number of wireless clients freely roam an area covered by a broadcast network. All clients read the broadcasted data stream synchronously, while wireless transmission parameters are idealized in order to focus on the evaluation of the scheduling process only. The dataset to be broadcasted contains a number of discrete data items with known sizes. All the

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 52–63, 2012.

aforementioned parameters may vary with time. Proposed scheduling techniques typically have an online expression [2–5], which can re-adapt the scheduling on the fly, based on new input regarding the client preferences or the update of the data set. The change in these parameters can be detected and handled by smart learning algorithms which are examined elsewhere, as a separate field of study [6,18]. Two types of scheduling are defined: in pull-based scheduling the clients pose specific queries to a server [19]. The server then serializes the answers to the requests in a way that minimizes the mean serving time. In push-based (or periodic) scheduling, examined in the present study, the clients do not post queries to the server. The learning algorithm monitors the request probability of each available *data item* (or item class), typically through a lightweight, indirect feedback system [2]. (E.g. by exploiting Facebook profile data in a subscription-based system). Thus the actual number of users is not directly relevant [5]. The goal is then to create a periodic schedule that minimizes the mean serving time of the clients, based on the request probability distribution of the data items.

The problem with existing periodic scheduling approaches is that they do not take into account important, practical parameters, as for example copyright costs per item transmission or computational cost per item scheduling. A realistic scheduling authority will strive for a balance between client satisfaction and broadcast cost. This aspect of the scheduling process was originally taken into account in [9], where an additional attribute, the broadcast cost, was assigned to each data item. The set goal was the creation of the schedule that minimizes the mean client serving time and the mean scheduling cost at the same time. However, the problem was soon proven to be NP-Hard [7]. Subsequent studies validated the NP-Hardness of several variations of the original problem [8]. In the context of the studied push-based scheduling, the cost attribute was then discarded, and related works focused on the minimization of the mean serving time (or related metric) only [2,4–6,12–17]. In this case, the assumption of very large schedule sizes ("infinity" assumption) was adopted as a means of facilitating the mathematical analysis of the simplified problem [5]. This in turn resulted into unrealistically high computational requirements [4].

In the context of the present work we reinstate the broadcast cost attribute, and present a scheduler than can achieve minimal mean client serving time for any user-defined mean scheduling cost. Furthermore, it is proven that this task does not require infinite schedules, thus decreasing the overall computational complexity by orders of magnitude. The proposed scheduler in shown to be more efficient than the existing approaches, in all test cases. It is clarified that the proven theoretical NP-Hardness of the problem is not alleviated, but rather shown not to be prohibitive in practical communication systems.

1.1 Related Work

Research on push-based, wireless broadcast systems initially focused on the minimization of the clients' mean serving time, over an infinite time horizon, in the context of Teletext systems [10]. It was proven that an optimal schedule is also

periodic. Therefore, one needs to define only the optimal number of occurrences of each data item inside the schedule. [10] provided a solution, assuming equally-sized data items. The problem was revisited in [11], heuristically studying items with small variation in their sizes. It was clarified that the mean serving time depends on data item attributes (i.e. item request probabilities and sizes), and not on the number of clients. The same study proposed scheduling algorithms that achieved optimality at the expense of increased complexity ($O(N \cdot B)$, N being the number of data items and $B > N$ the number of scheduled broadcasts). These algorithms worked for small variations in item sizes only. Authors in [4] presented an analysis-derived periodic scheduler that achieved the same performance with $O(N)$ complexity, for any item sizes. In the same study it was also shown that the schedule size is a determining factor of the overall scheduling complexity. Simulations indicated that finite schedules may also achieve optimality.

Heuristic, low complexity scheduling methods were introduced in [3] with the introduction of the Broadcast Disks model. According to it, items are grouped by popularity, forming virtual disks rotating around a common axis. Imaginary heads read and serialize data from the disks, producing the final schedule. In [12], the authors applied clustering techniques for performing the data grouping. In [13] the grouping of items was analytically optimized. The analytical results were exploited in [14] for producing minimal complexity schedulers. All aforementioned studies focused on the minimization of the clients' mean serving time.

As previously stated, a more strict version of the scheduling problem assigns an additional attribute, the scheduling cost, to each data item. It is clarified that the scheduling cost is not related to broadcast deadlines, an extension of the mean serving time problem [15]. The new goal is to define the schedule that minimizes the mean serving time and the mean scheduling cost at the same time. The authors in [8] map the updated broadcast scheduling procedure to the generalized maintenance scheduling problem, which is a known NP-Hard problem. Several greedy algorithms are presented, as well as in [7], which generally operate beyond the analytically optimal bounds. To the best of the authors knowledge, no studies since [8] have attempted a practical solution for push-based systems, possibly due to the proof of NP-Hardness.

Standard Assumptions and Notation

We regard a set of N data items arbitrarily indexed by $i = 1...N$. Each item i is associated with its size l_i (in *bytes*) and its request probability p_i, $\sum_{i=1}^{N} p_i = 1$. Finally, $u_i \in \mathbb{N}^*$ will denote the number of occurrences of item i in the schedule.

No assumptions are made concerning the nature of a data item during the analysis. In accordance with the related work on scheduling, an item is simply a piece of information that a client may acquire through a single query [2–5,7, 8,10–17,20,21]. It is clarified that in push-based, periodic broadcast scheduling, the term "*client query*" does not imply posting a request to a server, but rather waiting for the broadcast of a specific item. According to [4], the mean serving

time achieved by a schedule is given by:

$$\overline{W} = \frac{1}{2} \cdot \left(\sum_{i=1}^{N} u_i \cdot l_i \right) \cdot \left(\sum_{i=1}^{N} \frac{p_i}{u_i} \right) \tag{1}$$

Notice that $\overline{W}$ does not depend on the number of clients, which are handled collectively as a Gaussian process via the central limit theorem [5]. In addition, (1) measures $\overline{W}$ in size units (e.g. bytes). Conversion to time units requires knowledge of the physical wireless transmission rate. The mean serving time is minimized when the item occurrences in the schedule follow the relation [11]:

$$u_i(\lambda) = \left[\left[\lambda \cdot \sqrt{\frac{p_i}{l_i}} \right] \right], i = 1 \ldots N, \ \lambda >> 1 \tag{2}$$

where $[\![.]\!]$ is the rounding function. Equation (2) is also known as *the square root rule*, and the condition $\lambda >> 1$ expresses the schedule size "infinity" assumption [5,7,8,10,11]. In order to avoid the nullification of the item occurrences, $u_i, \forall i$, it must generally hold that $\lambda \in \left(max \left\{ \frac{\sqrt{l_i}}{2 \cdot \sqrt{p_i}} \right\}, \infty \right)$. Finally, for a given λ, the total size of the schedule is expressed as:

$$L(\lambda) = \sum_{i=1}^{N} u_i(\lambda) \cdot l_i \tag{3}$$

which is minimized when $u_i = 1, \forall i$.

The remainder of this paper is organized as follows: Section 2 presents the analysis leading to the definition of the novel, cost-aware optimal scheduler. Comparison with related work takes place in Section 3. Conclusion is given in Section 4.

2 Analysis of the Proposed Scheduling Scheme

We assign an additional, normalized attribute, $\alpha_i \in (0, 1)$, to each of the data items $i = 1 \ldots N$. This attribute corresponds to the scheduling cost of [7,8], and is open to any physical interpretation that can be efficiently expressed in the value set $(0, 1)$. Given a broadcast schedule of the available items, the mean cost is:

$$\overline{\alpha} = \frac{\sum_{i=1_i}^{N} u_i \cdot \alpha_i}{\sum_{i=1}^{N} u_i} \tag{4}$$

Notice that $\overline{\alpha}$ is minimized when $u_i = 0, \forall i \neq argin f_{(j)}\{a_j\}$, i.e. when we exclusively broadcast the item with the lowest cost. On the other hand, $\overline{W}$ is minimized when the relation (2) holds. Furthermore, the cost attributes, α_i, are not correlated in any way to the remaining item attributes, p_i, l_i. Therefore, minimizing $\overline{\alpha}$ and $\overline{W}$ concurrently requires the definition of a metric that combines both quantities. For example, [7] and [8] assume that both $\overline{\alpha}$ and $\overline{W}$ are of equal importance and define the combination:

$$S = 50\% \cdot \overline{\alpha} + 50\% \cdot \overline{W} \tag{5}$$

which needs to be minimized. However, the equal importance assumption is restrictive. In order to overcome this shortcoming, the following analysis will derive the full relation $\overline{W} = f(\overline{\alpha})$, i.e. the schedule that minimizes $\overline{W}$ for any given cost $\overline{\alpha}$. Any custom metric can then be satisfied at the intersection of the plot $\overline{W} = f(\overline{\alpha})$ and the line $\overline{W} = b \cdot \overline{\alpha}$, $b > 0$. As an example, the combination S of eq. (5) represents the very specific case of intersecting $\overline{W} = f(\overline{\alpha})$ and $\overline{W} = \overline{\alpha}$.

In order to enable the use of infinitesimal calculus, we will expand the value set of u_i from $\mathbb{N}$ to $\mathbb{R}_*^+$. Indeed, if u_i is extremely large $\forall i$, one can safely assume that $u_i \pm 0.5 \approx u_i$, where ± 0.5 represents any rounding error. Equation (4) then relates u_i to any arbitrary u_j, $j \in 1 \dots N$, $j \neq i$ as follows:

$$\frac{\partial u_j}{\partial u_i} = -\frac{\overline{\alpha} - \alpha_i}{\overline{\alpha} - \alpha_j} \tag{6}$$

Taking the first derivative of the mean client serving time, $\overline{W}$ (equation (1)), with regard to u_i, produces through (6):

$$\frac{\partial \overline{W}}{\partial u_i} = \frac{1}{2}\left[l_i - l_j \frac{\overline{\alpha} - \alpha_i}{\overline{\alpha} - \alpha_j}\right]_{A_1} \cdot \left(\sum_{k=1}^{N} \frac{p_k}{u_k}\right)_{B_1} + \dots$$
$$\dots + \frac{1}{2}\left(\sum_{k=1}^{N} u_k \cdot l_k\right)_{B_2} \cdot \left[-\frac{p_i}{u_i^2} + \frac{p_j}{u_j^2}\frac{\overline{\alpha} - \alpha_i}{\overline{\alpha} - \alpha_j}\right]_{A_2} \tag{7}$$

The reader may notice that the described procedure is an application of the Lagrange method of restricted optimization, on equations (1) and (4). The labels $A_{1,2}$, $B_{1,2}$ are added for quick referencing. Concerning the possible nullification of the $(\overline{\alpha} - \alpha_j)$ denominator, we simply note that α_j (i.e. reference item j) can be chosen to be different from the user-defined mean cost $\overline{\alpha}$, which is supplied as an input. We proceed to define Theorem 1:

Theorem 1. *Assume a large schedule (infinity assumption) and the request for minimal mean client serving time, $\overline{W}$, for a given mean scheduling cost, $\overline{\alpha}$. The corresponding optimal item occurrences are:*

$$u_i^{opt} = \left[\left[\lambda \cdot \sqrt{\frac{p_i}{l_j \cdot \frac{\overline{\alpha} - \alpha_i}{\overline{\alpha} - \alpha_j}}}\right]\right], \; j = argin f_{(i)} \left\{|\widetilde{\alpha} - \alpha_i|\right\}, \; \lambda >> 1 \tag{8}$$

where $\widetilde{\alpha}$ is the median of the $\{\alpha_i\}$ values.

Proof. We proceed to insert equation (8) in factor A_2 of (7). It is deduced after trivial calculations that $A_2 = 0$. Examining factor B_1 of equation (7) and statement $\lambda >> 1$ of (8), we derive that $B_1 \to 0$ due to the infinity assumption, while A_1 is finite. Therefore, the derivative of (7) is nullified, yielding optimality. This concludes the proof.

Remarks: The choice of the reference item j in equation (8) is not mandatory, but favors a restriction imposed by the square root of (8). i.e.:

$$l_i^{'} = l_j \cdot \frac{\overline{\alpha} - \alpha_i}{\overline{\alpha} - \alpha_j} > 0 \Longleftrightarrow \frac{\overline{\alpha} - \alpha_i}{\overline{\alpha} - \alpha_j} > 0, \forall i \tag{9}$$

which has the equivalent representation $(\overline{\alpha} - \alpha_i) \cdot (\overline{\alpha} - \alpha_j) > 0$. This restriction is not upheld when $\overline{\alpha}$ is inside the interval defined by α_j and α_i, $\forall i$. The choice of j as the item whose cost is nearest to the median, $\widetilde{\alpha}$, ensures that the interval between α_j and α_i, $\forall i$ is as small as possible, thus limiting possible issues.

However, the scheduling authority may require a mean cost $\overline{\alpha}$ that is indeed in the aforementioned interval. In this case we can exploit the fact that the infinity assumption makes for the existence of additional solutions. The reader may exemplary notice that inserting the transformation:

$$l_i^{'} = \left| l_j \cdot \frac{\overline{\alpha} - \alpha_i}{\overline{\alpha} - \alpha_j} \right| \tag{10}$$

in (8), also nullifies the derivative of (7), since the factors (B_1) and $(A_2 \cdot B_2)$ approach zero as λ (and therefore $u_i, \forall i$) increases. The drawback of this solution is that it requires much larger schedule sizes, since the factor $(A_2 \cdot B_2)$ must now also be nullified indirectly, by increasing the size of the schedule.

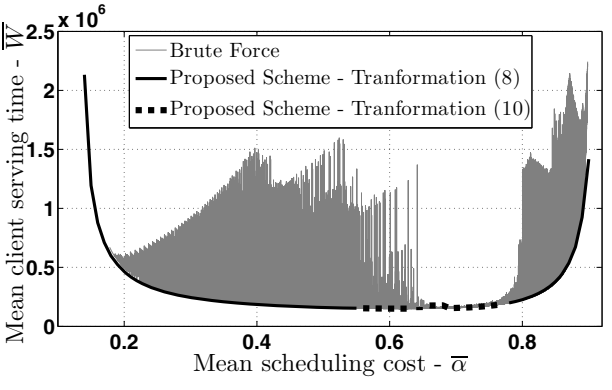


Fig. 1. Comparison of the proposed scheduler with results derived via brute force. For each given mean cost, $\overline{\alpha}$, the simple transformation of (9) produces a mean serving time, $\overline{W}$, that is optimal by brute-force standards. The dotted line corresponds to the use of transformation (10) in the cases where restrictions (9) do not apply. The mean serving time is measured in *Bytes*.

Figure 1 presents an indicative comparison of the proposed scheduling technique with the results derived via brute-force. We assume $N = 5$ items (a restriction enforced by the brute force procedure), with random sizes $l_{1-5} \in [10, 100] Kbyte$, random request probabilities, $\sum_{i=1}^{5} p_i = 1$, and random scheduling costs, $\alpha_{1-5} \in (0, 1)$. We simulate 1000 wireless clients listening to the same

broadcast schedule, each one waiting for the successive completion of 600 personal queries for items $1-5$. The creation of the queries obeys to the predefined, random request probabilities p_{1-5}. As in [2–17], we focus on the simulation of the scheduler. Therefore, it is assumed that there are no coverage, noise or interference issues, which could cause an altered perception of the efficiency of the proposed scheduler.

We seek a schedule with mean cost $\overline{\alpha} = 0.01 : 0.01 : 1$ and a minimum corresponding mean serving time, $\overline{W}(\overline{\alpha})$. For each $\overline{\alpha}$ value, we calculate the optimal number of item occurrences, u_i, through equation (8). Transformation (10) is employed only when (8) fails to comply with (9). Once the optimal u_i have been calculated, we can use any serialization technique which targets the creation of periodic schedules. In the case of Fig. 1 we adopt the serialization scheme of [4]. This serializer receives the desired item occurrences, u_i, and the item sizes, l_i, as input, and produces a periodic schedule by employing preemption. In the case of brute force we use the same serialization scheme, but we try all possible $u_{1-5} = 1 : 1 : 500$ combinations. Then, for each achieved mean scheduling cost, we log the smallest achieved mean serving time. The results of Fig. 1 yield convergence between the proposed scheme and the brute-force approach. Small discrepancies of the brute-force results are attributed to the computational limitations of the procedure.

The almost perfect results presented in Fig. 1 do not in any way falsify the proof of the NP-Hardness of the scheduling problem [8]. They do however pose a question of whether purely theoretical assumptions have magnified the practical significance of the issue. It has been proven that an optimal schedule is periodic: the interval between two consecutive occurrences of an item must be constant [10]. Therefore, knowledge of the u_i ratio is at first sufficient for creating an optimal schedule. However, large variations in the size of the data items may hinder periodicity. (E.g. a huge file may not fit in its predefined, periodic positions) [4]. The NP-Hardness stems from the resulting combinatorial problem of finding the optimal item serialization in this case. This issue is not a practical hindrance however: in the vast majority of the modern communication systems, large files are divided in much smaller segments or packets, prior to transmission. This approach is known to resolve the aforementioned issue in periodic scheduling as well [4]. For limited item size variations, non-preemptive serializers (e.g. [5]) produced identical, optimal results. Conclusively, having shown that the definition of the optimal u_i ratios is easily tractable, the authors claim that the undisputed, theoretical NP-Hardness of the scheduling problem does not pose a significant limitation in practice.

3 Comparison with Related Work

The proposed scheduler is compared with related approaches in terms of:

– **Tunability**. A scheduler should be able to operate at any $\{\overline{W}, \overline{\alpha}\}$ (mean serving time, mean cost) combination dictated by the scheduling authority.

- **Efficiency**. The scheduler should achieve minimal mean client serving time, $\overline{W}$, for any selected mean scheduling cost, $\overline{\alpha}$.
- **Complexity**. A scheduler should require the least possible computational resources for the production of the chosen schedule. As proven in [4,14], the size of a schedule defines the overall complexity and, therefore, should be kept minimal.

The proposed scheduler is compared to the *Randomized*, *Greedy* and *Periodic* schedulers of [7,8]. These schedulers are cost-aware, assigning a single attribute α_i per data item as well, and represented the most viable options prior to the present work. As a reference point, we include the cost-agnostic schedulers of [5] and [4], which target solely the minimization of the mean client serving time, disregarding the cost. The scheduler of [4] surpassed [5] in terms of complexity and performance, and represents a state-of-the-art (SOA) algorithm for periodic scheduling. Notice that [4] comprises a scheduling technique, and a serialization technique. The latter is generic and reusable, as discussed in the context of Fig. 1. We use this serialization scheme in all applicable compared approaches for fairness reasons. However, each compared approach has its own scheduling scheme, i.e. a way of setting the optimal item occurrences, u_i.

We assume the same simulation configuration employed previously, in the experiments of Fig. 1. The number of data items is raised to $N = 50$, and their request probabilities are set by the ZIPF distribution with a skewness factor of 0.6 [22]. Their sizes are picked randomly in $[10, 100]KBytes$, and theirs costs in $(0, 1)$. The topology, the number of clients and the number of queries remain unaltered.

We examine the values $\overline{\alpha} = [0.01 : 0.01 : 1]$ as possible requests for the mean scheduling cost. For each value we require from the compared schedulers to create corresponding optimal schedules, and we log the achieved mean client serving times (MST) and schedule sizes. The results are shown in Fig. 2 and 3.

The proposed scheduler can produce multiple optimal schedules per $\overline{\alpha}$ choice. This is due to the fact that the serialization technique of [4] can also fine-tune the size of the produced schedule, with trivial impact on the efficiency (smaller schedules-slightly higher mean serving time/cost). This phenomenon creates the grayed surface of possible operation points for the proposed solution in Fig 2 and 3. In terms of tunability, the proposed scheduler flawlessly covers the complete range of tested $\overline{\alpha}$ values. All other algorithms however present zero tunability, being able to operate only at one, not user defined mean cost. This is not surprising for the cost agnostic algorithms, but is not in favor of the cost-aware ones. Furthermore, the cost-awareness of the latter ones does not have a significant overall impact, as their operation points nearly coincide with the those of the cost-agnostic scheduler of [5]. This issue is more observable in the top view of Fig. 3.

The studies of [7,8], where the related cost-aware schedulers are proposed, do not target the minimization of the mean client serving time for a given cost. Instead, both studies seek to minimize the sum $S = 50\% \cdot \overline{\alpha} + 50\% \cdot \overline{W}$ of equation (5). This results into only one possible operation point and zero tunability.

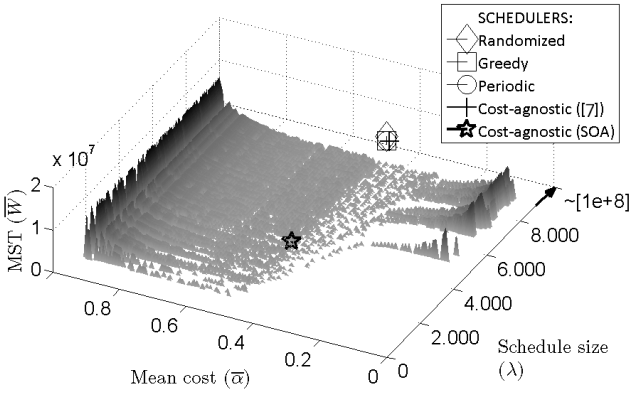


Fig. 2. Tunability of the compared schedulers in terms of possible operation points, $\{\overline{\alpha}, \overline{W}, \lambda\}$. The proposed scheduler can efficiently cover all requests for scheduling cost $\overline{\alpha} = [0.01 : 0.01 : 1]$, enabling application-specific, fine grained balancing between performance and cost (grayed plane). The related, cost-aware approaches are limited to one, not-user defined point of operation, with deterring corresponding schedule size. The arrow corresponds to a significant leap in the y-axis for presentational purposes.

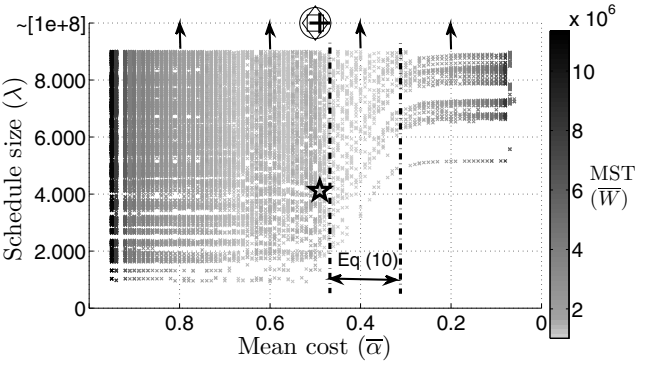


Fig. 3. Top view of Fig. 2. Even though cost-aware, the Randomized, Periodic and Greedy schedulers do not offer significant advantages over the cost-agnostic approaches. In fact, SOA may be a better choice instead, because of the decreased schedule size. The use of transformation (10) causes increase in the size of the schedules produced by the proposed scheduler, as expected by the analysis. Notice that the arrows correspond to a significant leap in the y-axis for presentational purposes.

Furthermore, with the NP-Hardness of the problem a given, these studies mainly target the proposal of a very lax (and therefore suboptimal) but safely achievable lower bound of the S quantity. Consequently, the *Randomized*, *Greedy* and *Periodic* schedulers presented therein do not behave significantly better than the cost-agnostic approaches [4,5].

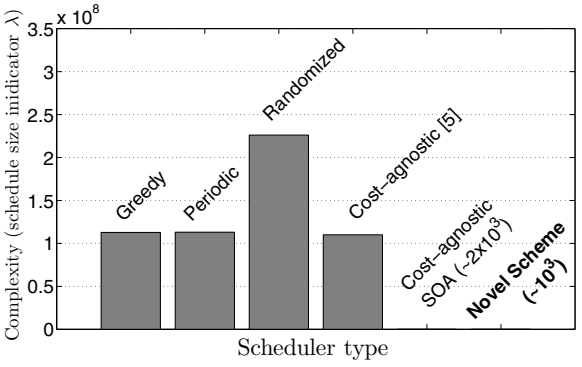


Fig. 4. Required schedule sizes, in the case of $\overline{\alpha} = 0.47$. As shown in Fig. 3, this case represents the sole possible operation point of the compared, related schedulers. The novel scheme requires $\sim 10^5$ times less computational resources. Randomized item serialization hurts periodicity and, utterly, performance.

As an additional remark on the results of Fig. 3, notice that the region between the vertical, dotted lines designates the use of transformation (10) by the proposed scheduler. According to the analysis of Section 2, the use of this transformation yields optimality at the expense of increased schedule size. The results concur to this claim and the possible operation points are sparser inside the designated region as well.

Concerning the size of the produced schedules (and therefore the complexity of the corresponding schedulers), the novel scheduler achieves smaller sizes by 5 orders of magnitude. This fact is observable in Fig. 2 and 3, clarifying that the black arrows represent a leap in λ values by five orders of magnitude ($\lambda = 9000 \rightarrow \lambda \approx 10^8$), for presentational purposes. The difference in complexity is better illustrated in Fig. 4. The figure presents the required schedule sizes in the case of $\overline{\alpha} = 0.47$, i.e. the only possible operation point for the related schedulers. It is evident that the computational requirements for the scheduling procedure can be reduced by a factor of 10^5, without significant impact on performance. The randomized scheduler of [8] produces the worst results, since random item serialization may hurt periodicity, which is a prerequisite for optimality [10].

4 Conclusion

The present study reinstated the broadcast cost per data item as a vital factor of the periodic scheduling process. A novel scheduler was proposed which can achieve minimal client serving times for any requested mean scheduling cost. The computational requirements of the scheduler were decreased by reducing the size of the produced schedule. Conclusively, the study demonstrated that the theoretically NP-Hard problem of periodic, push-based broadcast scheduling with costs may have an efficient solution in practice. Comparison with related approaches yielded improved performance, combined with fine-grained operational tunability.

Acknowledgment. This research has been co-financed by the European Union (European Social Fund – ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) - Research Funding Program: Heracleitus II. Investing in knowledge society through the European Social Fund.

References

1. 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 ACM Conference on Emerging Network Experiment and Technology (CoNEXT 2009), Rome, Italy, pp. 1–12 (December 2009)
2. Nicopolitidis, P., Papadimitriou, G., Pomportsis, A.: Continuous Flow Wireless Data Broadcasting for High-Speed Environments. IEEE Transactions on Broadcasting 55(2), 260–269 (2009)
3. Acharya, S., Alonso, R., Franklin, M., Zdonik, S.: Broadcast disks. ACM SIGMOD Record 24(2), 199–210 (1995)
4. Liaskos, C., Petridou, S., Papadimitriou, G.: Towards Realizable, Low-Cost Broadcast Systems for Dynamic Environments. IEEE/ACM Transactions on Networking 19(2), 383–392 (2011)
5. Vaidya, N.H., Hameed, S.: Scheduling data broadcast in asymmetric communication environments. Wireless Networks 5(3), 171–182 (1999)
6. Kakali, V.L., Sarigiannidis, P.G., Papadimitriou, G.I., Pomportsis, A.S.: A Novel Adaptive Framework for Wireless Push Systems Based on Distributed Learning Automata. Wireless Personal Communications 57(4), 591–606 (2011)
7. Schabanel, N.: The Data Broadcast Problem with Preemption. In: Reichel, H., Tison, S. (eds.) STACS 2000. LNCS, vol. 1770, pp. 181–192. Springer, Heidelberg (2000)
8. Kenyon, C., Schabanel, N.: The Data Broadcast Problem with Non-Uniform Transmission Times. Algorithmica 35(2), 146–175 (2002)
9. Gondhalekar, V., Jain, R., Werth, J.: Scheduling on airdisks: efficient access to personalized information services via periodic wireless data broadcast. In: Proceedings of the 1997 International Conference on Communications (ICC 1997), Montreal, Canada, pp. 1276–1280. IEEE (June 1997)
10. Gecsei, J.: The Architecture of Videotex Systems. Prentice-Hall, Englewood Cliffs (1983)
11. Su, C.J., Tassiulas, L.: Broadcast scheduling for information distribution. In: Proceedings of the 17th Annual IEEE Conference on Computer Communications, INFOCOM 1997, Kobe, Japan, pp. 109–117. IEEE Comput. Soc. Press (April 1997)
12. Liaskos, C., Petridou, S., Papadimitriou, G., Nicopolitidis, P., Obaidat, M., Pomportsis, A.: Clustering-driven Wireless Data Broadcasting. IEEE Wireless Comm. Magazine 16, 80–87 (2009)
13. Liaskos, C., Petridou, S., Papadimitriou, G., Nicopolitidis, P., Pomportsis, A.: On the Analytical Performance Optimization of Wireless Data Broadcasting. IEEE Transactions on Vehicular Technology 59, 884–895 (2010)
14. Liaskos, C., Petridou, S., Papadimitriou, G.: Cost-Aware Wireless Data Broadcasting. IEEE Transactions on Broadcasting 56(1), 66–76 (2010)
15. Xu, J., Tang, X., Lee, W.-C.: Time-critical on-demand data broadcast: algorithms, analysis, and performance evaluation. IEEE Transactions on Parallel and Distributed Systems 17(1), 3–14 (2006)

16. Zheng, B., Wu, X., Jin, X., Lee, D.L.: TOSA: a near-optimal scheduling algorithm for multi-channel data broadcast. In: Proceedings of the 6th International Conference on Mobile Data Management (MDM 2005), Ayia Napa, Cyprus, pp. 29–37 (May 2005)
17. Hu, C.-L., Chen, M.-S.: Online Scheduling Sequential Objects with Periodicity for Dynamic Information Dissemination. IEEE Transactions on Knowledge and Data Engineering 21(2), 273–286 (2009)
18. Poor, H.V., Hadjiliadis, O.: Quickest detection. Cambridge University Press, Cambridge (2009), http://www.worldcat.org/oclc/637444316
19. Bansal, N., Coppersmith, D., Sviridenko, M.: Improved Approximation Algorithms for Broadcast Scheduling. SIAM Journal on Computing 38(3), 1157 (2008)
20. Xu, J., Lee, D.-L., Hu, Q., Lee, W.-C.: Data Broadcast. In: Stojmenović, I., Zomaya, A.Y. (eds.) Wiley Series on Parallel and Distributed Computing, pp. 243–265. John Wiley & Sons, Inc., New York (2002)
21. Chen, M.-S., Wu, K.-L., Yu, P.: Optimizing index allocation for sequential data broadcasting in wireless mobile computing. IEEE Transactions on Knowledge and Data Engineering 15(1), 161–173 (2003)
22. Pietronero, L., Tosatti, E., Tosatti, V., Vespignani, A.: Explaining the uneven distribution of numbers in nature: The laws of Benford and Zipf. Physica A: Statistical Mechanics and its Applications 293(1-2), 297–304 (2001)

More for Less

Getting More Clients by Broadcasting Less Data

Christos Liaskos[1], Ageliki Tsioliaridou[2], and Georgios I. Papadimitriou[1,⋆]

[1] Dept. of Informatics, Aristotle University, Thessaloniki 54124, Greece
{cliaskos,gp}@ee.auth.gr
[2] Dept. of Electrical and Computer Engineering, Democritus University, Xanthi 67100, Greece
atsiolia@ee.duth.gr

Abstract. Broadcasting is scalable in terms of served users but not in terms of served data volume. Additionally, waiting time deadlines may be difficult to uphold due to the data clutter, forcing the clients to flee the system. This work proposes a way of selecting subsets of the original data that ensure near-optimal service ratio. The proposed technique relies on the novel *data compatibility distance,* which is introduced herein. Clustering techniques are then used for defining optimal data subsets. Comparison with related work and brute force-derived solutions yielded superior and near-optimal service ratios in all test cases. Thus, it is demonstrated that a system can attract more clients by using just a small portion of the available data pool.

Keywords: periodic broadcasting, service ratio, content selection.

1 Introduction

Data broadcasting is an efficient means of bandwidth preservation and information advertising. As the Internet expands incorporating a steadily increasing number of entities, common needs and preferences begin to appear among the clients. This fact calls for dissemination of information per user class instead of per single user, thus saving network resources by employing broadcasting. However, it is typical of contemporary content providers to attempt coverage of all information topics, in an effort to attract as many clients as possible. Nevertheless, broadcasting does not scale well when the data volume increases [11]. Users experience too long waiting times and flee the system. The present study addresses this issue in the context of wireless, push-based broadcasting.

Push-based broadcasting [1] relies solely on the knowledge of the data popularity distribution, disregarding individual client queries. Its opposite, pull-based process [7], deals with individual, known client queries, serializing then in an optimal manner. In terms of system setup, wireless broadcasting typically assumes

⋆ This research has been co-financed by the European Social Fund - ESF and Greek national funds through the Research Funding Program "Heracleitus II" - NSRF.

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 64–75, 2012.

a single frequency or cellular network, which covers a densely populated area. The clients therein are assumed to be interested in a common set of discrete data classes. Each class is associated with a request probability and an aggregate size of contained data measured in bytes. The class request probabilities may vary with time and any change is detected and handled by smart learning algorithms which are examined elsewhere, as a separate field of study [8, 14]. Once the class probabilities have become known, a central authority creates a broadcast schedule which optimizes a given criterion [2, 17]. The study of [4] proved that a periodic schedule minimizes the mean client waiting time. Periodicity refers to maintaining an approximately constant interval between consecutive broadcasts of each data class. The scheduling problem is then twofold: a) define the optimal number of occurrences of each class in the broadcast schedule and b) serialize data as periodicly as possible [11]. Once the final schedule has been constructed, it is broadcasted repeatedly over the wireless clients in range.

Initially, related studies assumed that a client may wait indefinitely for a wanted data class [4, 16, 17] and formulated the *square root rule*. The rule expresses the optimal number of occurrences for each class, in the form of a ratio of irrational numbers. However, these studies exhibited high computational complexity and subsequent works addressed the issue while still disregarding deadlines [10–12]. An initial attempt to reinstate deadlines took place in [6] where the authors presented a way of minimizing the variance of the clients' waiting time. Modified versions of the algorithms of [17] were demonstrated, that achieved tunable trade-off between the variance and the mean value of the waiting time, without discriminating between variance-sensitive and variance-independent applications. Subsequent studies model the clients' psyche, with a particular interest in impatience [3, 5, 15]. The optimization procedure follows the pattern of [17], typically applying the method of Lagrange multipliers, substituting the waiting time formula with a study-specific impatience metric. The impatience for a specific class is generally a rising function of the waiting time.

The problem with the related approaches is that they aim at improving the quality of the provided service, but not necessarily maximize the number of subscribed users. The issue stems from the fact that no data selectivity mechanism is offered. Even when presented with a huge bulk of obviously nonmatching data, the related approaches will attempt to disseminate all of them in a way that optimizes a given criterion. However, the very fact the volume of data has increased may hinder the upholding of the client deadlines. Furthermore, recent paradigms (e.g. Facebook) have demonstrated that a system becomes prosperous when the number of subscribed users is maximized, not necessarily requiring a top level of quality of service.

Differentiating from the majority of the related works, the present study will pursue to directly maximize the total number of the clients subscribed to the system. Initially, the optimal number of class occurrences inside the broadcast schedule will be defined. Notice that the second part of the scheduling process, i.e. a nearly-optimal serialization procedure, has been already proposed by the authors in [11]. Subsequently, it will be proven that optimality is not always

possible for the complete data set. We will then establish a procedure that can select an optimizable data subset that yields a high number of system clients.

1.1 Notation and Standard Assumptions

We assume a set of data items organized in classes, which is to be disseminated to a group of wireless clients. The operation of the system is push-oriented and subscription-based; the clients do not post queries to a server, but rather listen to the broadcast stream, waiting for updates on classes of information that are of interest to them. However, their attention span is limited; as their waiting time increases, the probability of abandoning the system for another, better service increases as well. These parameters are modeled in the form of the following information class attributes:

- The class index, $i = 1 \ldots N$.
- The popularity, p_i, expressing the probability that a given client request refers to class i.
- The cardinality c_i of class i measured in bytes, which is equal to the total size of individual data items contained therein.

A client may not wait for a wanted item indefinitely. Related studies [5] model the clients' attention span as an exponential probability distribution,

$$P_{abandon}(w) = s \cdot e^{-s \cdot w} \tag{1}$$

which expresses the probability of a client waiting for time w before abandoning a query and, potentially, the system. The attribute s regulates the steepness of the distribution. High values correspond to more demanding clients.

The class popularity distribution p_i and criticality metric s can be inferred from user subscriptions [17], Facebook profile data, simple polling or automated, adaptive schemes [14]. The present study refers to the scheduling process that follows the estimation of these attributes.

The final schedule has a total size of:

$$C = \sum_{i=1}^{N} v_i \cdot c_i \tag{2}$$

while the interval between two consecutive appearances of class i in the schedule is equal to:

$$w_i^{max} = \frac{C}{v_i} \tag{3}$$

where v_i represents the number of occurrences of class i in the schedule. Finally, as in the totality of the cited works, the idealized wireless environment is used as a generic, broadcasting-affinitive context. The scheduling algorithms presented in this study can be applied to any other broadcast-based environment (e.g. wired multicasting). The proposed dissemination scheduling technique is generic

enough to be agnostic of physical layer issues such as ray propagation, coverage, modulation type or encoding. In order to facilitate the presentation of the paper, the reader is encouraged to assume a TV broadcasting scenario, with the data items representing TV shows or classes of shows (e.g. politics, sports, news, etc.).

2 Analysis

Assume that there exist N available data classes with attributes $\{p_i, c_i\}$, $i = 1 \dots N$. The well-known study of [5, Theorem 1] concluded that the class occurrences v_i^{opt} that maximize the client service ratio obey to the rule:

$$\frac{p_i}{c_i}\left[\frac{1}{C \cdot s} - \left(\frac{1}{C \cdot s} + \frac{1}{v_i^{opt}}\right) \cdot e^{-s \cdot \frac{C}{v_i^{opt}}}\right] = const., \; i = 1 \dots N \tag{4}$$

The same study states that eq. (4) cannot be transformed further, and thus an analytic formula for v_i^{opt} is not provided. The authors then provide an algorithmic procedure that seeks to uphold the restrictions of (4) heuristically.

We begin by arguing that the actual problem is not the derivation of an analytic formula. In fact, such an expression is derived in the context of the work. What actually poses a serious issue is that the restrictions (4) cannot hold in most cases: Since v_i parameters express class occurrences in a schedule, they must always take positive integer values. However, equation (4) offers no such guarantee. In order to address this issue, we will be begin by extracting the analytic solution of (4) for v_i^{opt}.

Corollary 1. *The optimal class occurrence ratio, v_i^{opt}, that maximizes the client service ratio follows the relation:*

$$v_i^{opt} \propto \frac{-s}{1 + W\left(\frac{-p_i + V \cdot c_i \cdot s}{e \cdot p_i}\right)} \tag{5}$$

where $W(.)$ is the Lambert-W function and V a constant which is calculated from the expression:

$$\sum_{i=1}^{N} v_i^{opt} \cdot c_i = 1 \tag{6}$$

Proof. Begin by transforming the ratio of (5) into real occurrences by multiplying both parts by the total schedule size, C. The proportionality then becomes equality which is solved for $W(x)$. The Lambert-W function, $W(x)$, represents the solution to $x = W \cdot e^W$, which after trivial calculations leads to the original equation (4). Finally, restriction (6) is derived from (2) when both parts are divided by C.

The class occurrences must be expressed as positive integers. However, the proportionality of equation (5) ensures that a simple rounding can be applied with

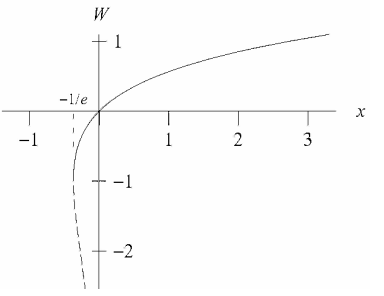


Fig. 1. A graphical illustration of the Lambert-W function, $W(x)$. The restriction $W \leq -1$ corresponds to $^{-1}/_{e} \leq x \leq 0$.

trivial precision loss, provided that the proportionality constant is big enough. Thus, the sole substantial constraint is:

$$v_i^{opt} \geq 0 \iff W\left(\frac{-p_i + V \cdot c_i \cdot s}{e \cdot p_i}\right) \leq -1, i = 1 \ldots N \quad (7)$$

The Lambert-W function, $W(x)$, represents the solutions to $x = W \cdot e^W$ for a given x. It is a multivalued function, which assigns two W values to each $x \in (^{-1}/_{e}, 0]$. For $x = -^{1}/_{e}$ the function is single valued and equals $W(^{-1}/_{e}) = -1$. Figure 1 presents a graphical illustration of the function.

The restriction $W(x) \leq -1$ of equation (7) corresponds to $^{-1}/_{e} \leq x \leq 0$. Notice that the duality of the W-values in $[^{-1}/_{e}, 0]$ does not actually offer two alternatives. Should the upper branch of $W(x)$ be employed, the item broadcast frequencies would be negative, which has no physical meaning. Therefore, for all $i = 1 \ldots N$ it must hold that:

$$^{-1}/_{e} \leq \frac{-p_i + V \cdot c_i \cdot s}{e \cdot p_i} \leq 0 \iff 0 \leq V \leq \frac{p_i}{c_i \cdot s}, \forall i \quad (8)$$

and equivalently:

$$0 \leq V \leq min\left\{\frac{p_i}{c_i \cdot s}\right\} \quad (9)$$

Remark 1. Recall from Corollary 1 that V is a constant which is defined through equation (6). In this context, relation (9) states that the value set of V is limited by the class with the smallest $\frac{p_i}{c_i \cdot s}$ ratio. In other words, if the optimization of equation (4) is unsolvable, then the class with minimal $\frac{p_i}{c_i \cdot s}$ ratio is to blame. In that sense, this class is incompatible with the others in terms of content.
The second condition of solvability stems from relation (6) of Corollary 1. The restriction can be rewritten in the form of the following equation:

$$f(V_o) = \sum_{i=1}^{N} v_i^{opt}(V_o) \cdot c_i - 1 = 0 \quad (10)$$

where $V_o \in [0, min\left\{\frac{p_i}{c_i \cdot s}\right\}]$, in accordance with (9).

Notice from Fig. 1 that the Lambert-W function is monotonous for the studied case of $W(V) < -1$. Therefore, the function

$$v_i^{opt}(V) = \frac{-s}{1 + W(V)} \tag{11}$$

is also monotonous, and so is the summation over all i. Thus, the function $f(V)$ of (10) is monotonous. Consequently, only a single value, $\{V_o | f(V_o) = 0\}$, may exist. Furthermore, the Bolzanno theorem must hold in the range $[0, min\left\{\frac{p_i}{c_i \cdot s}\right\}]$. According to it, the function must undergo a sign change:

$$f(0) \cdot f(min\left\{\frac{p_i}{c_i \cdot s}\right\}) \leq 0 \tag{12}$$

Theorem 1. *A set of information classes with attributes* $\{c_i, p_i\}, i = 1 \ldots N$ *is solvable in terms of maximizing the ratio of served clients, if it holds that:*

$$\sum_{i=1}^{N} \frac{s \cdot c_i}{1 + W\left[\frac{1}{e}\left(\frac{min\left\{\frac{p_i}{c_i \cdot s}\right\}}{\frac{p_i}{c_i \cdot s}} - 1\right)\right]} > -1 \tag{13}$$

Proof. Relation (13) comes as a direct expansion of (12).

Through relation (13), Theorem 1 provides a way of quickly checking the solvability of a data class set. Furthermore, Remark 1 can be used for pinpointing incompatible classes. Thus, an algorithm that detects solvable, optimal subsets of the original classes can be formulated.

2.1 Solvable Data Subsets that Yield Maximum Number of Clients

Let x_i denote the ratio:

$$x_i = \frac{min\{p_i/c_i \cdot s\}}{p_i/c_i \cdot s} \tag{14}$$

Equation (13) is transformed as:

$$\sum_{i=1}^{N} \frac{x_i \cdot p_i}{1 + W\left[\frac{1}{e}(x_i - 1)\right]} > -min\{p_i/c_i \cdot s\} \Longrightarrow$$

$$\left| \sum_{i=1}^{N} \frac{x_i \cdot p_i}{1 + W\left[\frac{1}{e}(x_i - 1)\right]} \right| < |min\{p_i/c_i \cdot s\}| \tag{15}$$

Consider the case of two data classes, $i = 1, 2$. Assume further that $x_2 = 1$, i.e. the second class has the lowest $p_i/c_i \cdot s$ value. From Fig. 1 notice that $W[0] \to -\infty$

in the studied case of $W < -1$. Therefore, in order for the two classes to be *compatible* (i.e. solvable), it must hold that:

$$\left| \frac{x_1 \cdot p_1}{1 + W\left[\frac{1}{e}(x_1 - 1)\right]} \right| < min\left\{ {}^{p_i}/_{c_i \cdot s} \right\} \tag{16}$$

Relation (16) quantifies the compatibility of any two information classes. At first, the class with the minimum ${}^{p_i}/_{c_i \cdot s}$ ratio is detected. For the remaining class, the quantity $\mathcal{S} = \left| \frac{x_1 \cdot p_1}{1 + W\left[\frac{1}{e}(x_1 - 1)\right]} \right|$ is calculated. If $\mathcal{S}$ is smaller than $\mathcal{R} = min\left\{ {}^{p_i}/_{c_i \cdot s} \right\}$, the classes are compatible and may be broadcasted together in a way that maximizes the client service ratio. If not, the classes are incompatible. Therefore, we can define a *solvability distance*, which will measure the compatibility of two given classes:

$$\mathcal{D} = \begin{cases} \frac{S}{\mathcal{R} - S} & , 0 \leq \mathcal{S} < \mathcal{R} \\ \infty & , \mathcal{S} \geq \mathcal{R} \end{cases} \tag{17}$$

The idea is to use the metric of (17) in an iterative clustering approach. Such schemes, such as the classic K-means algorithm [9] typically operate as follows. At first, a predefined number of classes are randomly selected from the original set. These are called cluster centers or *centroids*. Then, the distance from every available class to each centroid is calculated. The classes are then assigned to the centroid that is nearest to them, thus forming groups. At this point new centroids are selected. These are set as the classes nearest to the center of the corresponding formed groups. The process is repeated iteratively, until no change has been made to the groups or to the centroids.

We proceed to formulate a novel procedure that employs this customized version of the k-means algorithm for data clustering. The updated k-means uses the metric of equation (17 - *solvability distance*) for calculating the distance between a centroid and any other class. The update procedure constitutes of selecting the class:

$$I = argmin_{(i)}\{|{}^{p_i}/_{c_i \cdot s} - E\left[{}^{p_i}/_{c_i \cdot s}\right]|\},\ i \in CurrentCluster \tag{18}$$

as the new centroid. $E[.]$ denotes the mean value of a set (.). This modified version is then incorporated to the novel, Data Clustering Algorithm for Higher Service Ratio (DCA-HSR).

The algorithm attempts at first to create as big as possible, solvable clusters of data classes. Thus, the NoC variable, which regulates the number of clusters, is initialized to 1 and is then increased by unary steps. The algorithm checks the solvability of every cluster created at each step. If a cluster is solvable, the algorithm calculates the *coverage* of a cluster as follows:

$$Coverage = \sum_{\forall i \in CurrentCluster} p_i \tag{19}$$

The coverage metric represents the maximum service ratio that the cluster may achieve. Notice that when a class is dropped from the scheduling process, the

Algorithm 1. Data Clustering Algorithm for Higher Service Ratio (DCA-HSR)

INPUT: A set of N CLASSES: $\{p_i, c_i\}$
OUTPUT: The cluster with the maximum service ratio, **best_cluster.**

```
1  best_coverage=0;
2  FOR NoC=1 to N, step 1
3       [CLUSTERS]=ModifiedKmeans(CLASSES, NoC);
4       FOREACH cluster in CLUSTERS
5            IF (cluster is solvable) //eq (13)
6                 coverage=sum(cluster.p_i);
7                 IF (coverage>best_coverage)
8                      best_coverage=coverage;
9                      best_cluster=cluster;
10                ENDIF
11           ENDIF
12      ENDFOR
13 ENDFOR
```

total service ratio is guaranteed to decrease by the corresponding request probability, p_i. The algorithm proceeds to select the solvable cluster that yields the maximum coverage.

Since the selected cluster is guaranteed to be solvable, equation (5) can produce the optimal number of periodic class occurrences in the broadcast schedule, v_i^{opt}. These values, alongside the class cardinality attributes, c_i, are then passed as input to the serializing process of [11], which produces the corresponding periodic broadcast schedule.

3 Simulation Results

In this section we compare the proposed, DCA-HSR scheme with the related approaches of [5] and [15]. The *Service Scheduler* of [5] constitues a classic approach which targets the maximization of the service ratio in a broadcast environment. The *Impatience Scheduler* of [15] on the other hand, follows a contemporary approach of directly minimizing the mean client impatience that a schedule incurs. As already stated, a periodic scheduling process consists of two steps: i) defining the optimal number of occurrences of each class in the schedule, and ii) serializing the data according to the defined occurrences. Each compared study follows its own way of defining the optimal class occurrences. However, all studies follow the generic serialization process of [11] for fairness reasons.

The simulation scenarios consider a varying number of data classes (N), varying class p.d.f. (p_i) skewness and varying data criticality (s). Since the ratio p_i/c_i is already varied through p_i, the class cardinality (c_i) is considered to be equal to one size unit (e.g. *GBytes*), for all i. In each case, the topology consists of a central, broadcast scheduling server and a number of 10^3 tuned-in clients. Firstly, the server produces the optimal number of data class occurrences, v_i^{opt},

according to the employed algorithm. The calculated v_i^{opt} values are fed as input to the serialization process of [11], which has been designed to produce periodic broadcast schedules. The resulting schedule is broadcasted to the clients, fulfilling their demands. The simulation stops when 300 queries per client have been answered or dropped. The waiting time before a client drops a given query is picked randomly from the exponential distribution of (1). After each answer, a client waits for a random *think time* $\in [0, N]$ before posting a new query. The client queries as a whole follow the predefined class p.d.f., p_i. Each simulation is repeated 10^3 times in order to extract a dependable mean value. Concerning the class request probabilities, we employ the ZIPF [13] p.d.f., as in the majority of the cited works. A parameter $\theta \in [0, \infty)$ regulates the skewness of the distribution. The value $\theta = 0$ corresponds to a flat distribution, while higher values indicate an increasingly skewed p.d.f..

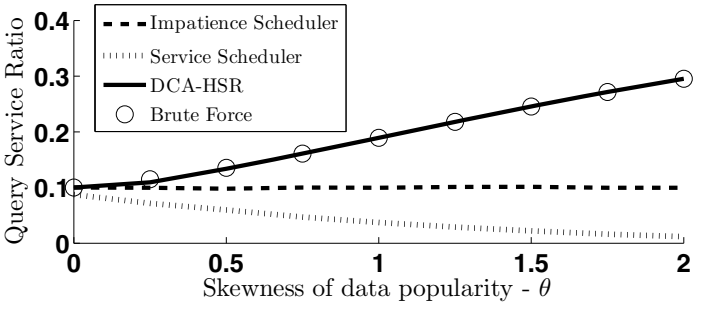


Fig. 2. The behavior of the compared approaches when the commonality in the clients' preferences increases. The proposed DCA-HSR approach focuses on the increasingly popular data classes and achieves near-optimal service ratio in all cases. The Impatience and Service scheduler strive to serve all classes, causing the abandonment of a considerable amount of queries.

Figures 2 and 3 study the case of $N = 5$ available classes. The number is purposefully small, in order to enable comparison with brute force-derived results. Brute forcing constitutes of running the simulation for all possible subsets that can be formed from the original N classes, and keeping the one that achieves the highest service ratio. Notice that there exist 2^N possible subsets in any case. As both figures 2 and 3 illustrate, the proposed DCA-HSR algorithms achieves near-optimal results in all test cases. In detail, Fig. 2 illustrates the performance of the compared approaches when the skewness of the p_i distribution is varied through the θ parameter of the ZIPF formula. The data criticality is kept constant at $s = 2$. Increased θ values correspond to a higher degree of commonality in the clients' demands (i.e. requesting common classes). The proposed DCA-HSR focuses on the common client needs, discarding non-profitable data classes. Thus, it achieves a much higher service ratio than the compared solutions, which strive to disseminate all data classes. When data criticality increases (Fig. 3, $\theta = 1.0$) all solutions suffer from decreased performance, since the client deadlines be-

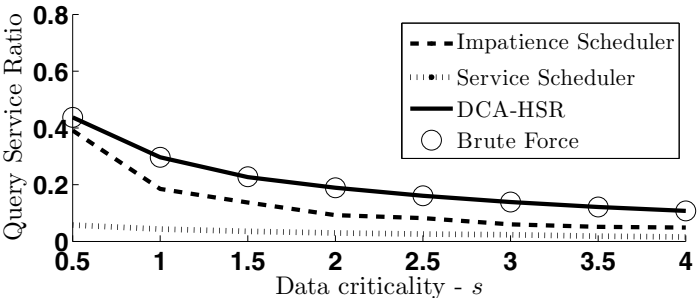


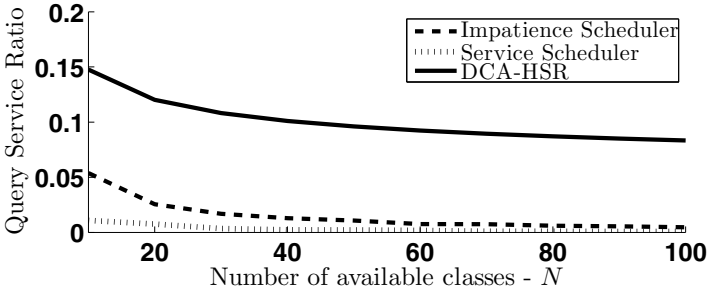
Fig. 3. All compared approaches exhibit lower performance when the data criticality is increased. However, the proposed DCA-HSR approach minimizes the losses by focusing on the most critical data classes.

come more restrictive. However, the proposed DCA-HSR can limit the losses in a nearly-optimal manner, by once again focusing on the most profitable classes.

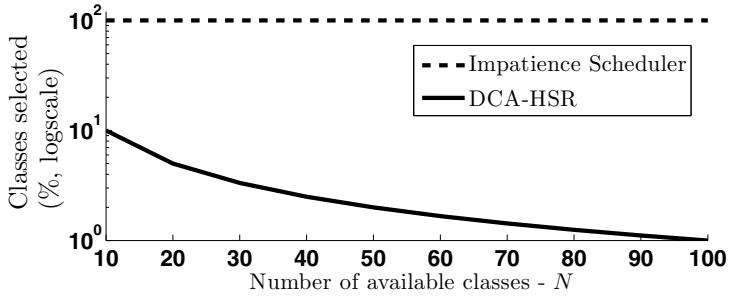
Content selectivity is more evident in Fig. 4 which studies the case of varying the number of available data classes N. The remaining variables are set as $\theta = 1.0$ and $s = 2.0$. Fig. 4a shows the achieved service ratio for each of the compared approaches. Notice that brute forcing is no longer applicable for practical reasons (2^N becomes excessively high). The proposed DCA-HSR algorithm achieves the highest service ratio in all cases $N \in [10, 100]$. What is equally important is the fact that this performance is achieved while using only a small fraction of the original data set (Fig. 4b). Notice that the higher the cardinality of the data set, the higher the aggregate broadcasting cost. Data items must typically be selected and preprocessed by humans in order to achieve sufficient publication quality. This factor aside, the scheduling process itself requires more computational power to serialize the broadcast of the items [11]. Therefore, the proposed DCA-HSR can not only attract more clients, but can actually do so with a much reduced cost.

4 Conclusion and Future Work

The present study proposed the Data Clustering Algorithm for Higher Service Ratio (DCA-HSR) which can choose the subset of the original data that yields near-optimal service ratio. Comparison with related studies showed that the DCA-HSR can attract several times more clients, while employing a small fraction of the available data. This fact can also be used for cutting down on the generic expenses associated with the broadcast process. Future work is directed towards broadcasting the secondary data sets produced by the DCA-HSR in a multichannel scheme, increasing the service ratio further.



(a) Service ratios achieved by the compared approaches. By focusing on promising data subsets only, the proposed DCA-HSR algorithm achieves three to ten times greater performance than the related schemes.



(b) Ratio of selected classes for broadcasting, corresponding to Fig 4a. The Impatience Scheduler and the Service Scheduler attempt to disseminate all available data classes in any case. On the other hand, the proposed DCA-HSR algorithm selects a nearly-optimal subset of the classes for broadcasting.

Fig. 4. The proposed DCA-HSR not only achieves much greater service ratio, but employs a minimal subset of the available data as well. Thus, DCA-HSR achieves both increased performance and minimal scheduling cost (pre-processing data for publication, data serialization [11]).

References

1. Acharya, S., Alonso, R., Franklin, M., Zdonik, S.: Broadcast disks. ACM SIGMOD Record 24(2), 199–210 (1995)
2. Balli, U., Wu, H., Ravindran, B., Anderson, J., Jensen, E.: Utility Accrual Real-Time Scheduling under Variable Cost Functions. IEEE Transactions on Computers 56(3), 385–401 (2007)
3. Chen, J., Lee, V.C.S., Zhan, C.: Efficient Processing of Real-Time Multi-item Requests with Network Coding in On-demand Broadcast Environments. In: Proceedings of the 15th Int. Conf. on Embedded and Real-Time Computing Systems and Applications (RTCSA 2009), Beijing, China (August 2009)
4. Gecsei, J.: The Architecture of Videotex Systems. Prentice-Hall, Englewood Cliffs (1983)
5. Jiang, S., Vaidya, N.H.: Scheduling data broadcast to "impatient" users. In: Proceedings of the 1st ACM International Workshop on Data Engineering for Wireless and Mobile Access (MoBiDe 1999), Seattle, Washington, United States, pp. 52–59. ACM (1999)
6. Jiang, S., Vaidya, N.H.: Response time in data broadcast systems: mean, variance and tradeoff. Mobile Networks and Applications 7(1), 37–47 (2002)
7. Xu, J., Tang, X., Lee, W.-C.: Time-critical on-demand data broadcast: algorithms, analysis, and performance evaluation. IEEE Transactions on Parallel and Distributed Systems 17(1), 3–14 (2006)
8. Kakali, V.L., Sarigiannidis, P.G., Papadimitriou, G.I., Pomportsis, A.S.: A Novel Adaptive Framework for Wireless Push Systems Based on Distributed Learning Automata. Wireless Personal Communications 57(4), 591–606 (2011)
9. Kanungo, T., Mount, D.M., Netanyahu, N.S., Piatko, C.D., Silverman, R., Wu, A.Y.: An efficient k-means clustering algorithm: analysis and implementation. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(7), 881–892 (2002)
10. Liaskos, C., Petridou, S., Papadimitriou, G.: Cost-Aware Wireless Data Broadcasting. IEEE Transactions on Broadcasting 56(1), 66–76 (2010)
11. Liaskos, C., Petridou, S., Papadimitriou, G.: Towards Realizable, Low-Cost Broadcast Systems for Dynamic Environments. IEEE/ACM Transactions on Networking 19(2), 383–392 (2011)
12. Liaskos, C., Petridou, S., Papadimitriou, G., Nicopolitidis, P., Pomportsis, A.: On the Analytical Performance Optimization of Wireless Data Broadcasting. IEEE Transactions on Vehicular Technology 59, 884–895 (2010)
13. Pietronero, L., Tosatti, E., Tosatti, V., Vespignani, A.: Explaining the uneven distribution of numbers in nature: The laws of Benford and Zipf. Physica A: Statistical Mechanics and its Applications 293(1-2), 297–304 (2001)
14. Vincent Poor, H., Hadjiliadis, O.: Quickest detection. Cambridge University Press, Cambridge (2009)
15. Raissi-Dehkordi, M., Baras, J.S.: Broadcast Scheduling for Time-Constrained Information Delivery. In: Proceedings of the 2007 IEEE Global Telecommunications Conference (GLOBECOM 2007), Washington, DC, USA, pp. 5298–5303 (November 2007)
16. Su, C.-J., Tassiulas, L., Tsotras, V.J.: Broadcast scheduling for information distribution. Wireless Networks Journal 5(2), 137–147 (1999)
17. Vaidya, N.H., Hameed, S.: Scheduling data broadcast in asymmetric communication environments. Wireless Networks 5(3), 171–182 (1999)

A Method to Improve the Channel Availability of IPTV Systems with Users Zapping Channels Sequentially

Junyu Lai and Bernd E. Wolfinger

Department of Computer Science, University of Hamburg, Hamburg, Germany
{lai,wolfinger}@informatik.uni-hamburg.de

Abstract. The crucial task leading IPTV services to be successful is to ensure users' quality of experience (QoE) better than, or at least comparable to the traditional cable/satellite TV. Among QoE measures, TV channel availability is of extreme importance. In this paper, we investigate how to improve the channel availability in IPTV systems with users zapping channels sequentially. Firstly, we present the negative influence on the channel availability, introduced by users' sequential zapping behavior. Then, an intentional Switching Delay (iSD) admission control method, in combination with a 2-layered (base/enhancement layer) scalable video coding (SVC) scheme, is proposed to mitigate the impact. Thereafter, comprehensive simulation experiments demonstrate the potential of the iSD method, to significantly improve the channel availability, with only slightly degrading the average service delay of channel enhancement layers. Finally, our recently proposed TV channel access control (TCAC) scheme is integrated, aiming to further enhance the iSD method's performance.

Keywords: IPTV, Admission control, Traffic characterization/modeling, Channel availability, Call blocking probability (CBP).

1 Introduction

The current trend of network convergence has been best exemplified by the recent emergence of new services such as *Internet Protocol Television* (IPTV). Typically, IP multicast technology is employed to deliver the TV channel streams. Each individual TV channel is mapped onto a dynamic IP multicasting stream with a unique multicast address. Fig. 1 presents a typical xDSL based IPTV delivery network architecture, where users register IPTV, IP phone and Internet access services. A network device called *Digital Subscriber Line Access Multiplexer* (DSLAM) aggregates traffic of hundreds of *set-top boxes* (STBs) in user premises and connects to a *Broadband Remote Access Server* (BRAS) which again aggregates traffic of its downlink DSLAMs to the high speed metropolitan backbone (i.e. to the Edge-routers). The *Head-end Office* directly connects to the backbone by a high speed link, and streams all provided TV channels towards Edge Routers by means of unicasting or multicasting.

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 76–89, 2012.

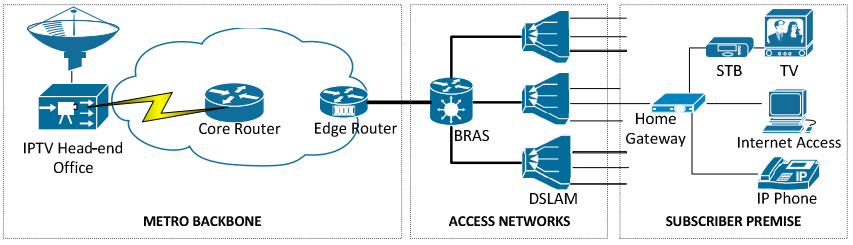


Fig. 1. Typical xDSL based network architecture for IPTV service delivery

When a user switches to a specific TV channel, the STB will send the request to join the corresponding multicast stream. Then, if successful, the user can receive and start to watch the requested channel. In practice, the service providers are likely to provide a large amount of diverse TV channels (to attract more users), whereas the bandwidth reservations (for IPTV services) on links of the access networks (between Edge Routers and the Access Points, e.g. DSLAMs) are usually limited, and there is no guarantee for all the user requests to be fulfilled at an arbitrary instant. We define a link, which is incapable to simultaneously transmit all provided TV channels, as a *Potential Bottleneck link* (PB-Link). Traditionally, *Call Blocking Probability* (CBP) is employed to denote the ratio of blocked user requests. A high CBP can dramatically decrease the *channel availability* (CA), and hence can significantly degrade users' watching experience (i.e. *Quality of Experience* [1], QoE)

In our previous work, comprehensive studies on topics such as IPTV CBP evaluation [2], CBP reduction [2-4], and user behavior modeling [5] have been carried out in both stationary state and peak-hour scenarios. One of the most important observations is that, IPTV users zapping channels sequentially can introduce strongly negative impacts onto the channel availability (i.e. CBP) in a bandwidth resource limited IPTV system. This was rarely presented in the previous literature. In this paper, for the sake of successful IPTV service deployments, we are motivated to investigate solutions to facilitate the aforementioned negative impacts, and hence to improve the channel availabilities. More specifically, the contribution of this paper includes:

- Demonstration of the significantly bad influence on the IPTV channel availability brought by users zapping channels sequentially.
- Proposal of an *intentional Switching Delay* (iSD) admission control method, combined with a 2-layered (i.e. base and enhancement layer) *Scalable Video Coding* (SVC) [6] scheme. According to comprehensive simulation experiments, the iSD method can efficiently reduce the CBP and improve channel availability, with only slightly increasing the average delay of the enhancement layer transmissions.
- Further enhancement of the iSD method's performance by integrating our recently proposed *TV channel access control* (TCAC) [2-4] scheme.

The rest of the paper is organized as follows. Section 2 introduces the related work. Then, the negative impact due to sequential zapping behavior is illustrated in Section 3. After that, an iSD admission control method is proposed in Section 4, including a

discussion on its pros and cons. In Section 5, the performance of the iSD method is evaluated by means of simulation. Moreover, Section 6 explores the possibility to integrate a recently proposed TCAC scheme with the iSD method to further improve the channel availability. Finally, conclusions and outlook are given in Section 7.

2 Related Work

IPTV channel availability is directly related to user CBP. There are plenty of research works in the area of IPTV CBP evaluation. Earlier publications have already provided well-known exact or approximate analytical algorithms for the CBP calculation in the context of unicasting, in static as well as dynamic multicasting scenarios [7-13]. In addition, simulation based evaluation is also an option. In [2] J. Lai et al. proposed a link state-vector-based model to simulate IPTV systems in detail. Besides, a TCAC scheme [2][3] was also proposed to decrease the CBP and to improve the channel availability. However, all the above efforts focused on the long-term stationary scenarios of the IPTV system, but the behavior of these stationary models could be far away from reality.Hence, J. Lai et al. [4][5] investigated the CBP evaluation in a more realistic peak-hour scenario. The extended TCAC scheme was applied in the peak-hour scenarios, and it was shown that it still can efficiently reduce the CBP [4].

As opposed to conventional TV systems, IPTV users can actively influence the traffic load of the delivery networks. Therefore, some studies [14-17] on the traffic characterization and modeling of IPTV user behavior have been conducted. In [14] T. Qiu et al. studied a large national IPTV system and developed a series of analytical models for different aspects of user activities. F. Ramos et al. [15] modeled the behavior of a typical user as a rapid burst of channel selection events followed by an extended period of channel viewing. G. Yu et al. [16] investigated IPTV user activities in terms of zapping rate and session lengths. M. Cha et al. [17] studied how users select channels in the real world based on a comprehensive trace from a commercial IPTV service, including channel popularity dynamics, aggregate viewing sessions and content locality. In our previous work [5] (which was an extension of earlier IPTV user behavior models), we modeled the channel switching behavior of a single IPTV user. Both, channel popularity and user activities have been taken into account. Our proposed IPTV user behavior automaton (IPTV-UBA) model relies on formal descriptions to capture the characteristics of user behaviors. The aggregate user behavior can be derived by overlaying the behaviors of a set of users.

Scalable video coding has already been included in video coding standards, such as H.262, MPEG-2 Video, H.263, and MPEG-4 Visual [6]. However, the past standardization efforts produced results with low coding efficiency and high decoder complexity, until the SVC extension of H.264/AVC [18] was proposed. Many research efforts [19] have been carried out in the field of delivery of IPTV channels using SVC profiles in wired/wireless networks, due to its flexible adaptation of video quality under bandwidth fluctuation and device capacity variation.

3 Impact of Sequential Zapping on IPTV Channel Availability

3.1 Assumptions and Notations

In this paper, we focus on a single link *L* in a typical xDSL based IPTV delivery system. This link is assumed to be located in the access networks (between Edge Routers and DSLAMs, cf. Fig. 1), and its bandwidth reservation for the IPTV service cannot support the concurrent transmission of all the provided TV channels. Therefore, blocking events may happen when users (downstream of link *L*) request a channel which is not yet available on link *L*. Accordingly, link *L* is a PB-link. We also assume that the other links on the channel delivery path are not PB-links. Below are other general assumptions that studies in this section will be based on:

- There are *N* TV channels provided in total. All the channel streams are *constant bit rate* (CBR) and *single-layer* encoded. The multicasting transmission of each channel consumes *C* bandwidth capacity units on each link of the delivery path.
- The bandwidth reserved for the IPTV service on the PB-link *L* is *BW* units, which is smaller than *N*C* units, the minimum bandwidth capacity required to concurrently transmit all the provided TV channels.
- Only the peak-hour scenarios are investigated. Peak-hour is defined as a period among 24 hours of a typical day, when the number of the active IPTV users reaches its peak, and meanwhile, the number is assumed to be relatively stable. In this paper, without loss of generality, we assume peak-hour is from 9 pm to10 pm.
- During peak-hour, the considered single link system is assumed to be a *closed system,* namely the number of users (downstream of the PB-link *L*) watching TV is constant, i.e., no user will enter into or leave from this system. The number of users in the system is defined as the *offered traffic* (denoted as OT).
- For each single user in the closed system, an IPTV-UBA (developed in [5], also cf. Section 3.2) is used for generating the traffic traces (i.e. the channel requests and dwell time sequence). The *aggregate trace* of the system during the peak-hour is obtained by means of simply overlaying the traces of all the users.
- As mentioned, call blocking events may happen due to insufficient bandwidth left on link *L*. Therefore, a user may possess two different states, namely *Watching State* and *Blocked State*. The former state refers that a user has tuned into a TV channel (in viewing or zapping mode), while the latter denotes the duration from the moment when a user's request is blocked to the instant when the same user has successfully requested another channel for the first time after that blocking.

3.2 IPTV User Behavior Modeling

In IPTV systems, characterization and modeling user behaviors are essential for many design and engineering tasks such as evaluation of various design options, optimal system parameter tuning and network capacity planning. Therefore, we recently

investigated the formal descriptions of IPTV user activities, and have proposed an IPTV-UBA user behavior model [5]. IPTV-UBA can produce, in a realistic manner, the trace of channel switching events of a typical IPTV user during its active sessions. IPTV-UBA categorizes the channel switching events into two different modes: *viewing mode* and *zapping mode*. Fig. 2 illustrates such an example. As can be seen, channel switching events performed by a typical user during the peak-hour are plotted against the time. Viewing mode (marked as V) refers to a user viewing a specific TV channel for a long period of time, namely several to tens of minutes. This implies that the currently broadcasting program on the channel is the user's preference. Zapping mode (marked as Z) denotes a user switching channels quickly in a short time, with the purpose to find something of interest. On the other hand, depending on the next channel chosen, channel switching events can also be considered as *sequential* or *targeted*. In the sequential channel switching, a user chooses the next channel by UP and DOWN buttons on the remote control, while the targeted switching represents the case that a user chooses the desired channel directly by pressing the numeral buttons or by using *Electronic Program Guide* (EPG). Additionally, IPTV-UBA assumes:

- In the viewing mode, channels are targetedly selected according to their popularities, which can be obtained from a probability distribution (e.g. Zipf distribution) or derived from empirical traces. In this paper, we employ a class based channel popularity distribution derived from a practical measurement [17] (cf. Table 1).
- In the zapping mode, channels can be chosen either sequentially or targetedly with different probabilities. Therefore, the zapping mode can be further divided into two sub-modes, namely *sequential zapping mode* and *targeted zapping mode*. The sequential zapping mode is defined as users' fast surfing TV channels by pressing UP and DOWN buttons of their remote controls.

In this paper we emphasize on the sequential zapping mode, due to its significant influence on the channel availability (cf. Section 3.4). The reader can refer to our previous work [5] for more information about the IPTV-UBA model.

3.3 Simulation Model to Investigate the Channel Availability

Channel availability (denoted as P_{CA}) on a link in an IPTV system is directly related to the user CBP (denoted as P_{CBP}), and hence can be expressed by $P_{CA} = (1- P_{CBP})$. With the aim to evaluate the CBP and the channel availability of the IPTV system in the peak-hour scenarios, we have developed a *Monte Carlo* simulation model, based on the *n-dimensional* state vector below:
(userNumChannel_1, userNumChannel_2, … , userNumChannel_n).
Here *n* is the total number of provided IPTV channels. This vector is used for recording the current status of the PB-link *L*, i.e., the *i*th element of the vector represents the current number of users watching channel *i* through *L*. Below is the simulation logic:
Step_1. Produce an aggregate trace of channel switching events for the peak-hour scenario of the studied IPTV system, by means of our IPTV-UBA model.

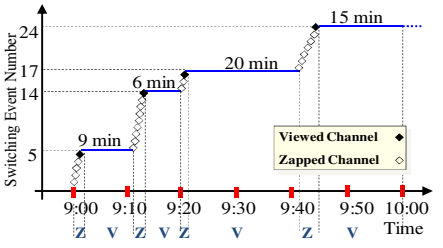


Fig. 2. Channel switching sequences

Table 1. Channel popularity distribution

Class	Num. of Ch. in Each Class	Overall Watching Prob. of Each Class	Individual Watching Prob. for Each Channel
A	7	63.9314%	9.1331%
B	13	21.5044%	1.6542%
C	20	9.9574%	0.4979%
D	40	4.2496%	0.1062%
E	40	0.3572%	0.0089%

Step_2. Determine the initial system state. At the beginning of the peak-hour, we need to determine what the user states would be like, i.e., the user being in a waiting state, or in a watching state. If a user is in a watching state, the channel he is currently watching also needs to be assigned. In this paper, we employ an efficient method to generate the initial system state. The principle of the method is to randomly assign a channel i to a single user according to the channel popularity distribution. If channel i is not yet successfully requested by another user, and the remaining bandwidth capacity of PB-link L is not sufficient to support its transmission, the user will be in a waiting state. Otherwise he will be in a watching state, tuned to channel i. This procedure will be repeated, until all the users have been assigned a specific channel or the waiting state. At last, the element values of the state vector are adjusted accordingly.

Step_3. Start to simulate user request arrival events and user departure events according to the aggregate trace (generated in Step_1), for the peak-hour duration. Because of the dynamic user behavior and the bandwidth constraint, we may need to adjust the element values of the state vector at the instants when arrival events or departure events happen. The adjustments are used for simulating the link status transitions. On the other hand, we will not have to change the value of the state vector at the instant when a call blocking event happens (due to insufficient bandwidth left).

Step_4. Calculate the CBP and the channel availability, based on the number of call blocking events and the total user requests recorded during the peak-hour duration.

It is worth noting that the main advantage of this simulation model as compared with other analytical models (e.g. [10] and [13]) lies in its flexibility, i.e., it can be used for simulating more general and complicated user behaviors (e.g. more realistic user request arrival processes, like the one generated by our IPTV-UBA model). Moreover, the simulation model is still applicable in situations when our proposed iSD user request admission control method is used.

3.4 Important Observation from Simulation Experiments

Based on the above simulation model, a comparative study has been conducted for two different scenarios, where the IPTV users' sequential zapping mode is enabled and disabled, respectively. The simulation parameter values are the same for both scenarios, and have been summarized in Table 2. The simulation results are plotted in Figs. 3 and 4. More specifically, Fig. 3 presents the CBPs against the offered traffic for both scenarios, and Fig. 4 demonstrates channel availabilities against the offered traffic for both scenarios. As can be seen, users' sequential zapping behavior can increase the CBP more than 10 times in high-load situations (cf. Fig. 3), and accordingly degrade the channel availability significantly (cf. Fig. 4). However, this observation and counter-measures for this service degradation have rarely been studied in the past. Therefore, since sequential channel zapping is very common in a practical IPTV system, we are strongly motivated to find a method to combat this negative impact introduced by users zapping channels sequentially.

Table 2. Parameter values for the simulations in Section 3

Notations	Descriptions	Values
N	The number of all the provided TV channels	120
C	The single-layered video stream bit rate of each TV channel	8Mbps (e.g. HD format)
B	The bandwidth reservation for IPTV service on the PB-link L	640Mbps (80 TV channels)
OT	Offered traffic (during the peak-hour)	From 200 to 7000
p_i	The channel popularity distribution for targeted channel switching	The distr. in Table 1.

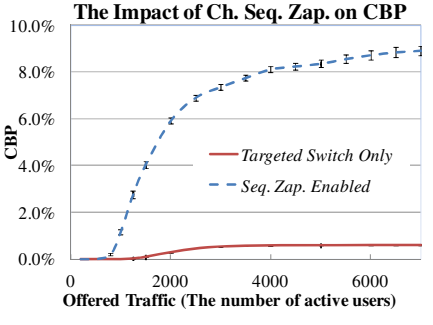


Fig. 3. Sequential zapping's impact on CBP

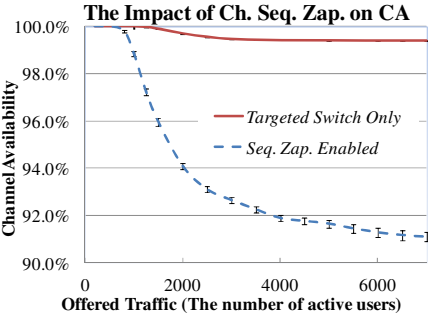


Fig. 4. Sequential zapping's impact on CA

4 A Method to Improve Channel Availability in Case of Sequential Zapping

4.1 Analysis of Sequential Channel Zapping

Users may have two different purposes, when conducting the sequential zapping. On one hand, a user may want to switch to a specific destination channel *i* for viewing. Therefore, except channel *i*, all the other channels during the sequential zapping are not the user's interest. On the other hand, the user may not yet have determined the preferable channel to view. The purpose of conducting the sequential zapping is to search for an interesting TV program. Having switched to a channel where the currently broadcasting TV program is the user's interest, the user will stop sequential zapping and start viewing this channel. On that basis, user' sequential zapping behavior is categorized into two different types, namely SZ_I and SZ_II. Note that, for users of SZ_II, the decision on whether or not to continue the sequential zapping will be based on the broadcasted program of the current channel. However, for users of SZ_I, due to the fact that their desired channels are determined in advance, the program on the current channel has no influence on users' sequential zapping.

4.2 Principle of the iSD Admission Control Method

In order to alleviate the negative impact due to sequential zapping, we have proposed an *intentional Switching Delay* (iSD) admission control method, in combination with a 2 layered (base/enhancement layer) SVC profile. Each TV channel is encoded as a base layer stream (for the basic quality) and an enhancement layer stream (for a better quality). The enhancement layer stream can be decoded only with the corresponding base layer. In practice, the SVC profile can be implemented in a "base layer first, full quality thereafter" manner. (A picture-in-picture preview could be an alternative). The principle of the iSD method lies in two aspects. Firstly, we always reserve enough bandwidth to dynamically multicast the base layers of all the channels, which ensures that the users can acquire the program of a channel immediately after switching to it. Secondly, the system does not deliver the enhancement layer of a requested channel at once in case of sequential zapping. Instead, the service of the enhancement layer will be intentionally delayed for a period of time ΔT. The incentive to do this is demonstrated in Fig. 5. If the interval between the current channel request R_3 (generated by a user's sequential zapping) and R_4 the subsequent request of the same user, is smaller than ΔT, then, the enhancement layer of the channel demanded by the first request does not have to be served. Therefore, it is possible to reduce the CBP of the enhancement layer. The pseudo code of the iSD method is given in Fig. 6.

Since the iSD admission control method always reserves enough bandwidth for the transmission of the base layers of all the TV channels, the CBP of the base layer is 0. Therefore, in the following sections, we focus on the CBP and channel availability of the enhancement layers, denoted as CBP_{en} and CA_{en}, respectively.

4.3 Advantages and Disadvantages of the iSD Method

There are advantages by applying the iSD method. Firstly, there are no call blocking events for the base layers of all the provided TV channels. Secondly, we could have a chance to further decrease the channel switching delay, due to the lower bit rate of the base layers (compared to a single-layer video coding profile). Thirdly, the most important gain is its potential to decrease CBP_{en} (the CBP of the enhancement layer) and accordingly to improve CA_{en} (the channel enhancement layer availability). On the other hand, the disadvantage in evidence, is the additional service delay ΔT of the enhancement layers introduced by the iSD method. We can expect that, as ΔT gets larger, CBP_{en} decreases, CA_{en} increases, and the average service delay of the enhancement layers also increases. However, we note that not all the user requests should be delayed by the iSD method. Only requests generated by users in the sequential zapping mode, and moreover, only those channels the enhancement layer of which is not yet transmitted, will be delayed, according to the iSD method.

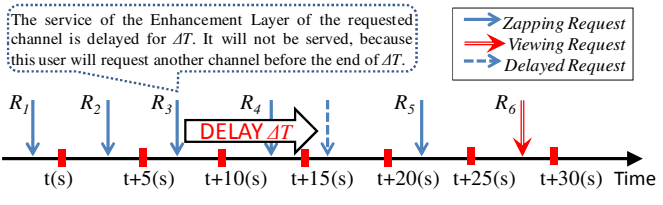


Fig. 5. Illustration of the incentive for the iSD user request admission control method

```
if (A channel request R is generated by a user's sequential zapping at T0){
        if (the En. Layer of the requested channel is not yet transmitted){
                The Base Layer of the requested channel will be served immediately;
                The service of the En. Layer of the requested channel by R is intentionally DELAYED for ΔT;
                if (the arrival of the next channel request from the same user happens in [T0, T0+ΔT])
                        The service of the En. Layer will be ignored by the system;
                else
                        The service of the En. Layer will be started at T0+ΔT;
                }
        else
                ACCEPT and serve both layers of the requested channels immediately;
        }
else
        NORMAL PROCEDURE;
```

Fig. 6. The pseudo code of the iSD user request admission control method

Table 3. Parameter values for the simulation experiments in Section 5

Notations	Descriptions	Values
N	The number of all the provided TV channels	120
C_{ba}	The base layer bit rate of each TV channel	2Mbps (e.g. SD format)
C_{en}	The enhancement layer bit rate of each TV channel	6Mbps (e.g. enhance for HD)
B	The bandwidth reservation for IPTV service on the PB-link L	720Mbps (240Mbps for the base layers)
ΔT	The intentional delay introduced by the iSD method	1, 2, 3, 4, 5, 7, 9, 12, 15, 20, 25, 30, 40, 50, 60 (s)
OT	Offered traffic (during the peak-hour)	From 200 to 7000
p_i	The channel popularity distr. for targeted channel switching	The distribution in Table 1.

5 Performance Evaluation by Means of Simulation

5.1 Simulation Introduction

To evaluate the performance of the iSD method, we integrated it into the simulation model proposed in Section 3.3. Assumptions for the studies in this section are consistent with those in Section 3, except that each TV channel is now encoded in a 2-layered profile, instead of a single-layered profile. We assume further that the two encoding profiles have the same coding efficiency, and the bandwidth reservation on the PB-link L for the IPTV service is always sufficient to simultaneously transmit the base layers of all the provided channels. Comprehensive simulation experiments have been carried out with the parameter values presented in Table 3, and the simulation scenario is demonstrated in Fig. 7. During the simulations, we mainly focus on four different measures, namely CBP_{en} (CBP of the enhancement layers), CA_{en} (channel enhancement layer availability), R_{suc} (the ratio of user requests for which the iSD method has been successfully applied), and D_{avg} (the average delay of serving the channel enhancement layers, introduced by the iSD method).

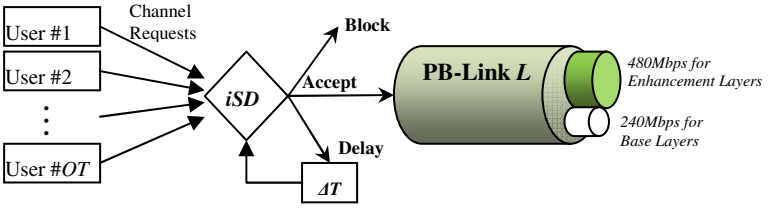


Fig. 7. Simulation scenarios demonstration

5.2 Simulation Results

Experimental results are plotted in Figs. 8-12 (in which the confidence intervals given are based on a confidence level of 95%). More specifically, Fig. 8 presents CBP_{en} against the offered traffic for different values of ΔT. In Fig. 9, the same data set is also plotted against the intentional Switch Delay ΔT, for different values of the offered traffic. As can be seen, CBP_{en} increases as the offered traffic grows, and decreases if ΔT increases. Fig. 10 illustrates the CA_{en} against ΔT. As can be expected, channel enhancement layer availability increases as ΔT grows. We also noticed from Figs. 9 and 10, when ΔT is getting larger and beyond a threshold, the gain (CBP_{en} reduction and CA_{en} increase) brought by further increasing the value of ΔT will not be obvious. This implies that the service provider can find a value for ΔT which, according to his service provisioning policies, is close to optimum (e.g. ΔT=20s). In Fig. 11, R_{suc} against ΔT are plotted for different values of offered traffic. We observe that R_{suc} increases as ΔT grows, and decreases as the offered traffic increases. Last but not least, Fig. 12 demonstrates D_{avg} against ΔT for different values of offered traffic. As is shown, only slight increase of D_{avg} can be observed, even at the maximum (i.e. about 3 seconds at OT=200, ΔT=60). The simulation experiments imply that our iSD admission control method can mitigate the impact due to users zapping channels sequentially, and significantly improve the channel (enhancement layer) availability.

6 Combination with a Recently Proposed TCAC Scheme

A recently proposed *TV channel access control* (TCAC) scheme [2-4] has demonstrated its potential to efficiently decrease the CBP, and accordingly to increase the channel availability of IPTV systems. In this section, we combine the TCAC scheme with the iSD admission control method, with the purpose to further enhance the performance of the iSD method. The basic idea of the TCAC scheme is that, at instants when remaining bandwidth on the PB-link L becomes scarce, it could be a good idea to deny the service of the enhancement layer of a requested low priority (unpopular) channel whose enhancement layer is not yet transmitted. For more details about the TCAC scheme, the reader can refer to [2-4]. The principle of the combination of the iSD method with the TCAC scheme can be depicted as: *Applying the iSD method for user requests generated by sequential channel zapping, while using the TCAC scheme for the user requests generated by targeted channel switching*. Simulation experiments have been conducted to investigate whether further improvement can be obtained by this combination. Taking the same assumptions as in Section 5, a series of scenarios with different values of ΔT have been simulated. The relative CBP_{en} changes (against the *OT*) in four typical scenarios are plotted in Fig. 13. As can been observed, when ΔT is small (ΔT=3s), combining TCAC scheme cannot further decrease CBP_{en}. As ΔT is getting larger (ΔT=25s), the TCAC scheme starts to bring positive gain. When ΔT continues to grow (ΔT=50s or 60s), combining the TCAC can bring significant decrease of CBP_{en} (up to 18%), and accordingly improve the CA_{en}. The possible reasons to explain this may consist in two aspects. On one hand, the TCAC scheme was developed based on the assumption of targeted channel switching only. Therefore, it cannot handle a scenario which contains a large number of

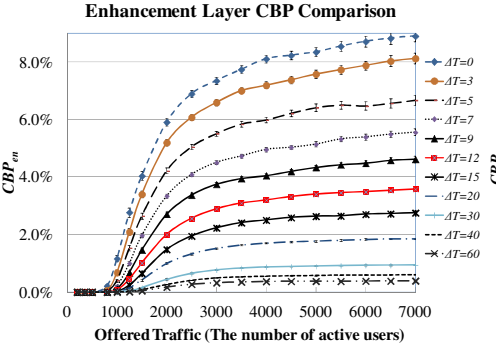


Fig. 8. CBP_{en} against Offered Traffic

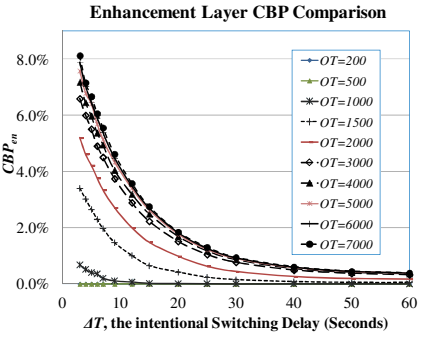


Fig. 9. CBP_{en} against ΔT

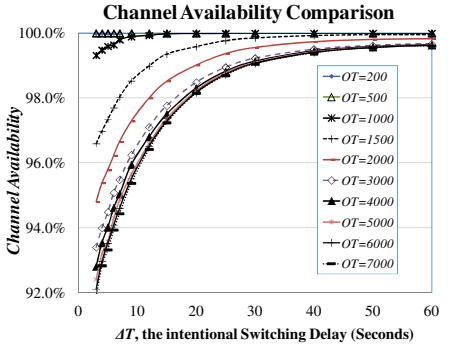


Fig. 10. Channel availability against ΔT

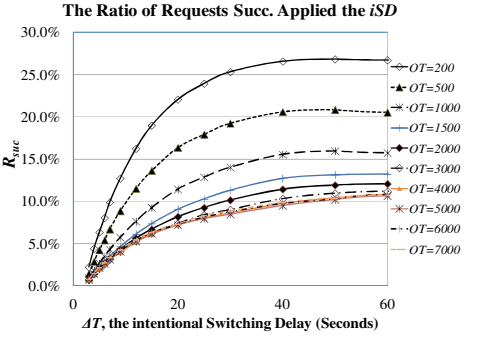


Fig. 11. R_{suc} against ΔT

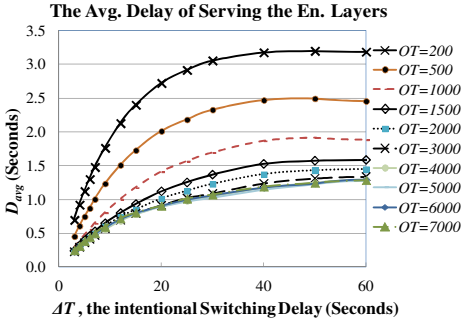


Fig. 12. D_{avg} against ΔT

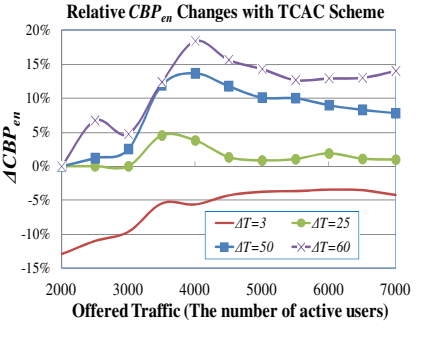


Fig. 13. Relative CBP_{en} changes against

sequential channel switching events. On the other hand, a longer delay introduced by the iSD admission control method (i.e. a large value of ΔT), can alleviate more sequential zapping requests (cf. Fig 11), and can therefore produce a scenario which is

more favorable for the TCAC scheme. Based on the above observations, we conclude that the combination of TCAC scheme with the iSD method can further increase the channel availability of the enhancement layers efficiently, when ΔT is large enough.

7 Summary and Outlook

In this paper, we focused on mitigating the negative impact brought by users zapping channels sequentially, in a bandwidth resource limited IPTV system. This impact and its solutions have been very rarely studied in the previous literature. Firstly, we demonstrated that, users' sequential zapping behavior can "kill" a bandwidth limited IPTV system, by significantly increasing the CBP and degrade the channel availability. In order to restrict this impact, we proposed an intentional Switching Delay (iSD) user request admission control method, combined with a 2-layered SVC profile. Comprehensive simulation experiments have illustrated the ability of the iSD method to significantly decrease the CBP of the enhancement layer (up to 90%, relatively), and hence to improve IPTV channel (enhancement layer) availability, with only slightly increasing the average service delay of the channel enhancement layers. Finally, a recently proposed TCAC scheme has been combined with the iSD method, with the potential to further enhance the channel (enhancement layer) availability.

Our planned work, in the future, will be the investigation of the iSD method with an n-layered (n>2) SVC profile. On the other hand, the application of the iSD method in an entire IPTV delivery network with a tree topology and not just for a single bottleneck link will also be of interest to us.

References

1. Kerpez, K., Waring, D., Lapiotis, G., Lyles, J.B., Vaidyanathan, R.: IPTV Service Assurance. IEEE Communications Magazine 44(9), 166–172 (2006)
2. Lai, J., Wolfinger, B.E., Heckmüller, S.: Decreasing Call Blocking Probability of Broadband TV Services by a Channel Access Control Scheme. In: International Conference on Ultra Modern Telecommunications and Control Systems, Moscow, Russia (2010)
3. Lai, J., Wolfinger, B.E., Heckmüller, S.: Decreasing Call Blocking Probability of Broadband TV Services in Networks with Tree Topology. In: SPECTS 2011, The Hague, Netherlands (2011)
4. Lai, J., Wolfinger, B.E., Heckmüller, S.: Decreasing the Call Blocking Probability of Broadband IPTV Services in Stationary and Peak-hour Scenarios. Journal of Networks (submitted)
5. Abdollahpouri, A., Wolfinger, B.E., Lai, J., Vinti, C.: Elaboration and Formal Description of IPTV User Models and Their Application to IPTV System Analysis. In: MMBnet 2011, GI/ITG Workshop, Hamburg, Germany (2011)
6. Schwarz, H., Marpe, D., Wiegand, T.: Overview of the Scalable Video Coding Extension of the H.264/AVC Standard. IEEE Transactions on Circuits and Systems for Video Technology 17(9) (2007)

7. Choudhury, G.L., Leung, K.K., Whitt, W.: An Algorithm to Compute Blocking Probabilities in Multi-rate Multi-class Multi-resource Loss Models. In: 5th IEEE Internat. Workshop on Computer-Aided Mod., Anal., and Design of Comm. Links and Networks (1994)
8. Ross, K.W.: Multiservice Loss Models for Broadband Telecommunication Networks. Springer (1995)
9. Chan, W.C., Geraniotis, E.: Tradeoff between blocking and dropping in multicasting networks. In: ICC 1996 Conference Record, vol. 2, pp. 1030–1034 (1996)
10. Karvo, J., Virtamo, J., Aalto, S., Martikainen, O.: Blocking of Dynamic Multicast Connections in a Single Link. In: Conference on Broadband Communications, pp. 473–483 (1998)
11. Karvo, J., Aalto, S., Virtamo, J.: Blocking Probabilities of Multi-layer Multicast Streams. In: Workshop on High Performance Switching and Routing, pp. 268–277 (2002)
12. Samouylov, K., Yarkina, N.: Blocking Probabilities in Multiservice Networks with Unicast and Multicast Connections. In: 8th International Conference on Telecommunications, vol. 2, pp. 423–429 (2005)
13. Lu, Y., Kuipers, F.A., Janic, M., Van Mieghem, P.: E2E Blocking Probability of IPTV and P2PTV. In: Das, A., Pung, H.K., Lee, F.B.S., Wong, L.W.C. (eds.) NETWORKING 2008. LNCS, vol. 4982, pp. 445–456. Springer, Heidelberg (2008)
14. Qiu, T., Ge, Z., Lee, S., Wang, J., Xu, J., Zhao, Q.: Modeling User Activities in a Large IPTV System. In: Internet Measurement Conference, Chicago, pp. 430–441 (2009)
15. Ramos, F., Song, F., Rodriguez, P., Gibbens, R., Crowcroft, J., White, I.H.: Constructing an IPTV Workload Model. In: ACM SIGCOMM, Barcelona, Spain (2009)
16. Yu, G., Westholm, T., Kihl, M., Sedano, I., Aurelius, A., Lagerstedt, C., Ödling, P.: Analysis and Characterization of IPTV User Behavior. In: IEEE Symposium on Broadband Multimedia Systems and Broadcasting, Bilbao, Spain (2009)
17. Cha, M., Rodriguez, P., Growcroft, J., Moon, S., Amatriain, X.: Watching Television Over an IP Network. In: ACM IMC (2008)
18. Wiegand, T., Sullivan, G.J., Reichel, J., Schwarz, H., Wien, M.: Joint Draft 11 of SVC Amendment, Joint Video Team, Doc. JVT-X201 (2007)
19. Wiegand, T., Noblet, L., Rovati, F.: Scalable Video Coding for IPTV Services. IEEE Transactions on Broadcasting 55, 527–538 (2009)

An Adaptive Bandwidth Allocation Scheme for Data Streaming over Body Area Networks

Nedal Ababneh, Nicholas Timmons, and Jim Morrison

WiSAR lab, School of Engineering
Letterkenny Institute of Technology
Port Road, Letterkenny, Ireland
{nedal,nick,jim}@wisar.org

Abstract. We present an adaptive joint routing and bandwidth allocation scheme for traffic streaming in Body Area Networks (BAN). Our solution considers BAN for real-time data streaming applications, where the real-time nature of data streams is of critical importance for providing a useful and efficient sensorial feedback for the user while system lifetime should be maximized. Thus, bandwidth and energy efficiency of the communication protocol must be carefully optimized. The proposed solution takes into account nodes' residual energy during the establishment of the routing paths and adaptively allocates bandwidth to the nodes in the network. We also formulate the bandwidth allocation problem as an Integer Linear Program that maximizes the network utility while satisfying the QoS requirements. We compare the resulting performance of our scheme with the optimal solution, and show that it closes a considerable portion of the gap from the theoretical optimal solution.

Keywords: Wireless Body Area Networks, Routing, Balancing Energy Consumption, Energy-Aware Rate Allocation.

1 Introduction

A Wireless Body Area Network (WBAN, widely simplifies as BAN) supports a variety of medical and non-medical applications. One such important application of growing interest to the medical profession is the health monitoring of patients and those healthy individuals who have potential health risks. BAN is particularly suitable for post-operative care in hospitals and for treatment of chronically ill or aged patients at home. In such application, sensors continuously monitor human's physiological activities and actions, such as health status and motion pattern, which may occur in a more periodic manner, and may result in the applications' data streams exhibiting relatively stable rates [1].

In general, all data in the BAN is generated at the nodes and continuously fed as a stream to the sink. Most of the data will flow to the sink, while a limited amount of control traffic will go in the other direction, which is neglected in this work. For these applications, it is essential to be able to reliably collect physiological readings from humans via sensor nodes. Such networks could

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 90–101, 2012.

benefit from Quality of Service (QoS) mechanisms that support prioritized data streams, especially when the channel is impaired by interference or fading [2]. For example, heart activity readings (e.g., Electrocardiogram (ECG) data) are often considered more important than body temperature readings, and hence can be assigned a higher priority in the system. QoS support is needed to ensure reliable data collection for high priority data streams and to dynamically re-allocate bandwidth as conditions change, especially when the effective channel bandwidth is reduced. In the standard system, when the effective throughput is scarce, data rate for all the nodes drops equally and, thus, the *utility* (the term utility is defined later in Section 2.2) of the whole monitoring system drops significantly. In this case, to guarantee the QoS we need to re-allocate resources from lower priority streams to higher priority streams. Adaptive QoS resource allocation is thus needed to provide bandwidth guarantees, which are essential for reliable data collection in BANs.

Several routing protocols have been proposed recently in the area of BANs [3], mostly validated either using theoretical analysis, testbed implementation or simulation, and involve MAC or power control mechanisms. These protocols deal with limitations and unique characteristics of BANs, such as communication range or irregular traffic patterns. However, research on routing protocols for BANs is still at its infancy and can be categorized into: (1) Cluster Based Routing[4], (2) Temperature Routing [5,6] and (3) Cross-layer protocols[7]. An important issue in BAN, from an application point of view, is offering QoS. A virtual MAC is developed in BodyQoS [2] to make it radio-agnostic, allowing the protocol to schedule wireless resources and to provide adaptive resource scheduling using aggregator nodes. When the effective bandwidth of the wireless channel degrades due to RF interference or RF fading, the bandwidth is adaptively scheduled to meet the necessary QoS requirements. However, integration into existing MAC and especially routing protocols is not optimal [3]. When tackling the specific QoS requirements of BANs, integration with all layers should be strong. Given the difficult environment, where the path loss is high and as a result bandwidth is low, QoS protocols should expect and use all available information about the network .

In this paper, we propose an adaptive routing and bandwidth allocation scheme, ARBA, to improve bandwidth utilization and routing in BAN while balancing out energy consumption throughout the network ensures longer network lifetime. In ARBA, we estimate the utility for different data streams and put some low utility streams offline and thereby maintain high utility of the monitoring system where utility of high priority streams is particularly improved. We select the best possible data rate for each node with efficient link utilization such that the network utility is maximized without violating the QoS constraints. ARBA constructs routing paths in terms of depth and residual energy. The goal of this basic approach is to force packets to move towards the sink through high energy nodes so as to protect the nodes with relatively low residual energy.

The remainder of this paper is organized as follows. In Section 2 we present an ILP formulation of the problem, which allows it to be solved efficiently for

small instances. Section 3 presents the ARBA scheme, which is then evaluated in Section 4. Finally, Section 5 concludes the paper.

2 Optimal Solution

We solve the rate and bandwidth allocation problem based on the tree constructed in Section (3.2). The performance of the proposed model benefits from a well-structured routing tree, however, constructing optimal routing trees is beyond the scope of this paper. For simplicity, we consider a single-radio, multi-channel BAN network, where potentially interfering wireless links should operate on different channels, enabling multiple parallel transmissions. The problem therefore consists of increasing the number of accepted high priority nodes (streams) in the routing tree at highest possible data rate while balancing out energy consumption to extend network operational lifetime.

2.1 The Problem Definition

The proposed BAN network is modeled as undirected graph $G = (V, E)$, called a connectivity graph, where V is the set of vertices representing the nodes in the network, and is composed of a group of sensor nodes denoted as V_n and one or more sinks denoted as V_s. E is the set of edges that represents the communication network topology, edge$(v_i, v_j) \in E$ iff v_i, v_j are within each other's communication range. Hence, nodes form a multi-hop ad-hoc network among themselves to relay traffic to the sink(s). Also, each edge (v_i, v_j) has a physical capacity L_{ij}, which represents the maximum amount of traffic that could pass through this particular link, and each node $v \in V_n$ represents a node, and all nodes in the network are assumed to work on the same fixed transmission power with circular transmission range R_T. At any given time, a node may either transmit or listen to a single wireless channel, with channel capacity C.

The resulting routing tree (final tree) $T = (V', E')$ is a subgraph of G, where E' represents the communication links in the final tree, and V' is the set of nodes and sinks included in the final tree. In order to evaluate the relative importance of nodes and the benefits gained when accepted in the network, we propose a *utility* function (represented below by the objective function) as in [8]. The utility of streaming data from node v_i is denoted by U_i. This utility depends on the minimum acceptable data rate W_{min} by node v_i, and the maximum possible data rate of a node W_{max}. It also depends on P_i, which represents the priority associated to each node (i.e., either high or low) based on the equipped sensor type or sensorial data, while r_i is the current rate (i.e., ratio to W_{max}) of data generated at node v_i. The utility of streaming data decreases with decreasing received data rate and available energy level, and the utility becomes insignificant beyond a certain value (i.e., R_{min} in this case). In cases where the data rate and residual energy level for a given node falls below the acceptable level (i.e., R_{min} and E_{min}), we propose to stop live data streaming and put the node offline.

2.2 Integer Linear Program Formulation

Let z_i be a 0-1 integer variable for each node $v_i \in V_n$, such that $z_i = 1$ if the node v_i is accepted as a traffic source in the resulting tree $v_i \in V'_n$. Let r_i be a positive real variable for each $v_i \in V_n$, representing the effective data rate of v_i such that $r_i = 0$ if v_i is not included in the resulting routing tree (i.e., $z_i = 0$). Let X_{ij} be a 0-1 integer parameter for each edge $(v_i, v_j) \in E$, $X_{ij} = 1$ if the edge is included the resulting tree (i.e., edge $(v_i, v_j) \in E'$). Furthermore, let y_{ij} be a positive integer variable for each edge $(v_i, v_j) \in E'$, showing the amount of data transmitted from node v_i to node v_j (i.e., uplink effective data rate), the receiver could be a sink node. The ILP for the rate (r) and bandwidth (y) allocation problem can thus be stated as follows:

Objective function:

$$max \sum_{i \in V_n} (z_i \times P_i \times U_{min} + \sum_{\forall j \in V:(i,j) \in E} z_i \times (e_i - E_{min}) \times X_{ij} \times U^e_{step} + P_i \times (r_i - (R_{min} \times z_i)) \times U^r_{step})$$

The first term of the utility function is the minimum utility for each node in the network, the second and third terms denote the utility evolution with energy and rate, respectively. Multiplying the second term by the coefficient U^e_{step} and the third term by the coefficient U^r_{step} ensures utility evolution with energy and rate. Also, multiplying each term by z_i guarantees the consideration of the included vertices nodes in the resulting tree only. Each accepted node $v_i \in V'_n$ is assigned rate $r_i \geqslant R_{min}$. The coefficient U_{min} is the minimum utility of each accepted node v_i, and must be set to any positive value greater than zero (i.e., $U_{min} = 1$ in this work).

Constraints:

$$y_{ij} \leqslant L_{ij} \times X_{ij}, \forall i \in V_n, \forall j \in V : (i,j) \in E \tag{1}$$

$$\sum_{\forall j \in V_n:(j,i) \in E} y_{ji} \leqslant C_i, \forall i \in V \tag{2}$$

$$\sum_{\forall j \in V:(i,j) \in E} y_{ij} - \sum_{\forall j \in V:(j,i) \in E} y_{ji} = r_i \times W_{max}, \forall i \in V_n \tag{3}$$

$$z_i \geqslant r_i, \forall i \in V_n \tag{4}$$

$$r_i \geqslant R_{min} \times z_i, \forall i \in V_n \tag{5}$$

$$E_i \geqslant E_{min} \times X_{ij}, \forall i \in V_n, \forall j \in V : (i,j) \in E \tag{6}$$

Constraint (1) ensures that the uplink effective rate of each included edge in the resulting tree is bounded by the maximum physical link capacity. Constraint (2)

provides an upper bound (i.e., the cell capacity) on the relay load constraint, it ensures that the incoming flow is always less than cell capacity C_i. Constraint (3) is for flow conservation. It implies that the difference between the outgoing traffic and the incoming traffic at node v_i is the volume of traffic generated by node v_i itself. Since all data flows are originated from nodes and do not return to the nodes, it will not lead to cycles in our solution. All data flows will eventually reach the sink. Constraint (4) ensures that node data rate is assigned to accepted nodes in the final routing tree only, i.e., nodes not included in the resulting tree have rate equal to zero. Constraint (5) ensures that each accepted node v_i in the resulting tree has to be assigned rate $r_i \geqslant R_{min}$, this is to satisfy the QoS requirements. Finally, Constraint (6) ensures that available energy for each accepted node v_i in the resulting tree has to be $\geqslant E_{min}$, as E_{min} is the residual energy value where a node is still able to send/receive messages properly.

3 Adaptive Routing and Bandwidth Allocation Scheme

In this section, we describe the design and implementation issues of the proposed scheme in depth. Our solution consists of four phases described as follows:

3.1 Topology Discovery

One of the essential aspects of any BAN is the topology discovery. In our network scenario, the nodes use a simple technique to discover neighboring nodes as well as to detect transient channel variations and topology changes (e.g., node or link failure, injection of new nodes, etc.) similar to the one proposed in [9]. During some predetermined intervals, the nodes exchange HELLO messages to detect nodes in the surrounding neighborhood. In order to keep a constant record of neighboring activity, each node in the network will form a registry of neighbors (i.e., neighbor table). This registry will hold only the required information for forming, maintaining, and breaking connections.

3.2 Energy-Aware Routing Tree Construction

To achieve high network performance in BAN, the route construction within the network is a crucial task. To maximize network operable lifetime and throughput capacity, we use nodes' residual energy as a selection metric, if two nodes have the same residual energy; the node with shorter path to the sink will then be selected as a parent. The sum of the residual energy of all nodes that reside in the path of a node, say node *a,* towards the sink is termed the *path-energy*, $P_{energy}(a)$. For each node to choose its parent (selecting node's parent is equivalent to build routing paths), it chooses the neighbor which renders the largest path-energy among all other potential parents (i.e., neighbor nodes). In fact, the path-energy metric is the sum of the residual energy metric of the node's predecessors. For example, assume *sink-c-b-a* is a path in the routing tree, and $P_{energy}(a)$ is the sum of node's residual energy through the path from node a whose parent

node is b to the sink. Thus, $P_{energy}(a)$ is calculated as follows: $P_{energy}(a) = energy(a) + energy(b) + energy(c)$. We suppose all the nodes enter the network one by one. When a node, say node a, enters, all its neighbors are eligible to be a's parent. In order to improve the network performance, node a selects a parent node, say node b, with largest value of $P_{energy}(b)$. The parent node is then selected such that $Prnt(a) = max_{i \in neighbor(a)} P_{energy}(i)$.

3.3 Rate and Bandwidth Allocation

Our solution consists of computing the number of high priority streams (flows) in the tree and then to assign a bandwidth (capacity) to each branch that is proportional to this quantity. We then check whether this amount of bandwidth is large enough to accommodate the data stream (i.e., $> W_{min}$). If this is not the case, the flow counts are adjusted accordingly and the capacity assigned to this branch is released. It means that the flow(s) that pass through this branch have to be put offline in this critical situation where there is not enough available bandwidth. The capacity of this branch is redistributed to the neighboring branches. Similarly, these capacity allocation must also be compared to the physical capacity of each branch (i.e. L_{ij}). When the assigned amount of capacity is too large to be accommodated by the branch, it is then adjusted and the remaining capacity is assigned to the neighboring branches. This procedure is repeated iteratively from the leaf nodes towards the sink in a leaves-to-sink manner. When several priorities are available (we consider two priority levels in this work i.e., high and low), the algorithm starts with the highest level of priority. The remaining bandwidth is then shared by the lower level of priority by a new instance of the algorithm and so on. The proposed algorithm runs independently at each sink in a centralized manner. The three steps of the algorithm are then described as follows:

Initialization Step: The algorithm starts with initialization of the variables. More precisely, the number of flows with priority p (p is either high or low in this paper) in the tree and denoted by F_p. The parent node of a node i is denoted by $prnt(i)$. Consequently, the set $Child(i)$ represents the set of nodes which are directly connected to node i. $H(i)$ denotes the hierarchical level of node i in the tree. (e.g., $H(sink) = 0$, $H(i) = 1$ for a node i that is directly connected to the sink, etc.).The capacity (bandwidth) allocated to node i is denoted by $UplinkCap(i)$. It corresponds to the bit rate available for this node at the uplink (how much traffic it can relay including its own generated data). This value is bounded by the cell capacity C at each node as well as to the physical link capacity $L_{i,prnt(i)}$. The $UplinkCap(i)$ is dynamically updated to ensure the accuracy of the algorithm based on the available capacity upstream and the number of high priority flows. Once initialized, these $UplinkCap$ values should be coherent. In fact, the available $UplinkCap$ of a node can be greater than that of the nodes in the path to the sink. Figure 1 demonstrates the procedure used to ensure a node is assigned $UplinkCap$ not greater than the $UplinkCap$ of the nodes in its path towards the sink.

```
1: ▷ Ensure consistent link capacity along the path
2: for each level h ← 1 to TreeDepth − 1 do
3:     for each node i such that H(i) ← h do
4:         if UplinkCap(i) > UplinkCap(prnt(i)) then
5:             UplinkCap(i) ← UplinkCap(prnt(i))
```

Fig. 1. Pseudo-code of the ensuring consistent link capacity step

```
1: ▷ Assign uplink capacity proportionally to Flow count
2: for each level h ← TreeDepth to 1 do
3:     for each node i such that H(i) ← h and F_p > 0 do
4:         UplinkCap(i) ← Σ_{j∈Child(i)} UplinkCap(j)
5:         UplinkCap(i) ← UplinkCap(i) + (AvailableBandwidth/F_p)
6:         ▷ Check physical link capacity constraints
7:         if UplinkCap(i) > L_{i,prnt(i)} then
8:             UplinkCap(i) = L_{i,prnt(i)}
9:         ▷ If allocated bandwidth does not satisfy QoS requirements,
           put node i offline
10:        if UplinkCap(i) < W_min then
11:            UplinkCap(i) ← UplinkCap(i) − r_i
12:            F_p ← F_p − 1, r_i ← 0
```

Fig. 2. Pseudo-code of the bandwidth assignment step

Rate and Bandwidth Allocation Step: Next, the main capacity allocation procedure, illustrated in Fig. 2, is executed (once for each priority level (high or low), starting with nodes with high priority value of p). Once this procedure is completed, the data rate of the nodes of priority p can be then determined. Starting from leaf nodes, a leaf node i will be assigned rate $r_i = UplinkCap(i)$, such that the QoS requirements are satisfied. The node will update its incoming and outgoing traffic ($inTraffic$ and $outTraffic$ variables, respectively) to allow accurate rate allocation for upstream nodes. For leaf node i, $inTraffic = 0$ and $outTraffic = r_i$. Nodes in the upper level will then compute their $inTraffic$, and their rates will be calculated as follows: $r_j = UplinkCap(j) - inTraffic(j)$. The rate assignment process will recursively continue level by level until the sink is reached.

Allocation Improvement Step: After the nodes' data rates are assigned according to the previous steps, we check for any extra available bandwidth at each node. For instance, a leaf node might be allocated an $UplinkCap > W_{max}$, in this case $UplinkCap - W_{max}$ extra bandwidth is available. In our algorithm, such extra bandwidth will be allocated to nodes without violating QoS constraints. Checking for extra bandwidth process commence in a leaves-to-sink manner, where unallocated bandwidth at node i is computed as $UplinkCap(i) - (r_i + inTraffic(i))$ and will be moved to the sink to be allocated to other nodes where possible. This procedure will be repeated recursively until

there is no extra bandwidth available in the network or the given constraints (QoS) are not permitting any further bandwidth allocation.

3.4 Load Balancing Routing and Energy Usage

ARBA employs a load balancing strategy so that all nodes remain up and running together for as long as possible. The proposed algorithm is run repeatedly in a series of rounds.To ensure fairness, the group of selected nodes is rotated periodically to ensure energy-balanced operations. A node switches state from time to time between being *Accepted* as traffic source and being *non-Accepted*. While *Accepted*, the node continuously streams sensorial data and, possibly, forwards other nodes' data up towards the sink. After a node remains in *Accepted* state for time T_{round}, it changes its state to *discovery* state to give a chance to other nodes in the neighborhood to be accepted by the sink. Recall that, nodes are ranked according to their importance (type of equipped sensors and sensorial data) and remaining energy levels. When the accepted node changes its state to *discovery*, it is more likely that it has less remaining energy in its battery than its neighbors because presumably the neighbor nodes were in the *power-saving* (non-accepted) mode for time T_{round}. Consequently, the node that was accepted is less likely to remain in its current state after the *discovery* phase. Also, when a node detects topology change (e.g., link or node failure) it sends an EMERGENCY message to the sink to react and adaptively re-allocate available bandwidth to the nodes. It is worth mentioning that another reason why we chose to run ARBA in rounds is to react to topology changes (e.g., node or link failure, adding new nodes, etc.) and dynamic network conditions. Figure 3 illustrates an example of 200 seconds application cycle with ARBA.

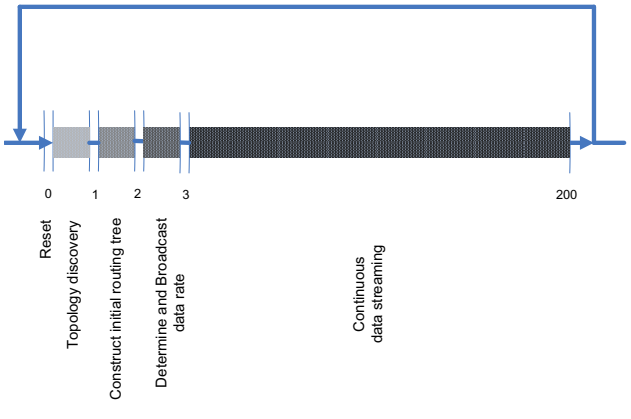


Fig. 3. The different phases of the network cycle with ARBA

4 Performance Evaluation

For the evaluation, we have conducted an extensive set of experiments using a VC++ coded simulator. We studied the performance of the the proposed scheme,

ARBA, by comparisons with the optimal solution. We simulated 16 node network illustrated in Fig. 4(a), where high priority nodes are randomly selected in each run. To ensure the fairness of the comparison, we used the energy-aware routing algorithm presented in Section (3.2) for both solutions (i.e., optimal and ARBA). To be accurate, we solved the integer linear program presented in Section 2 by using AMPL and CPLEX. The wireless channel capacity is $C = 2$ Mbps, and link capacity $L = 1$ Mbps. The minimum acceptable data rate generated by each node R_{min} is 128 Kbps and R_{max} is fixed at 512 kbps. The minimum utility of each accepted node $U_{min} = 1$ and $U^e_{step} = U^r_{step} = 1/5$. Our goal of implementing ARBA scheme is to minimize the total energy spent in the network to communicate the information gathered by sensor nodes to the information-processing center (i.e., the sink), while achieving higher network utility. . For all simulation results in this paper, each point in the plots is averaged over 10 runs, and the simulation lasts for 5000 s. One of the assumptions commonly made is the

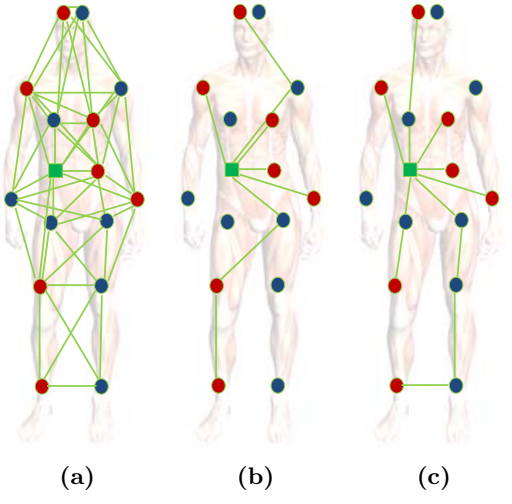


Fig. 4. Example of BAN routing tree formation based on ARBA: (a) original topology, (b) ARBA at round 1 and (c) ARBA at round 10. A red filled circle represents a high priority node, a blue filled circle represents a regular node and a green filled square represents the sink. Solid line indicates communication link, and each node is sending a Constant Bit Rate (CBR) stream to the sink with radios capable of transmitting up to 0.5 Mbps

sink node is located at the waist. To investigate how the sink placement affects the network performance behavior, we repositioned the sink at the ankle. Varying number of high priority nodes (data streams) we are interested in evaluating (1) utility, (2) energy dissipation distribution and (3) total size of routing trees.

Utility: To study the impact of the number of high priority nodes on the network performance we vary the number of high priority nodes from 1 to 8. The utility of a data transmission rate below R_{min} (i.e., 128 kbps) is considered

to be insignificant, hence, the node is put offline. Utility for each offline node is assumed to be zero. Other settings are the same as above. We can clearly see in Fig. 5 that ARBA scheme experiences a linear utility increase with the increase of high priority nodes. This is because as more high priority nodes are deployed, ARBA will try to accommodate as many of them as possible at the best possible data rate, which results in a better utility. In Fig. 5, we note that the gap between the outcome of ARBA and the optimal solution tends to remain nearly constant, which suggests that the number of high priority nodes (i.e., the problem size) has little impact on the performance ratio between the two. It is also interesting to note that, sink location has no impact on the network performance in terms of utility.

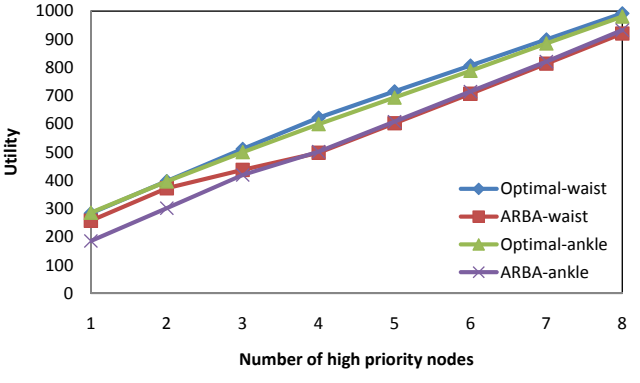


Fig. 5. Performance comparison of network utility as a function of number of high priority nodes

Load Distribution and Energy Savings: Network lifetime is a crucial metric of BAN, but it can be a crude measure of actual energy consumption because node life is binary, so n about to die nodes are considered good as n fresh nodes. In order to quantify how much energy ARBA saves, we instead plot the residual energy per node. In order to observe how well ARBA promotes load balancing and thus saves energy among the nodes compared to the optimal solution, we ran a simulation for 2000 s. At the outset, each node had 2 J battery energy. For simplicity, we only account for the radio receiving and transmitting energy. Figure 6 shows relative residual energy at the end of simulation across nodes of ARBA and the optimal solution. It is obvious that ARBA performs well in terms of balancing out energy consumption in the network, which yields to longer network lifetime. As expected, the sink location at the waist is more energy efficient than the ankle location. This is mainly due to the reduced average hops between source nodes and the sink resulting in lower forwarding energy cost. This effect is more pronounced for the ankle location. This confirms that depth has a negative impact on network energy distribution as the sink will have smaller node degree and network congestion may occur in this case. It is interesting to

note that when sink is placed at waist, nodes have more residual energy, this is because the routing tree size (i.e., number of nodes that forms the routing backbone) is smaller as depicted in Fig. 7.

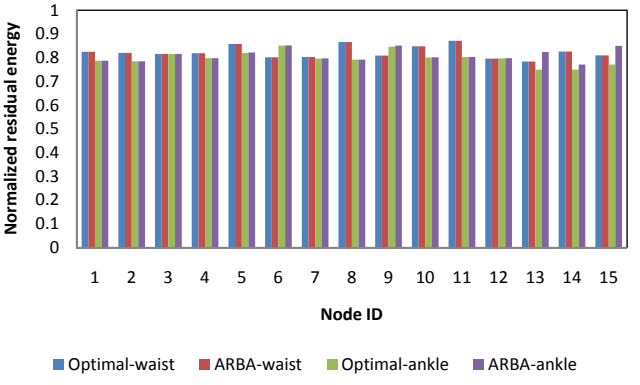


Fig. 6. Residual energy distribution of nodes after 2000 s simulation of ARBA: The nodes are initially equipped with 2 J battery energy

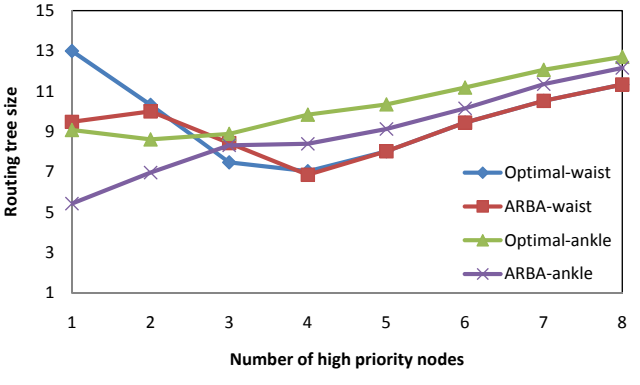


Fig. 7. Routing tree size as a function of number of high priority nodes

Routing Tree Size: Figure 7 shows the impact of the number of high priority nodes on the total size of the routing trees (i.e., total number of nodes in the resulting routing backbone) that each solution obtains. We note that all curves depict a larger routing tree for larger number of high priority nodes, which suggests that when increasing number of high priority nodes in the network the proposed protocol tries to accept as many nodes in the network as possible and thus increases the routing tree size. Tree size influences the total consumed energy and thus network operable lifetime. The larger the tree the more energy consumed in the network. It is apparent that, when the sink is located at the ankle both solutions attain larger routing trees, this is because far away nodes need to have more relay nodes to communicate their data to the sink.

5 Conclusion and Future Work

This paper proposes an adaptive routing and bandwidth allocation scheme, termed ARBA, that performs real-time monitoring of complex conditions on streaming data from various body sensors within a Body Area Network (BAN). We studied the energy saving and how it is affected by factors such as sink placement and the number of high priority data streams. It is shown in the paper that the ARBA is energy efficient for streaming communication while balancing energy consumption across the BAN guarantees longer lifetime.

References

1. Chen, M., Gonzalez, S., Vasilakos, A., Cao, H., Leung, V.C.M.: Body area networks: A survey. Mobile Networks and Applications 15 (2010)
2. Zhou, G., Lu, J., Wan, C.-Y., Yarvis, M.D., Stankovic, J.A.: Bodyqos: Adaptive and radio-agnostic qos for body sensor networks. In: Proc. of the IEEE INFOCOM, pp. 565–573 (2008)
3. Ullah, S., Higgins, H., Braem, B., Latre, B., Blondia, C., Moerman, I., Saleem, S., Rahman, Z., Kwak, K.: A comprehensive survey of wireless body area networks. Journal of Medical Systems, 1–30 (2010)
4. Moh, M., Culpepper, B.J., Dung, L., Moh, T.-S., Hamada, T., Su, C.-F.: On data gathering protocols for in-body biomedical sensor networks. In: Proc. of IEEE GLOBECOM (2005)
5. Tang, Q., Tummala, N., Gupta, S.K.S., Schwiebert, L.: Communication scheduling to minimize thermal effects of implanted biosensor networks in homogeneous tissue. IEEE Trans. Biomed. Eng. 52(7), 1285–1294 (2005)
6. Bag, A., Bassiouni, M.A.: Energy efficient thermal aware routing algorithms for embedded biomedical sensor networks. In: Proc. of IEEE MASS (2006)
7. Braem, B., Latre, B., Moerman, I., Blondia, C., Demeester, P.: The wireless autonomous spanning tree protocol for multihop wireless body area networks. In: Proc. of MobiQuitous (2006)
8. Ababneh, N., Rougier, J.-L.: Optimal rate assignment for higher utility wimax surveillance systems. In: Proc. of IEEE WCNC (2012)
9. Sohrabi, K., Gao, J., Ailawadhi, V., Pottie, G.J.: Protocols for self-organization of a wireless sensor network. IEEE Personal Communications 7(5), 16–27 (2000)

EQR: A New Energy-Aware Query-Based Routing Protocol for Wireless Sensor Networks*

Ehsan Ahvar[1], René Serral-Gracià[2], Eva Marín-Tordera[2], Xavier Masip-Bruin[2], and Marcelo Yannuzzi[2]

[1] Department of Information Technology and Communication
Payame Noor University, Iran
ahvar@pnu.ac.ir
[2] Advanced Network Architectures Lab (CRAAX)
Technical University of Catalunya (UPC), Spain
{rserral,eva,xmasip,yannuzzi}@ac.upc.edu

Abstract. Over the last years, a number of query-based routing protocols have been proposed for Wireless Sensor Networks (WSNs). In this context, routing protocols can be classified into two categories, energy savers and energy balancers. In a nutshell, energy saving protocols aim at decreasing the overall energy consumed by a WSN, whereas energy balancing protocols attempt to efficiently distribute the consumption of energy throughout the network. In general terms, energy saving protocols are not necessarily good at balancing energy and vice versa. In this paper, we introduce an Energy-aware Query-based Routing protocol for WSNs (EQR), which offers a good trade-off between the traditional energy balancing and energy saving objectives. This is achieved by means of learning automata along with zonal broadcasting so as to decrease the total energy consumption. We consider that, in the near future, EQR could be positioned as the routing protocol of choice for a number of query-based WSN applications, especially for deployments where the sensors show moderate mobility.

Keywords: WSN, query-based routing, energy.

1 Introduction

Wireless Sensor Networks (WSNs) are bringing unprecedented abilities to monitor different types of physical phenomena, and thereby control specific operations or processes without human intervention. In the near future, a large number of low-power and inexpensive sensor devices will become part of the landscape in cities, highways, farms, airports, etc. Since the deployment and operation of these networks is application dependent, the possibility of having a unified routing strategy capable of meeting the requirements of all foreseeable applications is simply unfeasible—e.g., consider the difference in terms of routing requirements between a fixed WSN for an urban monitoring application and those of a mobile WSN for finding survivors trapped after a natural disaster.

* This work was supported in part by the Spanish Ministry of Science and Innovation under contract TEC2009-07041, and the Catalan Research Council (CIRIT) under contract 2009 SGR1508.

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 102–113, 2012.

Indeed, the design and implementation of routing schemes able to efficiently support the exchange of information—basically by saving processing overhead and thus energy—is particularly challenging in WSNs. A number of theoretical and practical limitations must be thoroughly examined and considered during the design and the implementation phases, including the methods for route discovery and maintenance, the methods for handling losses and failures as well as potential radio interferences during network operation. Until now, many routing algorithms and protocols have been proposed for WSNs (see, for instance, [1], [2], [3]), among which we found a category known as *query-based routing*. In this category, a station S sends queries to find specific events among the sensor network (e.g., a query may encode a question such as: is the measured temperature higher than 40C?). In this context, the strategies used for routing the queries and their corresponding replies can be classified into two major groups, namely, energy savers and energy balancers. The former try to decrease the energy consumption of the network as a whole and thereby increase the operation lifetime—usually leading to the utilization of shortest paths. The latter, on the other hand, try to balance the energy consumption of the nodes in order to prevent the potential partitioning of the network. Overall, finding the best route based only on energy balancing objectives may lead to rather long paths and increased delays, whereas finding the best route based only on energy saving and optimal distance objectives may lead to network partitioning.

In light of this, we propose a routing strategy applicable to various forms of query-based applications that offers a reasonable trade-off between the energy saving and energy balancing objectives. More precisely, we propose an Energy-aware Query-based Routing protocol for WSNs called EQR, which is supported by learning automata and uses zonal broadcasting to decrease the total energy consumed. Our initial results demonstrate the potential and effectiveness of EQR, making it a promising candidate for a number of WSN applications, especially those where the sensors have moderate mobility.

The rest of the paper is organized as follows. In Section 2, we overview related work. Section 3 presents the main contribution of this paper which is basically the EQR routing strategy. The assessment of EQR is covered in Section 4, and finally, Section 5 concludes the paper.

2 A Brief Outline of Related Work

One of the references in the subject of energy-aware query-based routing protocols is RUMOR [4]. RUMOR is an energy saving protocol that provides an efficient mechanism combining push and pull strategies to obtain the desired information from the network. In RUMOR, the nodes generating events send notifications that leave a “sticky” trail along the network. Then, when query agents visit a node where an event notification agent has already passed, they can find pointers (i.e., the trail) toward the location of the corresponding source. In general terms, when a node receives a query two things can happen: i) the node has already a route toward the target event, so it only needs to forward the query along the route; or ii) the node does not have a route, and therefore, it forwards the query to a random neighbor. The random selection of the neighbor in this

case is relatively constrained, since each node keeps a list of recently visited neighbors to avoid visiting them again.

Clearly, the forwarding strategy in RUMOR might end up producing spiral paths, so the intuition for improving RUMOR is to reduce its level of routing indirection. To this end, Cheng-Fu Chou et al. proposed in [5] the Straight Line Routing (SLR) protocol, which aims at making the routing path grow as straight as possible. In recent years, Shokrzadeh et al. have made significant efforts to improve the basis of RUMOR in different aspects, and proposed Directional RUMOR (DRUMOR) [6]. Later on, Shokrzadeh et al. improved their DRUMOR protocol by means of what they called Second Layer Routing (SecondLR) [7]. The latter uses geographical routing right after locating the source of an event, and the authors have shown that this approach considerably improves the performance of DRUMOR.

Despite these efforts, current query-based routing protocols are mainly energy savers, and show relatively poor performance when it comes to balance the energy consumption. Although several papers can be found in literature addressing the issue of energy balancing in the context of routing protocols for WSNs, to the best of our knowledge, our work is the first attempt toward a better balance of the energy consumed in destination initiated query-based routing protocols.

3 Energy-Aware Query-Based Routing (EQR)

3.1 Overview

EQR is a routing scheme especially designed to consider both energy and distance while routing packets across a network. To this end, EQR balances whenever possible the load among the different sensors with a twofold goal: avoid that the sensors run out of battery while still keeping the routes to reach the destinations relatively short. As shown in Fig. 1, the architecture that supports EQR is basically composed of an API, a Zone Estimation Unit (ZEU), a Probability Computation Unit (PCU), a Data Path Selection Unit (DPSU), and an Error Detection Unit (EDU). In this architecture, the

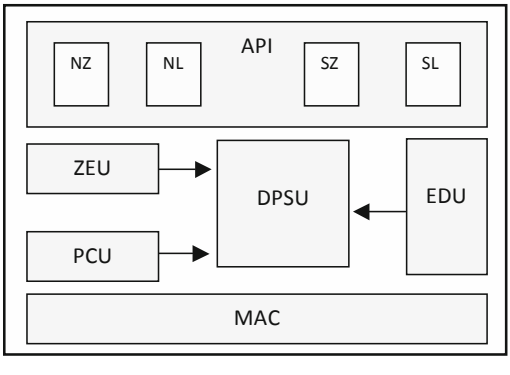


Fig. 1. The architecture supporting the EQR routing protocol

EQR routing protocol provides four application-level API modes called Non-specific Zone (NZ), Non-specific Location (NL), Specific Zone (SZ), and Specific Location (SL). The SZ and NZ modes are for event monitoring: the former for applications that monitor events in a specific zone of the network, while the latter is used when there is no prior knowledge about the area where such events occur. In this second mode, the query packets must be broadcasted throughout the entire network to find potential events. The two remaining modes, i.e., SL and NL, are designed for querying a given node and getting information directly from it: the first when there is prior knowledge about the expected location of the node, and the second when its location is unknown.

The ZEU unit is responsible for estimating the query zone, and similarly to [8], it is based on a low-power GPS module which is actively powered on and off in order to save energy. As its name indicates, the EDU is designed for detecting errors, while the DPSU is the core routing module and hence the one responsible for choosing the next-hop during the packet forwarding process. In EQR, the neighbor with the highest probability is designated as the "primary" neighbor, and it is the one that will be chosen by the DPSU as the next-hop. The probability associated to each neighbor is computed by the PCU unit, and the process for assigning and updating these probabilities shall be described in detail along this section. As shown in Fig. 1, the constellation of modules is mainly designed to assist the DPSU unit. Due to space limitations, we cannot develop in detail all the modules in the architecture, so the main focus of this work is centered on the DPSU and PCU units, leaving the description of the rest of the architecture and its building blocks for an extended version of this paper.

In EQR, the design of the DPSU and PCU are based on learning automata, where we also integrate other well-known strategies, such as piggybacking and overhearing techniques. More precisely, the next-hop in EQR is chosen using learning automata, which compute the probabilities and select the best possible neighbor considering two metrics: their energy level as well as the distance to reach the destination.

3.2 Initialization Process

When the WSN starts its operation, the ZEU unit in the station S that will issue the queries will lack any zonal information. Therefore, the query modes used by the station S at the beginning of the operations will typically be NL or NZ, which means that the entire region is assumed to be the "query zone". Once the ZEU starts collecting information, the subsequent queries issued by the station S can be made using the SZ or SL modes, thus exploiting the advantages of zonal broadcasting. In general terms, the station S will gather zonal information in its ZEU unit, and whenever required it will generate a query packet in which it will broadcast its ID, S_{ID}, the query mode (NZ, NL, SZ, or SL), its position, $(x_S(t), y_S(t))$, the zone information, and optionally, the destination ID, d_{ID}. The zone is defined in the query packet as a rectangle specified by four corners, so that the nodes receiving the query can forward or discard the same depending on their location. For instance, on the left-hand side of Fig. 2, when node j receives the query originally sent from the station S and forwarded by node k, it will process the packet but it will not forward it, since j is not within the "query zone" delimited by the four corners. Instead, nodes i and k will process and forward the query given that they are inside the zone determined by the ZEU unit.

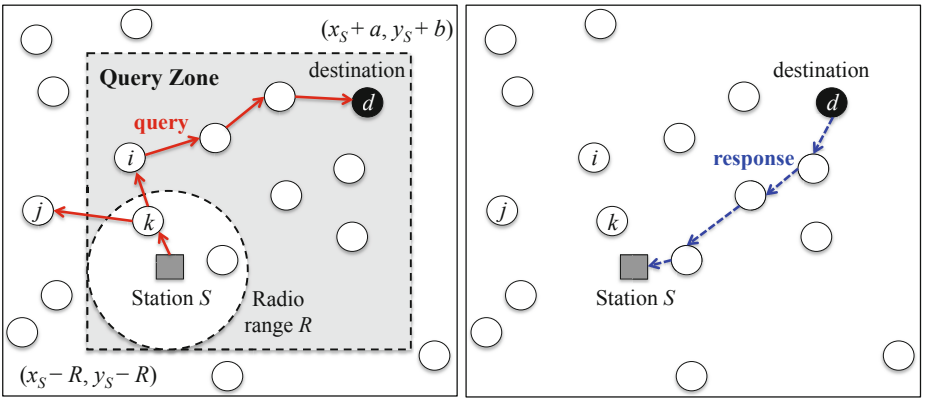


Fig. 2. (Left-hand side) Zonal broadcasting: nodes i and k will process and forward the query since they are inside the "query zone" determined by the ZEU, while node j will discard the query right after its processing. (Right-hand side) The response path is determined by the DPSU.

When a node i receives a query for the first time from a neighbor k, this produces a new entry in its "Neighbor List"— observe that the neighbor may also be the station S if node i is within the radio range of S. The Neighbor List is composed of four fields, and the data to be stored into these fields can be computed from the information contained in the query packet received from neighbor k. The fields in the Neighbor List are basically the following: 1) the neighbor's ID, k_{ID}; 2) its energy level at time t, $\mathcal{E}_k(t)$ (except for $k = S$, since the station's energy is assumed to be inexhaustible); 3) its position, $(x_k(t), y_k(t))$; and 4) the probability $P_k(t)$ associated with neighbor $k \mid k \neq S$, which is computed according to expression (1). As mentioned above, the probability $P_k(t)$ is computed using the PCU unit, where $\mathcal{E}_m(t)$ is the energy level advertised by neighbor m, N_i is the size of node i's Neighbor List (including now node k), $D_m(t)$ is the distance advertised by neighbor m to the station S, and the sums in the denominators represent the terms to normalize the probabilities and to make $\sum_{k=1}^{N_i} P_k(t) = 1$. The rationale of using (1) is that it produces a good balance between energy and distance, though at the cost of the potential recomputation of the probabilities right after receiving each query, since the sum of the probabilities for all neighbors must be equal to one.

$$P_k(t) = \frac{1}{2}\left(\frac{\mathcal{E}_k(t)}{\sum_{m=1}^{N_i}\mathcal{E}_m(t)} + \frac{\frac{1}{D_k(t)}}{\sum_{m=1}^{N_i}\frac{1}{D_m(t)}}\right) \qquad \forall k \neq S \mid k \leq N_i \tag{1}$$

In a nutshell, the nodes within the query zone distribute the queries complementing the information originally sent by the station S with their own ID, their energy level, their position and distance to the station, and a list of hops to prevent forwarding loops. This process is repeated until the event is found in a node d. At this point, every node in the zone knows the energy levels of their neighbors and the distance from them to the station S. As shown on the right-hand side of Fig. 2, the response to the station could use a different path, since this will depend on the primary neighbor chosen by node d. The response packets from d to S will follow the highest probability path, since every

node in the query zone has already chosen its primary neighbor to reach the station S. Overall, the queries are routed using zonal broadcasting while the responses are routed through the highest probability chain. This approach not only simplifies the design of the sensors but also saves a considerable amount of their energy.

The rest of this section shall be mainly devoted to describe the role of the learning automata and the process for updating the probabilities in the Neighbor List.

3.3 The Update Mechanism

The basics of the mechanism are illustrated in Fig. 3, and it works as follows. A node A selects the node with the highest probability as its primary neighbor (node B in this case), and then it sends the response packet to it. As shown in Fig. 2, this response corresponds to a query previously issued by the station S. An important feature in EQR is that every node along the highest probability chain piggybacks its energy level, its position, and its distance to the station S in the response packet, using a set of dedicated fields to this end. Thus, when node B receives the response packet, it updates the energy level and position of node A in its Neighbor List, and it may also update the probability $P_A(t)$ depending on the energy and distance reported by node A. As shown on the left-hand side of Fig. 3, all the other neighbors of A overhear the response and perform the same updates as B, though they discard the packets right after processing them. The routing process continues now with node B selecting node C as its highest probability neighbor. When B sends the response to C, its is now node A the one that overhears the packet sent by B, and thereby updates the position and energy level of the latter (cf. the right-hand side of Fig. 3).

Based on piggybacking and overhearing techniques, the nodes can compute and mutually update the probabilities in their Neighbor Lists according to the energy levels and distances obtained from their neighbors. In the example below, if the metrics received from node B are acceptable, then node B is rewarded by the learning automaton in A, and the probability associated to B is increased in node's A Neighbor List. Otherwise, B is penalized and its probability is decreased. In our model, we considered four behavioral cases for rewarding or penalizing a neighbor B. First, the worst case occurs when:

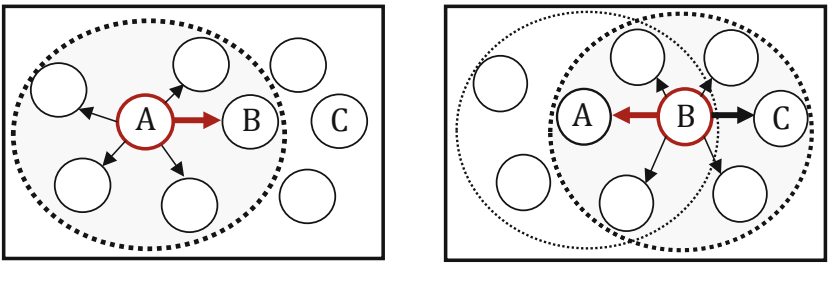


Fig. 3. Mutual updates of the energy levels, distances, and probabilities. (Left-hand side) The neighbors of node A perform the updates. (Right-hand side) Idem for the neighbors of node B.

$$\frac{\mathcal{E}_B(t)}{<\mathcal{E}_A(t)>} \times \frac{<D_A(t)>}{D_B(t)} < 1 \tag{2}$$

where $<\mathcal{E}_A(t)>= \sum_{m=1}^{N_A} \mathcal{E}_m(t)/N_A$ represents the average energy of all neighbors of node A, and $\mathcal{E}_B(t)$ stands for the energy level obtained from B. Likewise, $<D_A(t)>$ represents the average distance of all neighbors of A to the station S, while $D_B(t)$ represents the distance to S reported by node B. In this first case, the energy–distance relationship is below the average, and thus the learning automaton in A will penalize node B with a factor β—more details about this computation can be found later in Section 3.4. Second, a relatively bad situation occurs when:

$$\frac{\mathcal{E}_B(t)}{<\mathcal{E}_A(t)>} \times \frac{<D_A(t)>}{D_B(t)} = 1 \tag{3}$$

In this case, the penalization chosen is $\beta/2$. Third, if the relationship is such that:

$$1 < \frac{\mathcal{E}_B(t)}{<\mathcal{E}_A(t)>} \times \frac{<D_A(t)>}{D_B(t)} < 1.5 \tag{4}$$

then node A will reward node B with $\alpha/2$ — the details about this computation are described in Section 3.4. And fourth, we consider that the best case occurs when:

$$\frac{\mathcal{E}_B(t)}{<\mathcal{E}_A(t)>} \times \frac{<D_A(t)>}{D_B(t)} \geq 1.5 \tag{5}$$

where the reward chosen in this case is α. We proceed now to describe in more detail the incentive mechanism outlined above.

3.4 The Incentive Mechanism

As previously discussed, to incentivize the selection of next-hops that show a good energy–distance relationship, we designed a simple yet effective mechanism for rewarding or penalizing neighbors.

Reward Computation — The reward parameter α is used during the update mechanism in order to grant more priority to the nodes with more possibilities to forward the response packets to the station. The value of α is computed using:

$$\alpha = \lambda_\alpha + \delta_\alpha \left(\frac{\mathcal{E}_B(t)}{<\mathcal{E}_A(t)>} \times \frac{<D_A(t)>}{D_B(t)} \right) \tag{6}$$

where λ_α is the minimum reward granted to a well-positioned node, and δ_α is the limiting factor for the reward.

Penalty Computation — Similarly, we use:

$$\beta = \lambda_\beta + \delta_\beta \left(\frac{\mathcal{E}_B(t)}{<\mathcal{E}_A(t)>} \times \frac{<D_A(t)>}{D_B(t)} \right)^{-1} \tag{7}$$

where analogously to the reward mechanism, λ_β is the minimum penalty, and δ_β is the limiting factor. Note that in (6) and (7), the better (worse) the energy–distance relationship the greater the reward (penalization) assigned to node B.

Upon obtaining the energy and distance metrics from node B, the learning automaton in node A will update the probabilities of its N_A neighbors based on equations (8) and (9). The former applies for the rewarding cases, i.e., the third and fourth cases described above, with $x_\alpha = \alpha/2$ and $x_\alpha = \alpha$, respectively. The latter corresponds to the penalization cases, that is, the first and second cases, with $x_\beta = \beta$ and $x_\beta = \beta/2$, respectively.

$$\begin{cases} P_B(t_{n+1}) = P_B(t_n) + x_\alpha[1 - P_B(t_n)] \\ P_k(t_{n+1}) = (1 - x_\alpha)P_k(t_n) \qquad \forall k \mid k \neq B \wedge k \leq N_A \end{cases} \tag{8}$$

$$\begin{cases} P_B(t_{n+1}) = (1 - x_\beta)P_B(t_n) \\ P_k(t_{n+1}) = \dfrac{x_\beta}{N_A - 1} + (1 - x_\beta)P_k(t_n) \qquad \forall k \mid k \neq B \wedge k \leq N_A \end{cases} \tag{9}$$

A summary of the collective processing performed by EQR inside the query zone is shown in Fig. 4.

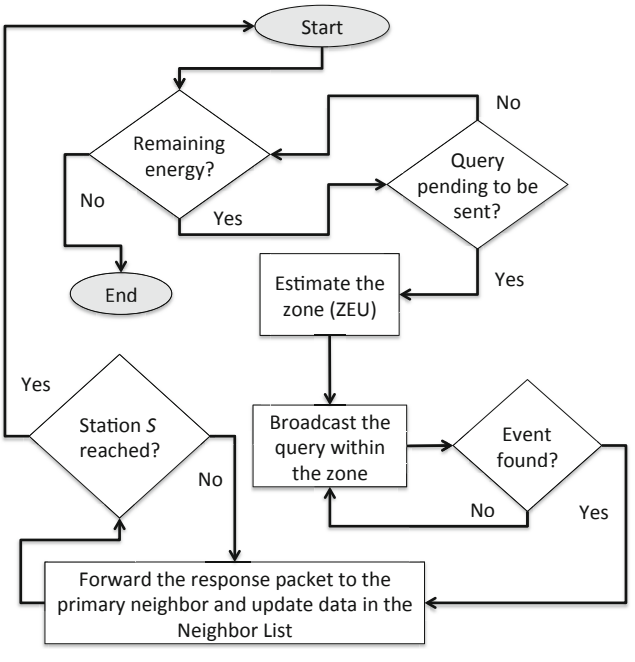


Fig. 4. Collective processing performed by the nodes inside the query zone

4 Performance Evaluation

In this section, we evaluate the performance of EQR by comparing against the following set of routing protocols: RUMOR [4], Straight Line Routing (SLR) [5], Directional Rumor (DRUMOR) [6], and Second Layer Routing (SecondLR) [7]. To this end, we used the Glomosim simulator developed by UCLA [9]. We proceed now to describe the simulation model used and the results obtained.

4.1 Simulation Model

During the simulations we made the following assumptions. We used a surface of terrain of 1000 m $\times$ 1000 m, where the sensors have a maximum energy level of $7 \times 10^{-4} mW$. The radio range was set to 177 m, with an available bandwidth of 2 Mbps and a radio TX power of 4.0 dBm using IP over 802.11. The duration of each simulation was of 4 hours, and the tests were run under various conditions, such as with different amount of sensors, namely, 800, 1000, and 1200 nodes. Moreover, the placement of the sensors in the terrain was randomly chosen. It is worth highlighting that, even though the placement of the nodes was random, once it was set it remained fixed for rest of the trials to obtain comparable results across experiments. In the simulations presented here, the traffic in the network was always initiated by a source station S, which periodically acquires information from a particular sensor d. Once the query is received at d, the sensor will immediately send back the response to S with the requested information.

4.2 Simulation Results

To evaluate the performance of our protocol we have carried out five different tests:

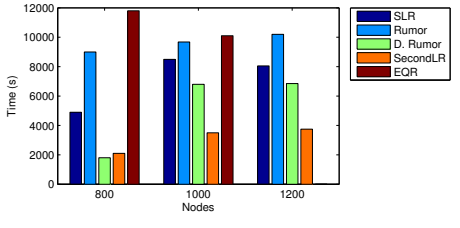
- *Test 1: Time until the first node fails*. This test is one of the indicators of the effectiveness in terms of energy management. In general terms, energy balancing protocols should last longer without failing nodes than others.
- *Test 2: Number of nodes that fail*. This test computes the number of sensors running out of power for each protocol, and therefore provides and indicator of the capacity of the protocols for saving energy.
- *Test 3: Percentage of active neighbors of the station S at the end of the simulation*. This test shows the ability of the routing protocols to keep the station S connected.
- *Test 4: Average energy consumption*. This test provides another indicator of which protocol is better at managing energy.
- *Test 5: Remaining energy level of the neighbors of the source station S*. This test provides a third indicator highlighting which protocol is able to perform better energy balancing among the nodes close to the station.

The results obtained for Tests 1 to 4 are shown in Fig. 5, while the ones for Test 5 are shown in Fig. 6. As it can be observed, in most of the tests EQR outperforms the rest of the routing protocols. In particular, for Tests 1 and 2 and 1200 nodes (Figs. 5(a)

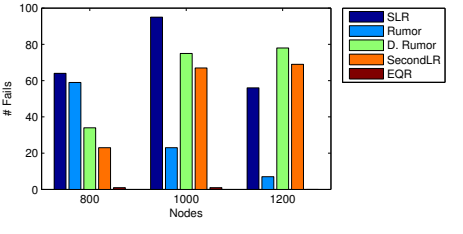
and 5(b)), EQR does not experience any node failure during the whole simulation runtime, while showing better performance than the other protocols for lower number of nodes. In Figs. 5(a) and 5(b), the protocol with the closest performance to EQR is RUMOR. However, the main issue with RUMOR is that when a query is issued, rather than flooding it throughout the network, it is sent on a random walk until the event path is found. As soon as the query discovers the event path, it can be routed directly to the event. Opposed to that, if the path cannot be found, the application can resubmit the query, or, as a last resort, flood the network with it. Therefore, RUMOR could in the best case be seen as an energy saver protocol, but as shown in Fig. 5(d), compared to EQR it lacks functionality for energy balancing.

Figure 5(d) shows that DRUMOR can provide a better energy balance than RUMOR, though this comes at the cost of a relatively poor performance in terms energy saving (see Figs. 5(a) and 5(b) together). One of the main penalties in DRUMOR is that if the length of the query path is too short, it might not reach its destination, so the source could need to flood the network with the query packet again.

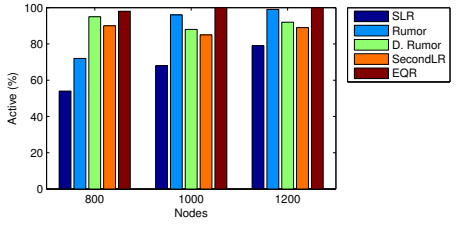
In the case of SLR, the protocol selects the next-hop from a special region. This implies that the nodes in this special region will have higher energy consumption. This is reflected in Fig. 5(b) by a higher amount of node failures, which is also reflected in the decrease suffered in the percentage of active neighbors of the source station in Fig. 5(c). Moreover, SLR selects the next-hop in two steps [5]; the first step targets the selection of a "candidate region", while the second chooses one node from this region



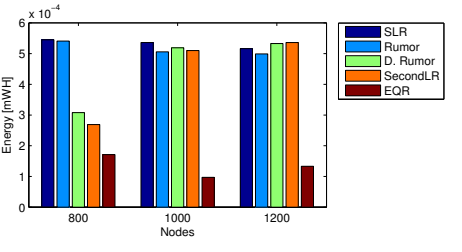
(a) Test 1: Time until first node fails.



(b) Test 2: Number of node failures.



(c) Test 3: Percentage of active neighbors.

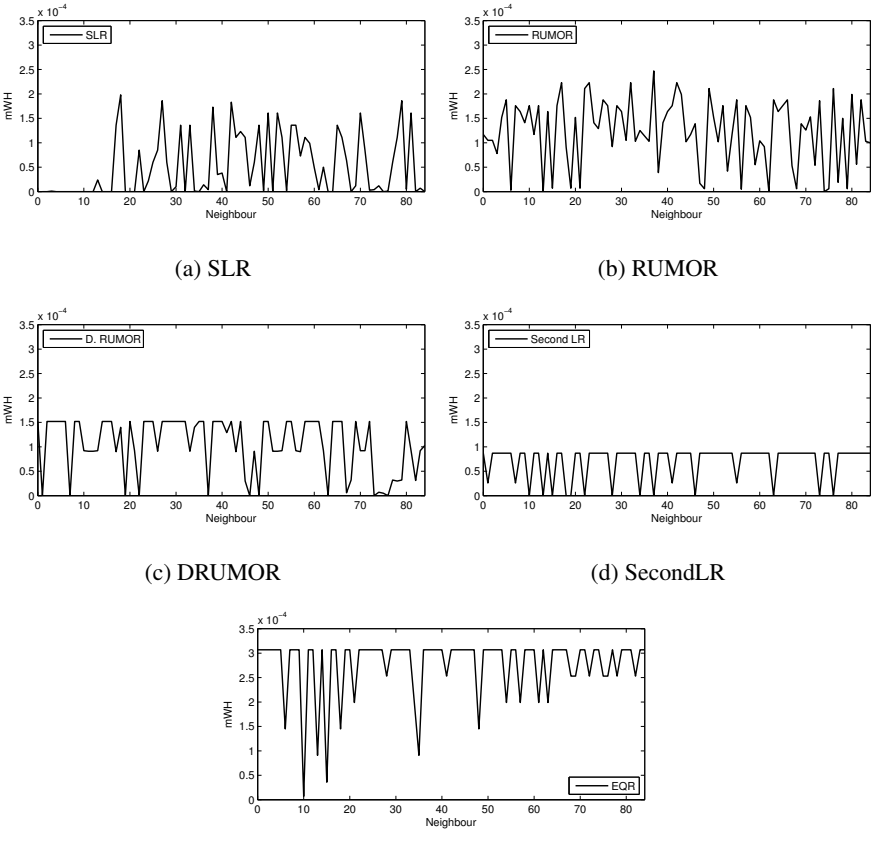


(d) Test 4: Average energy consumption.

Fig. 5. Simulation results for Tests 1-4

as the next-hop. To achieve this, the sender issues a route request and every node in the candidate region sets its own timer T_{wait}. This timer will determine which node could become the next-hop, since when T_{wait} expires the node will issue a message to notify its neighbors. The side effect of this behavior is that all these notification messages—which will be received by the rest of the nodes in the candidate region—are responsible for a higher energy consumption in SLR than in other protocols (see Fig. 5(d)). Indeed, the main idea behind SLR is to keep a straight routing path (supported by the definition of the candidate region), where the longest distance of every hop is less than a half of the transmission radius. As a consequence, if the length of the query path is short, this may lead to a lower hit rate, deriving in a larger number of broadcasts and thus more energy consumption as reflected in Fig. 5(d).

Concerning SecondLR, it can be observed from Fig. 5 that, except for Test1, SecondLR performs in general terms better than DRUMOR, but clearly its performance is much worse than EQR.



(a) SLR

(b) RUMOR

(c) DRUMOR

(d) SecondLR

(e) EQR

Fig. 6. Test 5: Remaining energy levels of the neighbors of the source station

Finally, the results for Test 5 are shown in Fig. 6. The latter reflects the remaining energy level of the neighbors of the source station for all the protocols assessed. As it can be observed, EQR is the most efficient protocol in terms of energy balancing. Even though other results are not reported here, it is worth mentioning that we have found that EQR also provides a good trade-off between latency and energy management. This is an important observation, since strict energy balancing protocols tend to use longer paths which usually lead to higher latencies.

5 Conclusion

Energy management is one of the central concerns in WSNs, and therefore it has been the subject of study of a large number of research works. From the routing perspective, we observe that current destination initiated query-based routing protocols can be considerably improved, especially, if we target a better balance between the energy saving and energy balancing objectives. In this line, we have presented EQR, a routing protocol which in our opinion is a promising candidate to achieve this goal. We have shown that EQR outperforms other (state of the art) destination initiated query-based routing protocols, such as RUMOR [4], SLR [5], DRUMOR [6], and SecondLR [5]. Indeed, five different types of tests were carried out and described in this paper, and in all of them the results obtained with EQR were significantly better. At this stage, the strengths of EQR are mainly limited to relatively static WSN deployments. We plan to extend our work to WSNs exhibiting more mobility among its sensors.

References

1. Akkaya, K., Younis, M.: A survey on routing protocols for wireless sensor networks. Ad Hoc Networks 3, 325–349 (2005)
2. Al-Karaki, J.N., Kamal, A.E.: Routing Techniques in Wireless Sensor Networks: A Survey. IEEE Wireless Communications Journal 11(6), 6–28 (2004)
3. Karl, H., Willig, A.: Protocols and Architectures for Wireless Sensor Networks. John Wiley & Sons, Ltd. (2005) ISBN: 0-470-09510-5
4. Braginsky, D., Estrin, D.: Rumor Routing Algorithm for Sensor Networks. In: Proceedings of the First ACM Workshop on Sensor Networks and Applications, Atlanta, GA, USA (2002)
5. Chou, C.F., Su, J.J., Chen, C.Y.: Straight Line Routing for Wireless Sensor Networks. In: 10th IEEE Symposium on Computers and Communications, Murcia, Spain (2005)
6. Shokrzadeh, H., Haghighat, A.T., Tashtarian, F., Nayebi, A.: Directional Rumor Routing in Wireless Sensor Networks. In: 3rd IEEE/IFIP International Conference in Central Asia on Internet, Tashkent, Uzbekistan (September 2007)
7. Shokrzadeh, H., Haghighat, A.T., Nayebi, A.: New Routing Framework Base on Rumor Routing in Wireless Sensor Networks. Computer Communications Journal 32 (January 2009)
8. Buchli, B., Sutton, F., Beutel, J.: GPS-Equipped Wireless Sensor Network Node for High-Accuracy Positioning Applications. In: Picco, G.P., Heinzelman, W. (eds.) EWSN 2012. LNCS, vol. 7158, pp. 179–195. Springer, Heidelberg (2012)
9. Glomosim Simulator: http://pcl.cs.ucla.edu/projects/glomosim

Performance Evaluation of Reliable Overlay Multicast in Wireless Sensor Networks

Gerald Wagenknecht, Markus Anwander, and Torsten Braun

Institute of Computer Science and Applied Mathematics
Universität Bern, Neubrückstrasse 10, 3012 Bern, Switzerland
{wagen,anwander,braun}@iam.unibe.ch

Abstract. Using multicast communication in Wireless Sensor Networks (WSNs) is an efficient way to disseminate code updates to multiple sensor nodes. For this purpose a multicast protocol has to support bulky traffic (typical traffic pattern for code updates) and end-to-end reliability. In addition, we are interested in energy-efficient operations due to the limited resources of WSNs. Currently no data dissemination scheme fits the requirements mentioned above. Therefore, we proposed the SNOMC (Sensor Node Overlay Multicast) protocol, an overlay multicast protocol, which supports reliable, time-efficient, and energy-efficient data dissemination of bulky data from one sender to many receivers. The protocol's performance in terms of transmission time, number of totally transmitted packets and energy consumption is compared to other often cited data dissemination protocols. Our results show superior performance of SNOMC independent of the underlaying MAC protocol.

1 Introduction

Wireless Sensor Networks (WSN) consist of wireless sensor nodes, which host different applications for the purposes of event detection, localization, tracking, monitoring. An application needs to be configured and continuously updated throughout the lifetime of the network. Such tasks can occur rather often, especially in the deployment phase. There are several challenges to that the configuration and update process. Configuration and updating should be done over the air [1]. Moreover, code update traffic is bulky in nature and has high reliability requirements. Finally, in contrast to the predominant multipoint-to-point communication in WSNs (data retrieval), code updates follow a point-to-multipoint pattern.

Several strategies can be used for data delivery from one sender node to many receivers. The simplest strategy is flooding, where data is transmitted using broadcast communication mechanisms. It is, however, inherently very inefficient, energy consuming, and unreliable. Current broadcast mechanisms for code dissemination such as TinyCubus [2], Deluge [3], and Trickle [4] unfortunately do not include any reliability mechanism. Another strategy is to deploy multiple unicast connections between the sender and any of the desired receivers. In the context of WSNs, however, redundant transmissions lead to higher probability of collisions and increased transmission times. A distribution strategy that can more efficiently meet the requirements of configuration and code updating, is multicast. Multicast is able to propagate data from a single sender

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 114–125, 2012.

to many receivers by affecting a smaller number of sensor nodes in the network. It is easily extendable with any kind of reliability mechanism.

Simply porting an existing IP Multicast solution designed for wired networks to wireless sensor networks is impractical or even impossible. There are three main challenges to that. In contrast to wired networks, resources such as energy, memory, and CPU power are limited in WSNs. Therefore, directly porting an existing IP Multicast solution would require memory and processing power, which a sensor node may lack. Even if resources are available, the lifetime of a node may be severely affected. While in wired networks each node has a dedicated role, nodes in WSNs can take the roles of sender, receiver, forwarding node, and branching node, see Fig. 1. Branching nodes duplicate packets and store state information about receivers and/or about other branching nodes. Forwarding nodes have less or no information about the multicast state and just forward the multicast data from one neighbor to the next one. Wireless communication links are more vulnerable to disruptions than wired links, which raises additional concerns about medium availability, collisions, and reliability.

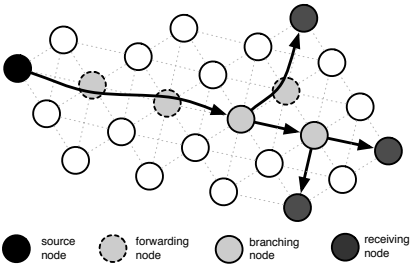


Fig. 1. Roles of the nodes in a multicast scenario

A multicast solution for WSNs needs to address the above-mentioned issues and in particular, reliability to ensure that code updates are disseminated efficiently. Despite many studies on multicast in WSNs most of them focus on multicast routing and not on reliable and efficient data distribution (see Section 2 for more details). Up to our knowledge, there is not a single multicast protocol able to meet the combined set of requirements for reliability and efficiency in both time and energy consumption for bulky traffic patterns. Moreover, we would like the multicast communication to be IP-based in order to access the WSN via the Internet [5].

To fill in the gap we proposed the SNOMC (Sensor Node Overlay Multicast) protocol [6], which supports the reliable transfer of bulk data in a WSN from one sender to multiple receivers. SNOMC has been designed as an overlay multicast protocol on top of the μIP stack from Contiki [7] and offers time- and energy-efficient data distribution and simple NACK-based mechanism for end-to-end reliability. A complete protocol description is provided in [6] while in this paper we are more interested on the protocol performance and how it compares to other proposed solutions.

The paper is organized as follows: Section 2 introduces related work on data distribution schemes as well as on multicast in WSNs. Section 3 describes the SNOMC protocol briefly. Evaluation, including simulation scenario, used protocol stack, and results is presented in Section 4. Section 5 concludes the paper.

2 Related Work

A commonly used data dissemination scheme of low complexity but low efficiency in WSNs is broadcasting. A number of protocols have been proposed to improve the efficiency of broadcasting such as Multipoint Relaying (MPR) [8]. Only a subset of nodes (so called multipoint relays) rebroadcast messages. The relays are chosen based on local knowledge at each node of its two-hop neighborhood. MPR does not support any reliability mechanism. Pump Slowly, Fetch Quickly (PSFQ) [9] is a reliable transport protocol and supports broadcast-based code distribution. It transmits data segments relatively slowly ('pump slowly') and uses an aggressive NACK mechanism to fetch missed data segments ('fetch quickly'). The aggressive NACK mechanism can lead to congestion in the WSN. TinyCubus [2] is an adaptive cross-layer framework for wireless sensor networks. It is also broadcast-based and deploys a role-based code distribution algorithm using cross-layer information such as role assignments to decrease the number of messages needed for code distribution to specific nodes. TinyCubus assumes that roles are assigned before code deployment. Directed Diffusion [10] can be used for both multipoint-to-point and point-to-multipoint communications.

In [11] a multicast protocol called BAM (Branch Aggregation Multicast) is presented. It supports single-hop link-layer multicast and multi-hop multicast by doing branch aggregation. Another multicast protocol for sensor nodes with of node mobility support is VLM^2 (Very Lightweight Mobile Multicast) [12]. VLM^2 provides multicast from a base station to sensor nodes and unicast from sensor nodes to a base station, but it has no reliability mechanisms. In [13] the authors present an effective all-in-one solution for unicasting, anycasting, and multicasting in wireless sensor and mesh networks. RBMulticast [14] is a stateless, receiver-based multicast protocol, which exploits knowledge on geographic node locations to reduce costly state maintenance. The authors of [15] adapt ADMR (Adaptive Demand-driven Multicast Routing), a multicast protocol for mobile ad-hoc networks, on a real wireless sensor node (MICAz). They show that protocol adaptation is not a trivial task and a number of problems have to be solved. At the same time, the authors of [16] analyze IP Multicast and show that it is possible to be used in WSNs. Further, there are several multicast solutions for WSNs based on the geographical sensor node positions [17, 18, 19]. All of these protocols support neither end-to-end reliability nor energy-saving mechanisms.

3 SNOMC Protocol Description

The main requirements towards the protocol we wish to design are multicast support in WSNs, reliable communication for bulky traffic (e.g. code updates), and protocol operation on top of IP. Multicast solutions for WSNs can be classified in different ways. First, depending on the layer of the protocol implementation, there are IP multicast (network layer) and overlay multicast (application layer) solutions. For IP multicast the distribution tree is built between routers in the Internet. For overlay multicast, the tree is built between the end systems. Second, we can distinguish between a sender-driven and a receiver-driven

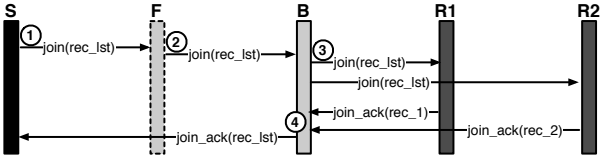


Fig. 2. SNOMC: Joining phase, sender-driven mode

formation of the multicast group. Moreover, different transport protocols (UDP or TCP) can be used, depending on requirements towards reliability support. Last, the network entity where caching occurs, can differ: sender nodes, branching nodes or all intermediate nodes. Detailed description can be found in [6], while here a shorter overview is presented.

To meet our design requirements we developed the Sensor Node Overlay Multicast (SNOMC) protocol, an overlay multicast protocol able to operate in both sender-driven mode and receiver-driven mode. We are using UDP as transport protocol and to ensure reliability we are using a simple NACK-based mechanism with all three caching modes. In the sender-driven mode, the sender decides which nodes should be in the multicast group as receivers. The join procedure is shown in Fig. 2. First, the sender creates a *join* message, which contains the list of receivers, the group id, and the address of the sender. This *join* message is transmitted towards all receivers via intermediate nodes. If an intermediate node can reach all receivers via a single one-hop neighbor it is a forwarding node. Otherwise, the intermediate node becomes a branching node and splits the receivers list into partial lists for the respective next-hop neighbors. Each branching node adds its address into the *join* message. When a *join* message reaches a receiver it confirms its role as receiver by transmitting a *join_ack* message back to the last branching node. The branching node waits for the *join_ack* messages of all subordinate receivers, combines them, puts its address as branching node into the message and transmits it back towards the sender node. Thus, the sender node knows the next branching nodes and each branching node knows its predecessor and its successor.

In the receiver-driven mode, see Fig. 3, the receivers themselves decide whether they want to be a member of the multicast group by sending a *join* message to the sender. A node on the path from receiver to sender decides on its role depending on the number of subordinated receivers, one or many respectively. Branching nodes need to notify the sender by adding their identity to the *join* message. The sender node responds with a *join_ack* message containing the receiver list, its own address as sender_id, and the collected branch_id. Branching nodes are responsible to split the *join_ack* message

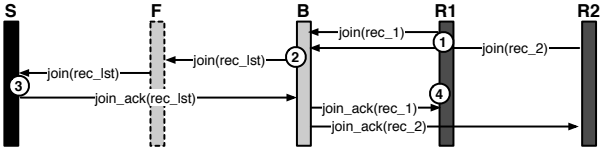


Fig. 3. SNOMC: Joining phase, receiver-driven mode

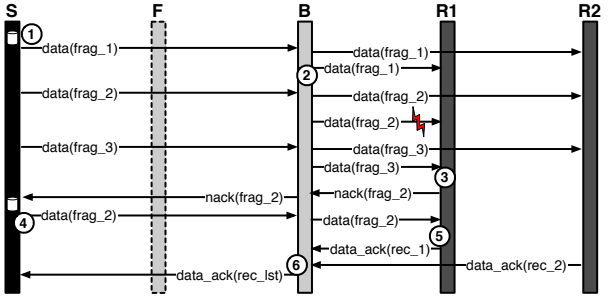


Fig. 4. SNOMC: Transmission phase, caching on sender node

when necessary. Data from sender to receivers are propagated using the overlay network established as result of the join procedure. Overlay links are established between the nodes, which cache the data.

The caching strategy 'caching on sender node' is depicted in Fig. 4. The sender node fragments the data and caches them. Each branching node duplicates and retransmits *data* messages. If a fragment gets lost, the receiver detects a gap in the fragment sequence and requests the missing fragment. This is done using a *nack* message. Since the sender node has the fragment in the cache, it retransmits it towards the requesting receiver. If the receiver gets all fragments successfully, it confirms this with a *data_ack* message. Branching nodes can combine *nack* and *data_ack* messages. Another strategy is when data can be cached additionally at the branching nodes. The cache size is 10 packets. If more packets are coming in the oldest one will be deleted from the cache. The benefit of this additional caching is that a receiver can request data directly from the branching node if it has detected a lost fragment.

In case the data can be cached at every intermediate node the overlay connections change such that every node knows its predecessor and successor in the distribution tree. If a receiver detects a missing fragment, it requests the fragment directly from its predecessor node. If the predecessor node has the fragment cached it retransmits it; otherwise it forwards the *nack* message up the distribution tree until a node is found where the fragment has been cached.

4 SNOMC Evaluation

This section presents the evaluation of the SNOMC protocol. First, we describe the protocol stack. Then, we move on to introduce the different simulation scenarios. Finally, we discuss the results of our measurements.

4.1 Protocol Stack

To evaluate the performance of SNOMC, we compare it to a number of transport protocols commonly found in wireless sensor networks in combination with different underlying MAC protocols. More specifically, these protocols are: Flooding, Multipoint

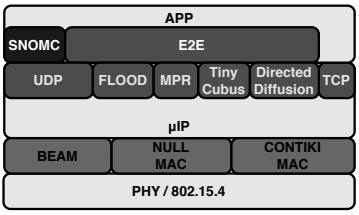


Fig. 5. Simulation protocol stack

Table 1. MAC Protocol Parameters

	acknowledgments	retransmissions	energy-saving
BEAM	positive ack	5	yes
ContikiMAC	early ack	2	yes
NullMAC	no	0	no

Relay, TinyCubus, Directed Diffusion, UDP, and TCP. For the description of the protocols we refer to Section 2. All protocols have been implemented in the OMNeT++ simulator [20]. The protocol stack is shown in Fig. 5 and is based on the μIP stack from Contiki. To enable a fair comparison we had to ensure end-to-end reliability for all protocols and implement the same simple NACK-based reliability mechanism used in SNOMC.

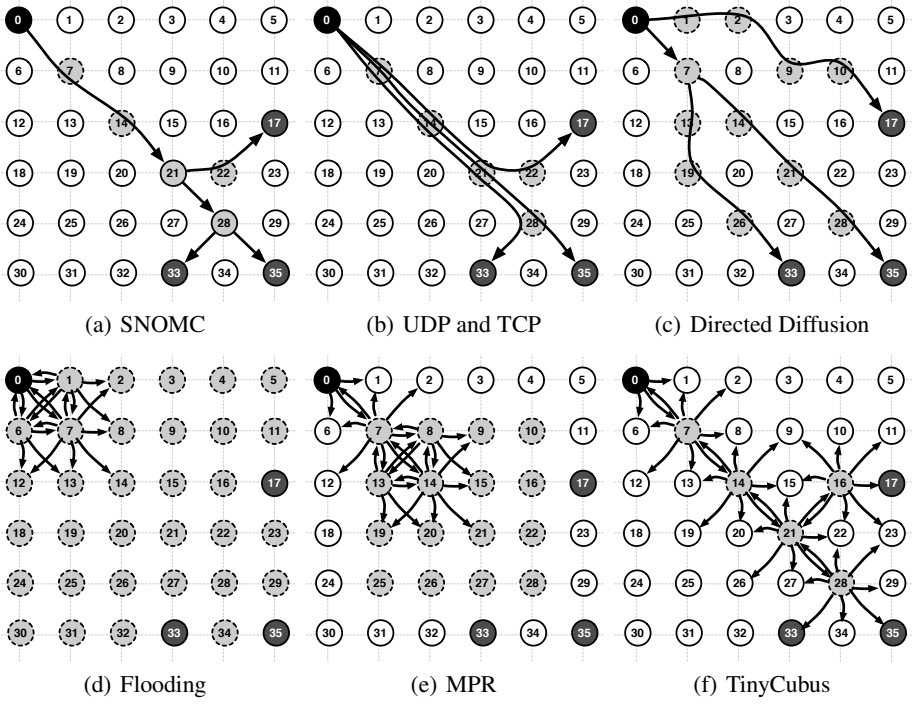
Finally, we compare SMOMC with two unicast-based transport protocols: UDP and TCP. While TCP has a reliability mechanism based on positive acknowledgments, we enhanced UDP with a NACK-based reliability mechanism same as in SNOMC.

The underlying MAC protocol plays an important role for the transmission of data between neighbor nodes. Hence, different MAC protocols can lead to significantly different results, irrespectively of the used transport or multicast protocol. We chose three MAC protocols with different support mechanisms for reliability and energy-efficient operation. The Burst-aware Energy-Efficient Adaptive MAC Protocol (BEAM) [21] uses an adaptive duty cycle mechanism, which reacts quickly to changes in both traffic loads and traffic patterns and ensures hop-to-hop reliability. ContikiMAC [22], which is part of the Contiki operating system, also supports energy-saving radio duty cycling mechanisms and reliability based on an acknowledgment mechanism. NullMAC, also part of Contiki, has no energy-saving mechanisms and does not support reliability. Table 1 shows an overview of the parameter of the used MAC protocols.

4.2 Simulation Scenario

We arranged 36 sensor nodes in a grid of 6x6 nodes with a distance of 100 meters between nodes as shown in Fig. 6. Since we are interested in a multicast scenario, we chose for a sender (node 0) with three receivers (node 17, 33, and 35).

Given the chosen simulation scenario, each of the compared protocols affects a different set of nodes. In the case of SNOMC there are two branching nodes (21, 28) and three forwarding nodes (7, 14, 22) as shown in Fig. 6(a). For UDP and TCP the same



(a) SNOMC (b) UDP and TCP (c) Directed Diffusion

(d) Flooding (e) MPR (f) TinyCubus

Fig. 6. Simulation scenarios

nodes are affected but there are three independent connections (cf. 6(b)). As shown in Fig. 6(c) a different set of nodes participates in the distribution tree in Directed Diffusion, which is a result of the different interest message routing compared to the static routing of SNOMC. Flooding is a radical case where all nodes are affected (cf. in Fig. 6(d)). In the chosen grid scenario the Multipoint Relay protocol calculates a rather high number of multipoint relay nodes (cf. Fig. 6(e)). In the case of TinyCubus, the same set of nodes as in SNOMC is affected. Due to design decisions the protocol does not distinguish between receivers and intermediate forwarders (cf. Fig. 6(f)). Hence, all nodes in the set (7, 14, 21, 22, and 28) will rebroadcast the packets.

We created two evaluation scenarios, which differ in the size of the transmitted messages - 20 bytes and 1000 bytes. The size of 1000 bytes is typically associated with software updates on the sensor nodes; the size of 20 bytes is related to a short configuration message for the sensor nodes. For each scenario, 50 simulation runs are used for evaluation. We measured three parameters: (i) transmission times from the sender to all receivers, (ii) the total number of packets it takes to ensure the successful reception of the data by all receivers, and (iii) the energy consumption of the nodes in the network. The energy consumption is measured according to the CC2420 state machine with real switching times and energy consumption according to [23] and [24] (values for sleeping, receiving, and transmitting) and is calculated per node and per transmitted byte. All parameters are measured only taking into account the data distribution phase. Any initial phases are not considered.

Table 2. Simulation Parameters

carrierFrequency	bit-rate	sensitivity	thermalNoise	TX power	modulation
2.4E+9 Hz	250 kbps	-94 dBm	-110 dBm	1mW	O-QPSK

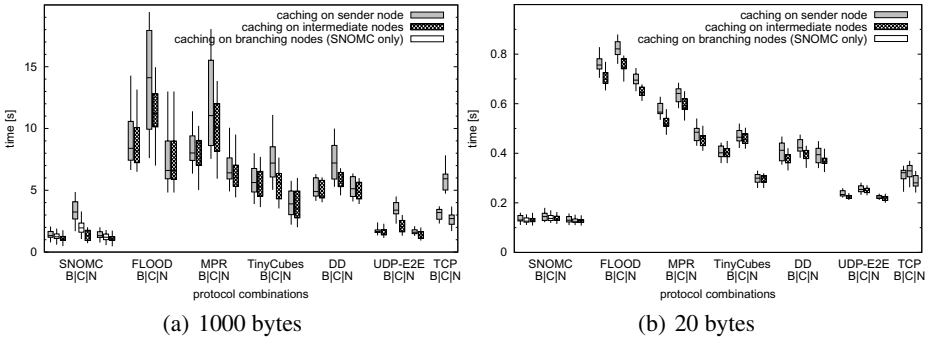
In order to get realistic results a radio model is implemented according to the CC2420 manual [23] and the Castalia Simulator [24]. It is used to calculate the signal to noise ratio (SNR) based on parameters shown in Table 2. Using the SNR and real measurements with a CC2420 radio transceiver the bit error rate (BER) is calculated. In addition, a normally distributed packet error rate of 5% is assumed to represent random noise and external interferences.

4.3 Results on Time Consumption

In this section, we present our findings transmission times. In all figures on the x-axis the combinations of transport and MAC protocols are shown. In our notation **B** stands for BEAM, **C** for ContikiMAC, and **N** for NullMAC.

First, in Fig. 7(a), we discuss results for the time required to transmit 1000 bytes from the sender node to the three receiver nodes. As expected, UDP-E2E require more time due to the redundant unicast flows that need to be established for each of the three receiver nodes. TCP performs even worse since every packet has to be acknowledged. This cross traffic caused by simultaneous data and acknowledgments increases collision probability and hence affects delay negatively. Flooding, Multipoint Relay, and TinyCubus are all broadcast protocols and are much worse in performance. On the one hand, broadcasting affects usually more nodes, which leads to a higher number of transmissions. Consequently, the probability of collisions increases and more retransmissions are necessary pushing delay up. On the other hand, to avoid collisions higher random back-off time, compared to unicast-based, are necessary. This, however, would also lead to longer transmission times. SNOMC requires the lowest time to transmit the 1000 bytes to the receivers.

If we now compare the performance in combination with the MAC protocol, we see in Fig. 7(a) that BEAM has a little lower performance, considering the time needed to



(a) 1000 bytes (b) 20 bytes

Fig. 7. Transmission time

deliver 1000 bytes, than NullMAC using SNOMC, Directed Diffusion, UDP and TCP. However, BEAM outperforms ContikiMAC, which is the result of two factors. First, BEAM is optimized for bulky traffic while ContikiMAC focuses only on constant (or slowly changing) traffic. Second, BEAM has a better congestion control and better duty cycle mechanism. The latter is also the reason why caching at intermediate nodes affects the performance with both protocols differently, i.e., BEAM has much smaller effects. Together with broadcast-based protocols, NullMAC works better than BEAM. Since BEAM has energy-saving mechanisms, the radio transceiver can be in sleep mode. If so, longer time is needed to transmit a packet from sender to receiver. On the contrary, the radio transceiver in NullMAC is always on and therefore the sender can immediately transmit the packet.

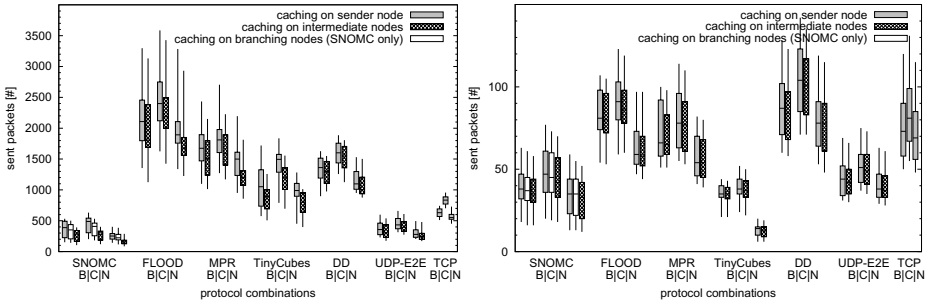
In case of 20 bytes of data a single packet has to be transmitted. Transmission times are shown in Fig. 7(b). We see that SNOMC achieves the best performance. In an ideal case TinyCubus would be better since it requires a smaller number of transmissions compared to SNOMC (due to using broadcast transmissions), but SNOMC has the added benefit of smaller random-back off times. UDP and TCP need more transmissions due to the three independent flows, hence the longer transmission times. Although TCP requires acknowledgments for each packet, in our scenario there will be a single acknowledgment only (due to only one data packet), explaining the much smaller differences between TCP and UDP compared to the scenario with 1000 bytes.

Further, in SNOMC, TinyCubus, Directed Diffusion and UDP collisions among data and acknowledgments generally do not occur and hence no retransmissions are required. Therefore, the corresponding boxplots in Fig. 7(b) are quite compact and have no big outliers. Finally, the differences between Flooding, Multipoint Relay and Directed Diffusion are similar to the 1000 bytes scenario.

4.4 Results on Number of Transmissions

Fig. 8(a) shows the number of total transmissions needed for the successful transfer of 1000 bytes. As we can see, SNOMC requires the fewest number of packets, followed by UDP, TCP and Directed Diffusion. The results of broadcast-based protocols (Flooding, Multipoint Relay, and TinyCubus) are considerably worse and are compliant with our observations on transmission times. Flooding requires most transmissions (inherent to its communication style), followed by Multipoint relay (result of the disadvantageous set of multipoint relays) and, with the best performance of the three, TinyCubus. Looking at the MAC protocols, BEAM requires more packets to ensure hop-to-hop reliability than NullMAC, irrespectively of the transport protocol. This is due to the fact that the receiver can be in sleeping mode and multiple attempts may be required before the packet is transmitted successfully. Using ContikiMAC always needs more packet retransmissions on link layer due to a worse duty cycle mechanism and thus higher number of necessary end-to-end retransmissions on the transport layer.

The results for the transmission of 20 bytes are shown in Fig. 8(b). TinyCubus achieves the best performance in combination with NullMAC. It has an optimal set of forwarding nodes and NullMAC keeps the radio transceiver always awake. Hence, this combination reaches the minimal number of packets (6) to ensure the successful transmission of 20 bytes. The other broadcast protocols (Flooding and Multipoint Relay)



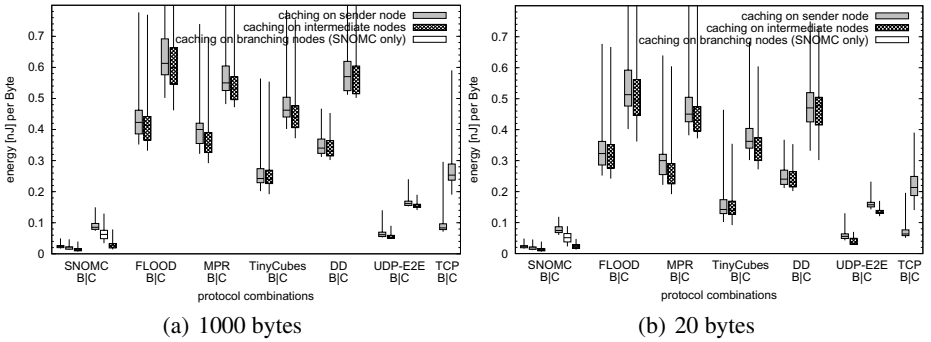
(a) 1000 bytes (b) 20 bytes

Fig. 8. Number of transmitted packets

show an improved performance as well, considering the scenario with 1000 bytes. Just one packet has to be transmitted, which causes less collisions and retransmissions. Out of all non-broadcast-based protocols, SNOMC has the best performance while Directed Diffusion has the worst. Directed Diffusion requires a larger number of packets because of the more extensive hop connectivity, i.e., more connections compared to UDP and TCP. Further, the differences between the three caching modes are quite small; a result of the smaller number of required retransmissions.

4.5 Results on Energy Consumption

We now move on to discuss the energy consumed per node and per transmitted byte. Results for the scenario with 1000 bytes are shown in Fig. 9(a). We compare only the performance with BEAM and ContikiMAC, since NullMAC does not have an energy saving mechanism. In general, it can be seen that for broadcast-based protocols there is a stronger relation between the consumed energy and the number of transmitted bytes. More specifically, the more bytes are transmitted the higher is the energy consumption per byte due to higher packet loss. The energy consumption of unicast-based protocols is generally good with the exception of Directed Diffusion, which performs rather poor



(a) 1000 bytes (b) 20 bytes

Fig. 9. Energy consumption per node and per transmitted byte

due to additional maintenance messages, e.g., for path reinforcement or the propagation of new interest. Concerning the impact of the MAC protocol, BEAM offers higher energy-efficiency than ContikiMAC since the latter has a worse duty cycle mechanism. Furthermore, caching does not significantly influence the performance of broadcast-based protocols from an energy point of view.

In Fig. 9(b) corresponding results on energy consumption are shown for the scenario with a single packet (20 bytes). As we can expect, energy consumption per transmitted byte is lower due to the smaller size of data to transmit; the transmission of 20 bytes implies a lower number of collisions and retransmissions.

5 Conclusions

We propose the Sensor Node Overlay Multicast (SNOMC) protocol to support a reliable, time-efficient and energy-efficient dissemination of bulky data from one sender node to many receivers. To ensure end-to-end reliability we designed and implemented a NACK-based reliability mechanism. Further, to avoid costly end-to-end retransmissions we propose different caching strategies implemented in SNOMC.

We compared the SNOMC protocol to other common protocols for data dissemination in terms of transmission time, number of transmitted packets and energy consumption. In general, SNOMC outperforms the other protocols. Further, we showed that our protocol performs well with different MAC protocols, which support different levels of reliability and energy-efficiency. We therefore conclude that SNOMC can offer a robust, high-performing solution for the efficient distribution of code updates in a WSN.

References

1. Wagenknecht, G., Anwander, M., Braun, T., Staub, T., Matheka, J., Morgenthaler, S.: MARWIS: A Management Architecture for Heterogeneous Wireless Sensor Networks. In: Harju, J., Heijenk, G., Langendörfer, P., Siris, V.A. (eds.) WWIC 2008. LNCS, vol. 5031, pp. 177–188. Springer, Heidelberg (2008)
2. Marron, P.J., Lachenmann, A., Minder, D., Hähner, J., Sauter, R., Rothermel, K.: TinyCubus: A Flexible and Adaptive Framework for Sensor Networks. In: EWSN 2005, Istanbul, Turkey (February 2005)
3. Hui, J.W., Culler, D.: The Dynamic Behavior of a Data Dissemination Protocol for Network Programming at Scale. In: SenSys 2004, Baltimore, MD, USA (November 2004)
4. Levis, P., Patel, N., Shenker, S., Culler, D.: Trickle: A Self-Regulating Algorithm for Code Propagation and Maintenance in Wireless Sensor Networks. In: NSDI 2004, San Francisco, CA, USA (March 2004)
5. Anwander, M., Wagenknecht, G., Braun, T.: Management of Wireless Sensor Networks using TCP/IP. In: IWSNE 2008, Santorini Island, Greece (June 2008)
6. Wagenknecht, G., Anwander, M., Braun, T.: SNOMC: An Overlay Multicast Protocol for Wireless Sensor Networks. In: WONS 2012, Courmayeur, Italy (January 2012)
7. Dunkels, A., Grönvall, B., Voigt, T.: Contiki - a Lightweight and Flexible Operating System for Tiny Networked Sensors. In: EmNetS 2004, Tampa, FL, USA (November 2004)
8. Quayyum, A., Viennot, L., Laouiti, A.: Multipoint relaying: An Efficient Technique for Flooding in Mobile Wireless Networks. TR, INRIA, Sophia Antipolis, France (2000)

9. Wan, C.Y., Campbell, A.T., Krishnamurthy, L.: PSFQ: A Reliable Transport Protocol for Wireless Sensor Networks. In: WSNA 2002, Atlanta, GA, USA (September 2002)
10. Intanagonwiwat, C., Govindan, R., Estrin, D., Heidemann, J., Silva, F.: Directed Diffusion for Wireless Sensor Networking. ACM/IEEE Transactions on Networking 11(1), 2–16 (2002)
11. Okura, A., Ihara, T., Miura, A.: BAM: Branch Aggregation Multicast for Wireless Sensor Networks. In: MASS 2005, Washington, DC, USA (November 2005)
12. Sheth, A., Shucker, B., Han, R.: VLM2: A Very Lightweight Mobile Multicast System For Wireless Sensor Networks. In: WCNC 2003, New Orleans, LA, USA (March 2003)
13. Flury, R., Wattenhofer, R.: Routing, Anycast, and Multicast for Mesh and Sensor Networks. In: INFOCOM 2007, Anchorage, Alaska, USA (May 2007)
14. Feng, C.H., Heinzelman, W.B.: RBMulticast: Receiver Based Multicast for Wireless Sensor Networks. In: WCNC 2009, Budapest, Hungary (April 2009)
15. Chen, B., Muniswamy-Reddy, K., Welsh, M.: Ad-Hoc Multicast Routing on Resource-Limited Sensor Nodes. In: REALMAN 2006, Florence, Italy (May 2006)
16. Silva, J.S., Camilo, T., Pinto, P., Ruivo, R., Rodrigues, A., Gaudncio, F., Boavida, F.: Multicast and IP Multicast Support in Wireless Sensor Networks. Journal of Networks 3(3), 19–26 (2008)
17. Koutsonikolas, D., Das, S., Hu, Y.C., Stojmenovic, I.: Hierarchical Geographic Multicast Routing for Wireless Sensor Networks. In: SENSORCOMM 2007, Valencia, Spain (October 2007)
18. Sanchez, J.A., Ruiz, P.M., Stojmenovic, I.: Energy Efficient Geographic Multicast Routing for Sensor and Actuator Networks. Computer Communications 30(13), 2519–2531 (June 2007)
19. Lee, J., Lee, E., Park, S., Oh, S., Kim, S.H.: Consecutive Geographic Multicasting Protocol in Large-Scale Wireless Sensor Networks. In: PIMRC 2010, Istanbul, Turkey (September 2010)
20. OMNeT++: Discrete Event Simulation System, http://www.omnetpp.org
21. Anwander, M., Wagenknecht, G., Braun, T., Dolfus, K.: BEAM: A Burst-Aware Energy-Efficient Adaptive MAC Protocol for Wireless Sensor Networks. In: INSS 2010, Kassel, Germany (June 2010)
22. Dunkels, A., Mottola, L., Tsiftes, N., Österlind, F., Eriksson, J., Finne, N.: The Announcement Layer: Beacon Coordination for the Sensornet Stack. In: Marrón, P.J., Whitehouse, K. (eds.) EWSN 2011. LNCS, vol. 6567, pp. 211–226. Springer, Heidelberg (2011)
23. CC2420: Datasheet for the Chipcon CC2420 RF Transceiver, Online (December 2011)
24. Pham, H.N., Pediaditakis, D., Boulis, A.: From Simulation to Real Deployments in WSN and Back. In: WoWMoM 2007, Helsinki, Finland (June 2007)

Experimental Comparison of Bluetooth and WiFi Signal Propagation for Indoor Localisation

Desislava C. Dimitrova, Islam Alyafawi, and Torsten Braun

University of Bern, Switzerland
{dimitrova,alyafawi,braun}@iam.unibe.ch

Abstract. Systems for indoor positioning using radio technologies are largely studied due to their convenience and the market opportunities they offer. The positioning algorithms typically derive geographic coordinates from observed radio signals and hence good understanding of the indoor radio channel is required. In this paper we investigate several factors that affect signal propagation indoors for both Bluetooth and WiFi. Our goal is to investigate which factors can be disregarded and which should be considered in the development of a positioning algorithm. Our results show that technical factors such as device characteristics have smaller impact on the signal than multipath propagation. Moreover, we show that propagation conditions differ in each direction. We also noticed that WiFi and Bluetooth, despite operating in the same radio band, do not at all times exhibit the same behaviour.

1 Introduction

Positioning of people and resources has always been a necessity for society throughout human history. Indoor environments, however, still pose a challenge to the localisation paradigm and foster vigorous research by both academia and industry. Indoor spaces are typically characterised by restricted dimensions and multiple structure elements such as walls, doors, furniture. As a result, radio signals have stronger multipath components compared to outdoor scenarios. Moving human bodies are an additional complication. The combined effect of these factors challenges the pervasive application of a single positioning solution. While some authors, e.g., [8,18], try to find a solution based on a single wireless technology, others, e.g., [9,19], propose to combine multiple technologies. Still, the optimal choice of technology and localisation technique depends on the application requirements towards accuracy, cost and ease of deployment.

In the scope of the Location Based Analyser (LBA) project[1] we are interested in a positioning solution that is easy to deploy, is low-cost and scales well with the size of the indoor area. The application targets the support of Location Based Services (LBS) and statistical profiling for enterprises such as exposition centres, shopping malls or hospitals. We are interested in providing precision up

[1] An Eureka Eurostars project no. 5533, funded by the Swiss Federal Office for Professional Education and Technology and the European Community.

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 126–137, 2012.

to few meters in order to support variety of applications with different accuracy requirements. Furthermore, the positioning mechanism should be non-intrusive because we want to avoid placing dedicated software in the tracked devices and hence we cannot rely on their cooperation. Given these requirements, we decided to base the positioning mechanism on a radio technology such as Bluetooth or IEEE 802.11 (with the trade name WiFi). These technologies benefit from large support by personal devices and the radio signals being freely available.

As many other studies using similar approaches we stumbled upon the challenges of indoor signal propagation and its implications for a localisation system. Despite the large number of studies addressing radio-based indoor positioning, only few actually investigate the various factors that impact the localisation system. There are plenty of studies [2], proposing a novel propagation model but results are often not convincing or the model performs well only in a particular setting. Other studies take a more practical approach where propagation conditions are monitored in order to adapt the localisation scheme. For example, in some fingerprinting solutions one out of several radio maps is selected depending on periodically updated readings on humidity or temperature. Often, however, only a couple, if not a single factor is observed. To our knowledge, a detailed study, covering several factors and reflecting their impact on both Bluetooth and WiFi signals has not been conducted so far.

With this paper we aim to extend the state-of-the-art by investigating the impact of (1) device's technical characteristics, (2) manufacturing discrepancies and (3) device orientation. Without being exhaustive, we try to gain insights on the complex effects of each factor and the implications for indoor positioning. Our purpose is to identify which factors should be considered and which can be disregarded in the design of a positioning algorithm. The paper is, however, not concerned with the development or testing of such an algorithm.

The rest of the paper is structured as follows. In Section 2 we briefly summarise advances in indoor localisation and in radio-based solutions in particular. The following Section 3 introduces our monitoring system and the testing environment. Evaluation results are presented in Section 4. Finally, in Section 5 we draw conclusions and identify open discussion topics.

2 Indoor Localisation

Multiple technologies have been proposed to tackle the problem of indoor localisation some examples being infrared [22], ultrasound [16] and Radio Frequency IDentification [5]. Still, most research is dedicated to the usability of two technologies. Large number of papers, e.g., [10] and [23], argue that Ultra Wide Band (UWB) radio offers excellent means to determine one's location with high precision. Unfortunately, UWB-based solutions have longer deployment time and are expensive. Equally many studies campaign for the use of IEEE 802.11, e.g., [7,12], or Bluetooth, e.g., [14,19] since their ubiquitous support by personal devices is convenient for the quick, cost-efficient development of practical solutions.

2.1 Radio-Based Localisation

A radio-frequency technology can provide feedback on multiple parameters related to signal reception, which can be used for localisation. Some localisation mechanisms, see [11,17], use the Received Signal Strength Indicator (RSSI), which is derived from the received signal strength and should be therefore directly related to distance. Unfortunately, RSSI measurements are vulnerable to the strong multipath effects indoors. Other mechanisms, see [7,20], base the location estimate on Time of Arrival (TOA) or Time Difference of Arrival (TDOA) parameters. This approach, although more accurate, comes at a higher cost and requires intervention at the target devices. In [3] the Response Rate (RR) of a Bluetooth inquiry is introduced as the percentage of inquiry responses out of the total inquiries in a given observation window. The authors claim to achieve good positioning accuracy. We remain sceptical on the use of RR alone due to its vulnerability to the Bluetooth channel hopping and WiFi contention.

For our purposes we believe that the RSSI parameter is fitting. RSSI measurements are readily available and still can deliver satisfying accuracy, given that appropriate processing is applied. We should, however, account for the impact of radio propagation conditions on the RSSI values.

2.2 Radio Signal Propagation

Generally, radio signals are shaped by the transmitter, receiver and propagation environment. The transmitter and receiver affect the signal by their technical characteristics while the propagation channel's effects are related to path loss due to the propagation medium and any obstacles on the propagation path. Indoor environments make the reconstruction of signals more difficult due to their smaller dimensions and the significantly bigger number of obstacles on the signal path. These obstacles can be part of the indoor construction, e.g., walls and doors, as well as individual objects such as furniture and people. As a result, shadowing and multipath propagation exhibit strongly and multiple copies of the same signal, travelling over several paths. The signal reconstructed at the receiver is formed by all individual paths and is more difficult to relate to the actual distance between the nodes.

Characterising the indoor radio channel has been an active research area dating back to the early '90s, e.g., [13]. There are many works, such as [1,2,6,21], which study the radio channel in general and investigate the path loss distribution over distance or for different propagation scenarios, including line-of-sight or non-line-of-sight. Studies focusing on radio-based indoor positioning [4], examine the specific effects of the above factors - distance and obstacles - on radio signal parameters used for positioning. Other factors such as technical characteristics or orientation are also important but rarely studied in detail. To fill in the gap we investigate how a radio signal is affected:

- at the transmitter side by the technical specifications of different manufacturers and even models of the same type of device;

– at the receiver side by manufacturing discrepancies occurring during the production process;
– during propagation by the propagation path that a signal takes;
– by type of radio technology - Bluetooth or WiFi.

3 Monitoring Approach

Technology. In order to observe the impact of various factors on the received signal we deployed sensor nodes, which can scan for transmissions on two interfaces - one for Bluetooth and one for IEEE 802.11b/g.

In the context of Bluetooth we rely on the inquiry procedure, introduced in the Bluetooth's Core Specification 4 [15]. For an inquiry to be successful a Bluetooth device should only be discoverable. We prefer to work with the inquiry procedure due to several advantages. First, the RSSI reported by an inquiry procedure is not affected by power control and hence can be directly related to distance. Second, although long lasting - the inquirer needs to check all 32 Bluetooth radio channels - an inquiry procedure can monitor a large number of target devices. Last, we can gather measurements without requesting any privacy-sensitive information from the mobile devices.

In the context of WiFi the sensor nodes overhear WiFi signals from the target devices. Contrary to Bluetooth, there is no inquiry procedure defined in WiFi. A mobile device becomes visible only after it sends out a request to associate to an access point. In the associated state there is a periodic exchange of control messages. By overhearing these messages, or any potential data messages, a scanning sensor node can derive information on RSSI levels.

Test-Bed. All experiments were set up in an indoor office with dimensions 6.90x5.50x2.60m. A schematic is shown in Figure 1. The office is equipped with desks, chairs and desktop machines. The sensor nodes (SNs) and mobile devices (MDs) hang at 0.50m below the ceiling and are at 1.50m above the tables. Such test environment allows us to judge the relevance of the tested factors for a positioning system under realistic propagation conditions.

Metrics. Our first challenge was to select the appropriate metric to compare performance. We considered four groups of metrics to characterise the RSSI, namely, instantaneous values, probability density function, mean and standard deviation, median and percentiles; as well as the response rate of a scan.

4 Evaluation

Below we evaluate the impact of each of the three factors: technical characteristics, manufacturing discrepancies and direction-specific multipath propagation. During the measurements collection in all experiments no humans were present in the test-bed area.

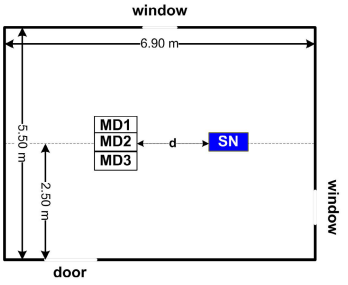


Fig. 1. Experiment A: Set-up

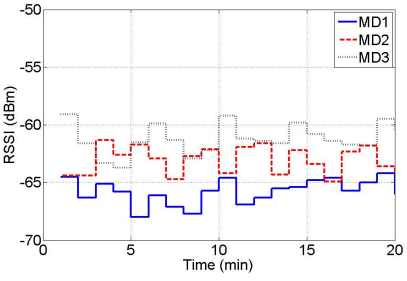


Fig. 2. RSSI time variation of three MDs

4.1 Technical Characteristics

The transmit power of a personal device is a result of propagation conditions and technical specifications but also of manufacturer preferences. Differences between manufacturers, or even between different models of the same manufacturer, could additionally (on top of multipath effects) aggravate the problem of localisation. In order to investigate how such differences affect the RSSI we performed *Experiment A*. The test set-up is shown in Figure 1. Three mobile phones by different manufacturers were placed at one and three meters away from the same sensor node. At each distance, measurements are gathered for 30 minutes, which allowed us to collect about 200 samples for WiFi and 400 for Bluetooth.

The choice of evaluation approach should be made carefully. By placing the mobile phones next to each other we try to minimise the spatial and temporal difference in their propagation paths. Yet, this rises some concerns on interference between the phones, which could be avoided by doing independent measurements. The latter approach, however, catches different temporal states of the propagation channel. Furthermore, we can choose between measuring (i) the transmitted signal at the antenna, which allows to isolate the impact of the propagation environment or (ii) the received signal, which is affected by the multipath propagation but shows how a real system sees different mobile phones. Since we are interested to develop an operational localisation system we looked at the second.

Instantaneous RSSI. Figure 2 shows the changes in time of the *instantaneous RSSI* of a Bluetooth signal at distance one meter. With instantaneous RSSI we refer to a single momentary RSSI value. The strong variations of the RSSI show that this metric is much affected by multipath propagation. Therefore, relying on instantaneous RSSIs for localisation can be misleading.

A better analysis would be based on metrics that can (partially) eliminate the impact of multipath propagation. The latter causes temporal, unpredictable RSSI variations. Evaluating a set of samples rather than a single value can isolate temporal changes and provide a more distinct main trend. We discuss the appropriate metrics in the coming three sections.

Probability Density Function. The *probability density function* (PDF) of the RSSI, constructed for each combination of mobile device and distance, is shown in Figure 3(a) for Bluetooth and in Figure 3(b) for WiFi. On the x-axis of a graph we plot the RSSI values whereas the y-axis plots the PDF.

Although the PDF shapes are similar for the three mobile devices, the maximum RSSI value is not the same, suggesting that the impact of the technical characteristics of the device should not be underestimated. Further, as it can be expected, RSSI values are lower at three meters due to larger path loss. Also, we notice that at distance one meter (upper row) the graphs are more compact whereas at three meters (lower row) the PDFs are generally wider, i.e., the set of observed RSSI values is larger. This can be explained by the stronger effect of multipath propagation as distance increases. Another consequence of multipath propagation is the slight asymmetry of the PDF with longer tail towards lower RSSI values.

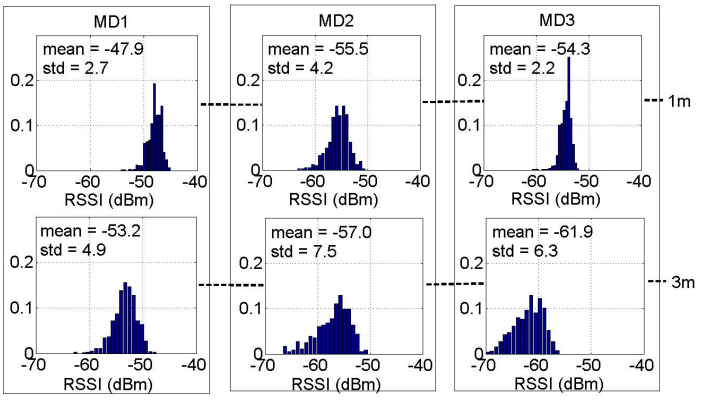
Differences between Bluetooth and WiFi are minor: WiFi signals have by default higher transmit power and subsequently stronger multipath components, which causes higher deviation of the RSSI signals, i.e., broader PDF shape. This is also the reason for the generally weaker received Bluetooth signals. For MD2 we could not identify the reasons for the little effect of distance on its WiFi signal.

Median and Percentiles. An alternative to a PDF representation is a *boxplot*, which depicts a population's median, lower and upper quantiles, minimum and maximum, and outlier samples. Using boxplots makes it easier to identify the main concentration of the RSSI values and how much the RSSI deviates. Another advantage of a boxplot is that outliers are visible; they are difficult to spot in a PDF due to their low probability.

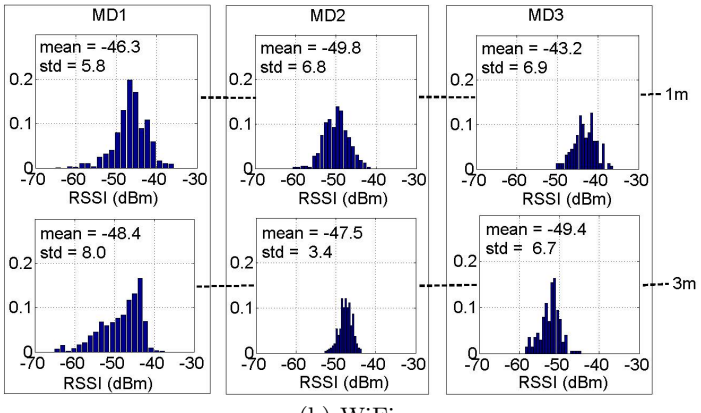
The boxplots corresponding to the PDF curves for both Bluetooth and WiFi are shown in Figure 4. Along with differences in the behaviour of mobile phones, we can directly observe a much larger deviation of RSSI values at three meters than at one meter. We also observe that WiFi signals are less robust to deviation than Bluetooth signals.

Mean and Standard Deviation. Although PDFs and boxplots are very descriptive, they require the collection of many samples (corresponding to long observation periods). Their use in a real-time positioning system, where samples are evaluated every few seconds, is challenging. An easier to derive set of metrics is the *mean* and *standard deviation*. The corresponding metrics for each PDF graph in Figure 3 are shown in the upper left corner.

We note that the mean is often off-set at 1-2dBm from the median, see Figure 4. These differences are caused by the asymmetry in the PDF distribution - the mean and standard deviation take into account all samples, including outliers, while the median excludes them. All other observations are consistent with previously made ones.

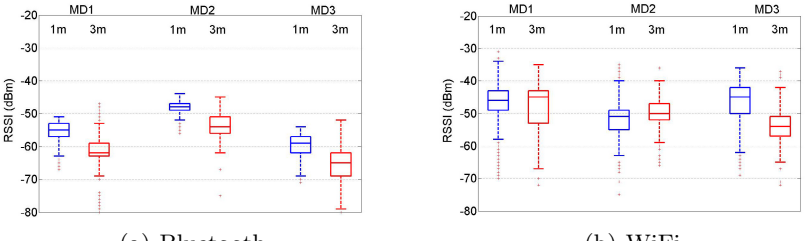


(a) Bluetooth



(b) WiFi

Fig. 3. PDF of the RSSI levels for three mobile devices measured at the same sensor node; distances one and three meters



(a) Bluetooth (b) WiFi

Fig. 4. Boxplots of three MDs, RSSIs measured by the same sensor node; distances one and three meters

Table 1. Experiment 1: Response Rates

	Bluetooth			WiFi		
	MD1	MD2	MD3	MD1	MD2	MD3
1 m	12.9	6.8	11.3	23.3	12.9	14.2
3 m	13.6	6.0	10.8	18.8	5.2	5.1

Table 2. Experiment 2: Response Rates

	WiFi			Bluetooth		
	SN1	SN2	SN3	SN1	SN2	SN3
1 m	20.6	16.2	19.7	43.4	30.1	41.0
3 m	15.4	16.8	16.3	37.8	20.3	55.5

Response Rate. While RSSI-related metrics are vulnerable to multipath propagation, the *response rate* (RR) of a device is not and has potential for localisation. The response rate is defined as the average number of times per minute that a device (i) responded to an inquiry procedure in Bluetooth or (ii) was overheard in WiFi. By comparing the RRs of the same device at several anchor nodes one can derive conclusions on the devices location.

Results for the RR of both Bluetooth and WiFi for all studied scenarios are shown in Table 1. We see that the RR of Bluetooth varies in an incoherent way making it difficult to relate it to distance. Frequency hopping in Bluetooth causes the RR to depend on channel synchronisation and obstructs its use for positioning. No such discrepancies are observed in the case of WiFi, where the RR is a function of the distance. Although values among devices differ, the changes in RR in distance are consistent.

Concluding Remarks. In terms of evaluation metrics we conclude that the choice of metric depends on the time granularity needed by the localisation algorithm. Probability density functions and boxplots are more representative but they also require the collection of many RSSI samples. They are better used in positioning applications whose main purpose is the collection of long-term statistics. When a quick evaluation is desired, e.g., as in real-time systems, the mean of a group of samples is more convenient to handle. In all cases using a single instantaneous RSSI value is not recommended.

In terms of performance we conclude that mobile devices show significant difference in performance. This fact should be considered in the development of a localisation algorithm. One possible approach to compensate for these differences is to relate a device's measurements from several scanning nodes.

4.2 Experiment B: Manufacturing Discrepancies

In order to observe the impact of manufacturing discrepancies on signal reception we designed *Experiment B*. We placed three sensor nodes of the same manufacturer and model (Gumstix Overo Fire) but different manufacturing runs according to the experiment set-up in Figure 5. The sensor nodes are at virtually the same spot (sensor's dimensions cause some displacement) at distance one and three meters of a mobile device. At each distance measurements were collected during 30 minutes. Based on the conclusions of Section 4.1 we selected as evaluation metrics the median and percentiles (depicted by a boxplot diagram) and the response rate.

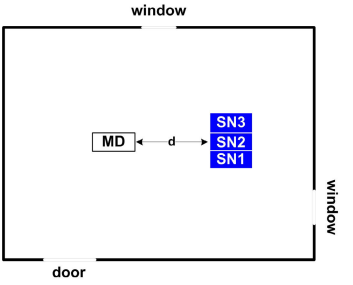
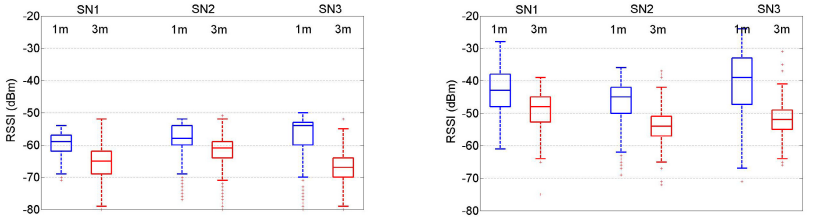


Fig. 5. Scenario B: set-up



(a) Bluetooth (b) WiFi

Fig. 6. Boxplots of three SNs measuring the RSSI values of the same mobile device; distances one and three meters

The boxplots in the case of Bluetooth signals are shown in Figure 6(a). The median of different sensor nodes changes in the order of 2-3dBm. This is much less than the 10-15dBm registered by different mobile devices in Figure 4(a); the RSSI deviation for sensor nodes is also lower. In the case of WiFi, see Figure 6(b), the differences between the median values of sensors increases to 5-6dBm coming close to the results for device specifics of Section 4.1. Other observations on the RSSI deviation and behaviour of Bluetooth and WiFi signals, already made in Section 4.1, continue to hold.

The response rate RR of both Bluetooth and WiFi signals calculated at each sensor node is shown in Table 2. Two observations are worth noting. First, the RR of different sensors is similar, given the same technology and distance. This leads us to believe that manufacturing tolerances have little impact on the response rate. Second, the RR is difficult to relate to distance for Bluetooth signals but can be helpful in WiFi.

Concluding Remarks. Given that measurements were made in a realistic environment and not a well controlled one, it is difficult to pinpoint the cause of the RSSI degradation to only manufacturing tolerances or only multipath propagation. Still, we can observe that for the same propagation environment, although at different time instants, manufacturing tolerances seem to show smaller impact on the RSSI than device characteristics. Therefore, we claim that in the development of an indoor localisation system the designer can assume that all receiving devices have the same behaviour, given they are from the same model.

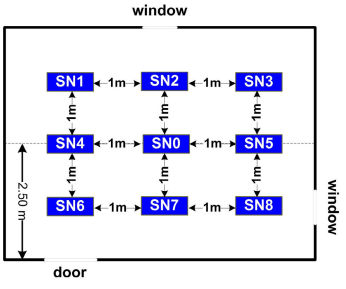


Fig. 7. Scenario C: set-up

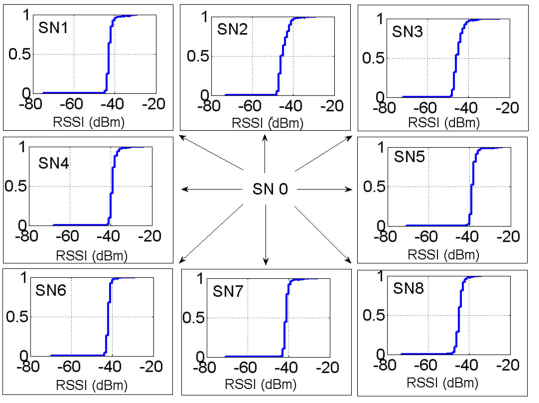


Fig. 8. CDFs in eight communication directions

4.3 Experiment C: Propagation Paths

Depending on the locations of sender and receiver the signal between the two traverses propagation paths of different length through different obstacles. Many studies have shown that the orientation of the device indeed has a strong impact on the propagation and should be taken into account. This is particularly relevant for fingerprinting-based localisation solutions. Majority of these studies, however, only perform short-term measurements, which makes them vulnerable to temporal variations in the propagation channel. We are more interested in the long term behaviour of the propagation channel in different direction such that to allow drawing conclusions relevant for the creation of radio maps. Therefore, we performed *Experiment C* based on the set-up of Figure 7.

Eight sensors in scanning modes (SN1-8) are organised in a grid around a central sensor (SN0) that periodically sends out WiFi beacons. The scanning nodes SN1 to SN8 collect RSSI measurements of SN0's beacons. The experiment was run for 24 hours in order to collect a reliable number of samples per SN (ten thousands), which allows us to construct a stochastic profile of the radio channels in each direction. Grid step size is one meter. Sensor nodes' antennas are omnidirectional.

The Cumulative Distribution Functions (CDFs), constructed by the scanning sensors, are shown in Figure 8. The position of the CDF graph in the figure corresponds to the position of the scanning sensor, e.g., the CDF graph at position left-middle corresponds to SN5.

Our main conclusion is that no two nodes have the same distribution of the RSSI values, which is expected and explained with the distinct propagation conditions of each path. Despite the differences there are certain similarities. CDF curves of nodes on the diagonals to SN0 (SNs 1, 3, 6 and 8) have a 5dBm lower mean and a larger variance than SNs 2, 4, 5 and 7 as a result of different path lengths. Interestingly, SN6 is an exception with higher RSSIs, which we attribute to the node's location. A SN near a corner receives stronger reflected

signals from the near walls than a SN in the centre of the room. Nodes from opposite directions also show similar behaviour - SN4 and SN5 have RSSI values mainly spread between -40 and -30dBm, while the CDFs of SN2 and SN7 are in the range of -45 to -33dBm. Although the specific causes for such behaviour are hard to determine, we explain it with the asymmetric shape, i.e., rectangular, of the room and the consequences of that on signal propagation.

Concluding Remarks. The propagation path-specific distribution of the RSSI, besides reconfirming the observations of others, has given us the idea to base our positioning algorithm on a ratio-based approach. This approach is similar to fingerprinting but instead of characterising an indoor location by the absolute RSSI values heard by anchor nodes we can use proportions of the RSSI readings.

5 Conclusion

This paper presented an investigation on the impact of technical characteristics of mobile devices (targets for localisation), manufacturing differences of sensor nodes (used for localisation) and direction-specific multipath propagation. Our main conclusions are: (i) signal strength varies less between sensors of the same type than between mobile devices from different manufacturers; (ii) multipath propagation seems to have strong effect on signal strength; (iii) radio signals experience distinct propagation conditions in different directions.

In parallel, we analysed the usability of four signal metrics, namely, instantaneous values, probability distribution, median and percentiles, mean and standard deviation, as well as the signal's detection rate. We show that the choice of evaluation metric depends on the time-granularity of localisation, i.e., mean values are convenient for real-time positioning while probability distributions may be better for off-line processing.

Based on our findings as a next step we envision to develop a localisation system for indoor applications that can compensate for the specific behaviour of different personal devices and their orientation in a flexible, on-the-fly manner.

References

1. Ahmed, I., Orfali, S., Khattab, T., Mohamed, A.: Characterization of the indoor-outdoor radio propagation channel at 2.4 ghz. In: 2011 IEEE GCC Conference and Exhibition (GCC), pp. 605–608 (February 2011)
2. Akl, R., Tummala, D., Li, X.: Indoor propagation modeling at 2.4 ghz for IEEE 802.11 networks. In: Sixth IASTED International Multi-Conference on Wireless and Optical Communications. IASTED/ACTA Press (2006)
3. Bargh, M.S., de Groote, R.: Indoor localization based on response rate of Bluetooth inquiries. In: Proc. of 1st ACM International Workshop on Mobile Entity Localization and Tracking in GPS-less Environments, MELT 2008, pp. 49–54. ACM (2008)
4. Bose, A., Foh, C.H.: A practical path loss model for indoor wifi positioning enhancement. In: 2007 6th International Conference on Information, Communications Signal Processing, pp. 1–5 (2007)

5. Byoung-Suk, C., Joon-Woo, L., Ju-Jang, L., Kyoung-Taik, P.: Distributed sensor network based on RFID system for localization of multiple mobile agents. In: Wireless Sensor Network, vol. 3-1, pp. 1–9. Scientific Research (2011)
6. Cherukuri, J.: Comparative study of stochastic indoor propagation models. Technical report, The University of North Carolina at Charlotte (2004)
7. Ciurana, M., Barceló-Arroyo, F., Cugno, S.: A robust to multi-path ranging technique over IEEE 802.11 networks. Wireless Networks 16, 943–953 (2010)
8. Fang, S.-H., Lin, T.-N.: Projection-based location system via multiple discriminant analysis in wireless local area networks. IEEE Transactions on Vehicular Technology 58(9), 5009–5019 (2009)
9. Fuchs, C., Aschenbruck, N., Martini, P., Wieneke, M.: Indoor tracking for mission critical scenarios: A survey. Pervasive Mobile Computing 7, 1–15 (2011)
10. Gezici, S., Zhi, T., Giannakis, G.B., Kobayashi, H., Molisch, A.F., Poor, H.V., Sahinoglu, Z.: Localization via ultra-wideband radios: a look at positioning aspects for future sensor networks. IEEE Signal Processing Magazine 22(4), 70–84 (2005)
11. Gwon, Y., et al.: Robust indoor location estimation of stationary and mobile users (2004)
12. Haeberlen, A., Flannery, E., Ladd, A.M., Rudys, A., Wallach, D.S., Kavraki, L.E.: Practical robust localization over large-scale 802.11 wireless networks. In: Proc. of 10th Annual International Conference on Mobile Computing and Networking, MobiCom 2004, pp. 70–84. ACM (2004)
13. Hashemi, H.: The indoor radio propagation channel. Proceedings of the IEEE 81(7), 943–968 (1993)
14. Hay, S., Harle, R.: Bluetooth Tracking without Discoverability. In: Choudhury, T., Quigley, A., Strang, T., Suginuma, K. (eds.) LoCA 2009. LNCS, vol. 5561, pp. 120–137. Springer, Heidelberg (2009)
15. https://www.bluetooth.org/Technical/Specifications/adopted.html
16. http://www.cl.cam.ac.uk/research/dtg/attarchive/bat/
17. Kotanen, A., Hannikainen, M., Leppakoski, H., Hamalainen, T.D.: Experiments on local positioning with bluetooth. In: International Conference on Information Technology: Coding and Computing [Computers and Communications], pp. 297–303 (2003)
18. Liu, H., Darabi, H., Banerjee, P.: A new rapid sensor deployment approach for first responders. Intelligent Control and Systems 10(2), 131–142 (2005)
19. Mahtab Hossain, A.K.M., Nguyen Van, H., Jin, Y., Soh, W.S.: Indoor localization using multiple wireless technologies. In: Proc. of Mobile Adhoc and Sensor Systems, MASS 2007, pp. 1–8 (2007)
20. Martin-Escalona, I., Barcelo-Arroyo, F.: A new time-based algorithm for positioning mobile terminals in wireless networks. Journal on Advances in Signal Processing, EURASIP (2008)
21. Perez-Vega, C., Garcia, J.L., Lopez Higuera, J.M.: A simple and efficient model for indoor path-loss prediction, vol. 8, p. 1166 (1997)
22. Want, R., Hopper, A., Falcão, V., Gibbons, J.: The active badge location system. ACM Trans. Inf. Syst. 10, 91–102 (1992)
23. Zhang, G., Krishnan, S., Chin, F., Ko, C.C.: UWB multicell indoor localization experiment system with adaptive TDOA combination. In: IEEE 68th Vehicular Technology Conference, VTC 2008-Fall, pp. 1–5 (2008)

Localization in Presence of Multipath Effect in Wireless Sensor Networks

Kaushik Mondal[1,⋆], Partha Sarathi Mandal[1], and Bhabani P. Sinha[2]

[1] Indian Institute of Technology, Guwahati, India
[2] Indian Statistical Institute, Kolkata, India

Abstract. Localization in an urban area is a challenging problem due to the blocking of Line-of-Sight (LOS) signal by various obstacles and also the multipath effect arising out of reflections and scattering of signals. Assuming that there are a few anchor nodes which know their positions accurately and which transmit ultrasonic signals, we propose here a technique to find the position of other sensor nodes based on receiving these ultrasonic signals reflected by some reflectors. Our proposed technique can calculate the position of a node correctly by receiving two reflected signals (Non-Line-of-Sight (NLOS)) from an anchor. We, however, assume that a signal is reflected at most once before reaching a node and the two reflecting surfaces are non-parallel to each other.

Keywords: Localization, NLOS, Ultrasound, Reflectors, Wireless networks.

1 Introduction

Estimating the position of a sensor node in wireless sensor networks (WSN) is very important for many real-life applications based on sensor networks. Knowledge about the position of sensor nodes is the basic requirement for position-based or geographic routing protocols.Also, in case of data aggregation, it is important to know the position of a sensor node. The goal of localization is to establish the position of each node as accurately as possible. There exist many localization algorithms [1,2,3,4,6,11,13,18] by which sensor nodes can know their positions. Although GPS is one of the widely used techniques for location discovery in outdoor networks [8], from the point of view of accuracy, cost and energy preservation, it is not always practical to equip each node with a GPS receiver.

For localization without taking help of GPS, we often need a few anchor nodes whose positions are known very accurately. Beacon signals are being transmitted from these anchor nodes to be received by other sensor nodes. In an urban area, the received signal may be a Line-of-Sight (LOS) one, or one that is reflected and/or scattered by various obstacles (e.g., walls of buildings) before reaching the destination node. Various localization techniques have already been proposed in

⋆ The first author is thankful to the Council of Scientific and Industrial Research (CSIR), Govt. of India, for financial support during this work.

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 138–149, 2012.

the literature which are often based on measuring the time of arrival (ToA) [18] or angle of arrival (AoA) [11] of the received signals, to calculate the distances and angles, respectively. In ToA-based techniques, the round trip delay [15] of the signal is measured to calculate the distance between the anchor node and a specific sensor node. In AoA-based techniques, a sensor node can calculate the angle of arrival of the signal from the anchor by using directional antennas or digital compass [11]. However, suitable technique for finding the location of sensor node which mitigates the effect of multiple reflections and/or scattering in the most general environment, is still called for.

1.1 Our Contribution

In this paper we propose a deterministic algorithm that can calculate the position of a sensor node accurately by using one anchor which transmits an ultrasonic signal, and assuming the presence of two reflectors (in the absence of a direct path or LOS communication) for this signal to be received by the node in question whose position is to be estimated. We assume the following in our model:

- A node receives two beacons (ultrasonic signals) from any particular anchor where the reflectors are not parallel to each other.
- Each beacon is reflected only once before reaching the destination node.
- Each node is equipped with the appropriate mechanism to measure both the time of arrival (ToA) and angle of arrival (AoA) of the received signal.

It follows from the last assumption above that in presence of a direct path (LOS), one beacon is sufficient for estimating the position of the node in question.

In most of the earlier works [1,2,3,4,6,13,18], usually three anchor nodes are used to locate a sensor node. But in our approach, only one anchor node is sufficient to calculate the position of a node. This gives us an advantage particularly in dealing with the sparse networks, by having only a single anchor node, which can locate all other nodes in its range.

2 Related Works

There are two basic classes of the localization algorithms in WSNs, *range-based* and *range free*. The range based algorithms are more accurate than range free [2] ones. Usually for range estimation, ToA [6,15], time difference of arrival (TDoA) [4,7], AoA [7] and received signal strength (RSS) [5] are used. Researches are going on for better and accurate range estimations. Zhang *et al.* [17] proposed a distributed angle estimation method for WSN localization with multipath fading. There is another range-based method TPS [4], which uses TDoA to detect location of the nodes. Delaet *et al.* proposed a deterministic secure localization algorithm [6] based on RSS and TDoA techniques. *Trilateration* [14] is used when the distances between a node and at least three anchor nodes are known. It finds the intersection point of the three circles centered at the anchor nodes as the position of the sensor node. This is almost same as circular triangulation

which is used in three masters method proposed by Patil *et al.* in [13]. Error in distance measurement leads to an intersection area of those three circles instead of a point in the method of trilateration. To handle these errors, *multilateration* [3] has been proposed. When the number of anchors or verifiers are more than three, then multilateration is useful. Minimum mean square error method (MMSE) [3,18] is used in multilateration which attempts to detect the position of a node by minimizing the error using an objective function. On the other hand, in range free techniques [2] information like hop counts are used. Sub-area localization (SAL) [1] is a range free technique, where the central server finds the correct sub-area and sets the center of mass of the sub-area as the node's position.

Sometimes sensor nodes may be under the control of some adversaries or some attackers. In that case after calculating the position, verification is also required to ensure correct position of the node before using the position for some applications. This is known as secure localization [2,6], where the authors used plane geometry to estimate the position of a node deterministically.

Ebrahimian and Scholtz proposed a source localization scheme using reflection in [7], where direct and reflected signals are used. Here the sensor nodes are equipped with unidirectional antenna. Salvador *et al.* [10] uses trigonometric figures for node localization in a probabilistic model. Uchiyama *et al.* proposed UPL algorithm in [16] for positioning mobile users in urban area. In this work authors consider known positions of the obstacles such as walls and calculate the area of presence of mobile users with certain degree of accuracy. In [9], authors proposed an ultrasonic-based localization system for mobile robot. Their proposed algorithm provides sufficient accuracy in the positioning of a robot based on ultrasonic reflection. Pahlavan *et al.* [12] proposed indoor geo-location in the absence of direct path. Oberholzer *et al.* proposed ultrasonic-based ranging platform, called SpiderBat [11] which is the first ultrasonic-based sensor node platform that can measure absolute angles between sensor nodes accurately. With the help of measured angles it is possible to estimate node positions with a precision of the order of a few centimeters.

3 Basic Idea

We assume that the sensor nodes have been deployed on a two dimensional plane and each has been assigned a unique *id*. There are some reflectors and anchor nodes in the same plane. The position of a node is calculated based on a chosen coordinate system. The position of the anchor nodes are known and an anchor can be identified uniquely by its position. An anchor node broadcasts beacon along with its position. A sensor node may receive the beacon directly (LOS communication) or after a reflection from some reflector. We assume that there can be at most one reflection in the path from any anchor to another node. One node may, however, receive more than one beacons from different anchors. After receiving a beacon from an anchor, a node transmits back the signal in the same direction from which it received the beacon (the angle of arrival with

respect to the common coordinate system being measured by AoA technique). After receiving those beacons, the anchors also use ToA and AoA techniques to calculate the round-trip time of arrival and angle of arrival with respect to the same coordinates system for the same node. We show later that with the help of those angles of arrival and the distances calculated from the ToA technique, it is possible for the anchors to find the exact position of a node.

The connection between the geometry of Fig. 1 used in the following Lemma 1 and our localization problem is the following: the point S can be considered as an anchor, the point P is the point of reflection on a reflector and the point Q is the position of a sensor node, whose position is to be computed. The node at Q receives the beacon from S through the path SQ via P, after one reflection at P. We now state the following result :

Lemma 1. *Consider a fixed point S on a straight line l with gradient m_l. Let L be the set of all parallel straight lines with gradient m_L such that $m_l \neq m_L$. Let P_i be the point of intersections of l with a line $\ell_i \in L$, for $i = 1, 2, \cdots$ (refer to Fig. 1). Let Q_is be the points on ℓ_i such that $SP_i + P_iQ_i = d$, for $i = 1, 2, \cdots$, where d is a fixed distance. Then all the Q_is must lie on a straight line.*

Proof. Without loss of generality, we consider the fixed point S on the straight line l to be the origin of the coordinate system, having the coordinates (0,0). As shown in Fig. 1, P is the point of intersection of l with some straight line $\ell \in L$. Let Q be the point on ℓ such that $SP + PQ = d$. We need to show that the locus of Q is a straight line.

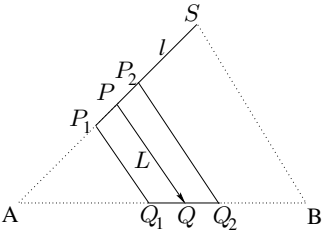


Fig. 1. Figure showing one *fixed_distance_line*, Q_1Q_2

Noting that the equation of the straight line l is $y = m_l x$, let the coordinates of the point P be equal to $(c, m_l c)$, for some $c \neq 0$ and those of $Q = (\alpha, \beta)$. Hence, the gradient of PQ is $m_L = \frac{\beta - m_l c}{\alpha - c}$, which implies that $\beta = \alpha m_L + c(m_l - m_L)$. Now, $SP = \pm c\sqrt{1 + m_l^2}$, $PQ = \pm(\alpha - c)\sqrt{1 + m_L^2}$, the positive or negative signs are chosen depending on the values of c and $(\alpha - c)$. Then from $SP + PQ = d$ and $\beta = \alpha m_L + c(m_l - m_L)$, we get $\beta = \alpha \left[\frac{m_l\sqrt{1+m_L^2} - m_L\sqrt{1+m_l^2}}{\sqrt{1+m_L^2} - \sqrt{1+m_l^2}} \right] \pm \frac{d(m_l - m_L)}{\sqrt{1+m_L^2} - \sqrt{1+m_l^2}}$ or $\beta = \alpha \left[\frac{m_l\sqrt{1+m_L^2} + m_L\sqrt{1+m_l^2}}{\sqrt{1+m_L^2} + \sqrt{1+m_l^2}} \right] \pm \frac{d(m_l - m_L)}{\sqrt{1+m_L^2} + \sqrt{1+m_l^2}}$, which implies that the locus of Q is a straight line. □

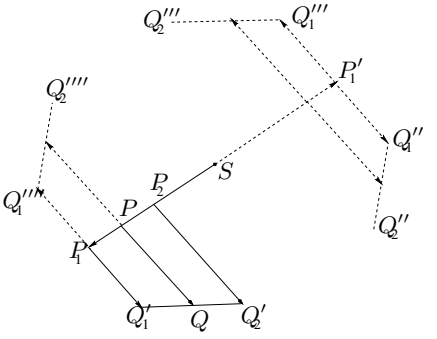


Fig. 2. Showing four possible *fixed_distance_lines*, $Q_1'Q_2'$, $Q_1''Q_2''$, $Q_1'''Q_2'''$, $Q_1''''Q_2''''$

Let the coordinates of the point P_1 in Fig. 1 be $(c', m_l c')$, for some $c' \neq 0$ and those of Q_1 be equal to (α', β'). Then using $SP_1 + P_1Q_1 = d$ and $m_L = \frac{\beta' - m_l c'}{\alpha' - c'}$, we get $\alpha' = c' \pm \frac{d \pm c'\sqrt{1+m_l^2}}{\sqrt{1+m_L^2}}$ and the corresponding $\beta' = m_l c' \pm m_L \frac{d \pm c'\sqrt{1+m_l^2}}{\sqrt{1+m_L^2}}$. Similarly, let the coordinates of the point P_2 in Fig. 1 be $(k, m_l k)$, for some k such that $c' \neq k \neq 0$ and those of $Q_2 = (\alpha'', \beta'')$. Then using $SP_2 + P_2Q_2 = d$ and $m_L = \frac{\beta'' - m_l k}{\alpha'' - k}$, we get, $\alpha'' = k \pm \frac{d \pm k\sqrt{1+m_l^2}}{\sqrt{1+m_L^2}}$ and the corresponding $\beta'' = m_l k \pm m_L \frac{d \pm k\sqrt{1+m_l^2}}{\sqrt{1+m_L^2})}$.

Considering all combinations of α', β' calculated above, we get four possible positions – Q_1', Q_1'', Q_1''' and Q_1'''' for Q_1, as shown in Fig. 2, with the above computed values of the coordinates of Q_1. Similarly, there are four possible positions – Q_2', Q_2'', Q_2''' and Q_2'''' for Q_2. Each such possible position of Q_1 corresponds to one possible position of Q_2 and we get four lines $Q_1'Q_2'$, $Q_1''Q_2''$, $Q_1'''Q_2'''$ and $Q_1''''Q_2''''$, as shown in Fig. 2. From the computed coordinate values of Q_1 and Q_2, the gradient of the line joining Q_1 and Q_2 is either $M' = \frac{m_l\sqrt{1+m_L^2} - m_L\sqrt{1+m_l^2}}{\sqrt{1+m_L^2} - \sqrt{1+m_l^2}}$ or $M'' = \frac{m_l\sqrt{1+m_L^2} + m_L\sqrt{1+m_l^2}}{\sqrt{1+m_L^2} + \sqrt{1+m_l^2}}$. Also, we see that $M'M'' = -1$

Thus, among the four lines $Q_1'Q_2'$, $Q_1''Q_2''$, $Q_1'''Q_2'''$ and $Q_1''''Q_2''''$, as shown in Fig. 2, the non-intersecting lines are parallel to each other, *i.e.*, $Q_1'Q_2' \parallel Q_1'''Q_2'''$ and $Q_1''Q_2'' \parallel Q_1''''Q_2''''$ and the intersecting lines are perpendicular to each other, *i.e.*, $Q_1'Q_2' \perp Q_1''Q_2''$, $Q_1''Q_2'' \perp Q_1'''Q_2'''$, $Q_1'''Q_2''' \perp Q_1''''Q_2''''$, and $Q_1''''Q_2'''' \perp Q_1'Q_2'$.

Definition 1. *The locus of Q is a straight line, which we define as the $fixed\_distance\_line_{S,m_l,m_L,d}$ corresponding to the point S on the straight line l with gradient m_l, the gradient m_L of the set L of parallel lines and the fixed distance d. However, when the values of the subscripts S, m_l, m_L and d are apparent from the context, we would, for simplicity, denote it only by $fixed\_distance\_line$, by dropping the subscripts. Also, the gradient of the locus of Q is defined as the $fixed\_distance\_gradient_{S,m_l,m_L,d}$.*

The straight lines $Q'_1Q'_2$, $Q''_1Q''_2$, $Q'''_1Q'''_2$ and $Q''''_1Q''''_2$, as shown in Fig. 2, are four possible $fixed\_distance\_line_{S,m_l,m_L,d}$. From the above discussions, we get the following result.

Lemma 2. *Among the four $fixed\_distance\_line_{S,m_l,m_L,d}$, the intersecting lines are perpendicular to each other, and the non-intersecting lines are parallel to each other.*

Lemma 3. *A bisector of one of the angles between the straight lines l and any line $\in L$ is parallel to the $fixed\_distance\_line_{S,m_l,m_L,d}$.*

Proof. It follows that the gradients of the angular bisectors of the angles between l and any line $\in L$ are $(m_l\sqrt{1+m_L^2} - m_L\sqrt{1+m_l^2})/(\sqrt{1+m_L^2} - \sqrt{1+m_l^2})$ and $(m_l\sqrt{1+m_L^2} + m_L\sqrt{1+m_l^2})/(\sqrt{1+m_L^2} + \sqrt{1+m_l^2})$. According to Lemma 1, these are the possible gradients of the $fixed\_distance\_line_{S,m_l,m_L,d}$. □

Lemma 4. *The position of a sensor node cannot be uniquely identified by the above method, if and only if the node receives two beacons from the same anchor node which are reflected from two parallel reflectors.*

Proof. First, we examine the characteristics of a pair of beacons, received by a node from an anchor. As shown in the Fig. 3, S is the anchor, Q is the actual position of the node. Q receives beacons from S through P_1 and P_2, where P_1 and P_2 are the two points of reflection. Let $\angle P_1QX' = \theta_1$, $\angle P_2QX' = \theta_2$, (all the angles are measured counterclockwise with respect to the positive X-axis), $SP_1 + P_1Q = d_1$ and $SP_2 + P_2Q = d_2$. By virtue of ToA and AoA measurements, all of these parameters, i.e., θ_1, θ_2, d_1 and d_2 will be known to S (the exact details of finding the values of these four parameters are explained later in this section.

In Fig. 3, we have shown another arbitrarily chosen point P'_1 on the line SP_1 of gradient, say, m_l and another point Q' such that $\angle P'_1Q'X'' = \theta_1$ and $SP'_1 + P'_1Q' = d_1$. This implies that the point Q' is on the $fixed\_distance\_line_{S,m_l,m_L,d_1}$ where m_L is the gradient of the line P_1Q. That means, line QQ' itself is the $fixed\_distance\_line_{S,m_l,m_L,d_1}$.

Similarly, if we find another point P'_2 on the line SP_2 having gradient, say, m'_l such that $\angle P'_2Q'X'' = \theta_2$ and $SP'_2 + P'_2Q' = d_2$, then the line QQ' will also be the $fixed\_distance\_line_{S,m'_l,m'_L,d_2}$, where m'_L is the gradient of the line P_2Q. Now we will have problems in uniquely identifying the position of the sensor node from the signals reflected from P_1 and P_2, as the solution for the possible position of Q in this case will be infinitely many (corresponding to the line QQ'). We note that, under such a situation, according to Lemma 3, QQ' is parallel to the bisectors of both the angles $\angle SP_1Q$ and $\angle SP_2Q$. Hence, the two reflectors are parallel to each other.

Conversely, assume that there are two parallel reflectors. The two beacons from an anchor reflected from those two parallel reflectors reach a sensor node. The bisectors of the angles formed at the points of reflections will hence be parallel. According to Lemma 3, the *fixed_distance_lines* corresponding to these two reflected paths will have the same gradient, i.e., the possible solution for the position of the sensor node will be a line. □

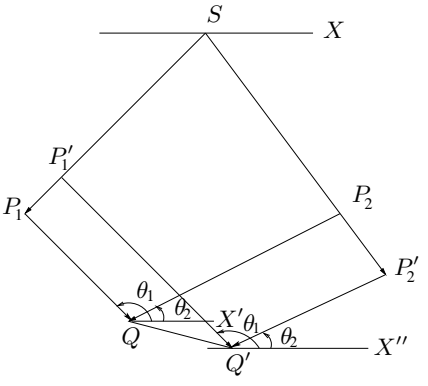


Fig. 3. Figure showing infinite many solutions for the position of a sensor node along the line QQ'

We assume that whenever a sensor node Q receives a beacon from the anchor node S, it measures the angle of arrival, say θ of the received signal, with respect to the positive X-axis (measured in the counterclockwise direction) and immediately sends back a signal along the same angle θ which will be received by S. Through this signal, S also receives the information about this angle θ from Q. The anchor node S, on receiving this signal back from Q, can then measure the round-trip delay of this signal from S to Q and back to S, from which S can compute the distance between S and Q along this path of signal propagation. S also measures the angle of arrival, say δ while receiving the signal from Q, with respect to the positive X-axis (measured in the counterclockwise direction). We now have the following result.

Theorem 1. *A node finds its position correctly if it receives either i) the direct (LOS) signal from an anchor node, or ii) two reflected signals from an anchor node with the corresponding reflectors not being parallel to each other.*

Proof. We assume, without loss of generality, that S is the origin of the coordinate system. Let the position of the sensor node Q be (α, β). We need to compute α and β.

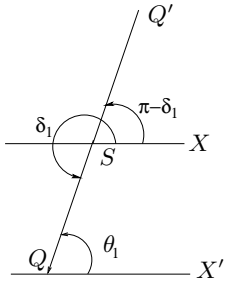


Fig. 4. An example showing another possible position Q' of the node Q

Case 1: Q receives a direct signal from S.
Fig. 4 shows a path through which Q receives the beacon from the anchor node S directly. By virtue of the actions taken by Q and S as discussed above, S knows the distances $SQ = d$ and also the angles $\angle SQX' = \theta_1$ and $\angle XSQ = \delta_1$ as shown in Fig. 4. If S finds that $\theta_1 = \delta_1 \pm \pi$, then S identifies that the signal path SQ is a direct one (LOS) and then it calculates the position of Q as follows.

The equation of the straight line SQ is $y = m_1 x$, where $\tan\delta_1 = m_1$. Hence, $\beta = m_1\alpha$. Since $SQ = d$, we have $\alpha^2(1 + m_1^2) = d^2$, from which we get $\alpha = \pm d/\sqrt{1 + m_1^2}$. The coordinates $\left(d/\sqrt{1 + m_1^2},\ m_1 d/\sqrt{1 + m_1^2}\right)$ and $\left(-d/\sqrt{1 + m_1^2},\ -m_1 d/\sqrt{1 + m_1^2}\right)$ are the two possible positions of Q as shown in Fig. 4. Now to choose the correct position of Q from the above two coordinates, S finds whether the measured angles δ_1 and θ_1 are related as $\theta_1 = \delta_1 - \pi$ or $\theta_1 = \delta_1 + \pi$. S then verifies which of the above two computed coordinate values of Q corresponds to the required relationship between δ_1 and θ_1, and selects that one as the final position of Q.
Case 2: Q receives two reflected signals from S.

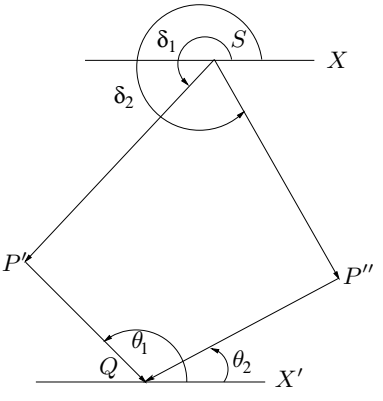


Fig. 5. An example showing both reflected beacons received by a node Q

S can identify this case if $\theta_1 \neq \delta_1 \pm \pi$ and $\theta_2 \neq \delta_2 \pm \pi$. The situation can be illustrated by Fig. 5, where we assume that P' and P'' are the two points of reflection on the reflecting surfaces. By virtue of the actions taken by S and Q discussed above for each of the two reflected signals, S knows the angles $\angle XSP' = \delta_1$, $\angle P'QX' = \theta_1$, $\angle XSP'' = \delta_2$, and $\angle P''QX' = \theta_2$, as well as the distances $SP' + P'Q = d_1$, and $SP'' + P''Q = d_2$. Let $\tan\delta_1 = m_1$ and $\tan\delta_2 = m_2$. Then the equations of SP' and SP'' are $y = m_1 x$ and $y = m_2 x$, respectively. Similarly, let $\tan\theta_1 = m_3$ and $\tan\theta_2 = m_4$. Then the equations of $P'Q$ and $P''Q$ are $y - \beta = m_3(x - \alpha)$ and $y - \beta = m_4(x - \alpha)$, respectively. Now S computes the coordinates of the point P' in terms of α, β as the intersection point of the straight lines SP' and $P'Q$. Similarly, the coordinates of P'' are computed by S as the intersection point of the lines SP'' and $P''Q$. Thus, the coordinates of $P' = ((\beta - m_3\alpha)/(m_1 - m_3), m_1(\beta - m_3\alpha)/(m_1 - m_3))$, and those

of $P'' = ((\beta - m_4\alpha)/(m_2 - m_4), m_2(\beta - m_4\alpha)/(m_2 - m_4))$. Then S solves the following two equations:

$$SP' + P'Q = d_1 \text{ and } \mathrm{SP''} + \mathrm{P''Q} = \mathrm{d_2} \tag{1}$$

which actually means solving the following two second degree equations :

$$\alpha^2 + \beta^2 + 2(\beta - m_3\alpha)(d_1\sqrt{1 + m_1^2} - (\alpha + m_1\beta))/(m_1 - m_3) - d_1^2 = 0$$
$$\alpha^2 + \beta^2 + 2(\beta - m_4\alpha)(d_2\sqrt{1 + m_2^2} - (\alpha + m_2\beta))/(m_2 - m_4) - d_2^2 = 0$$

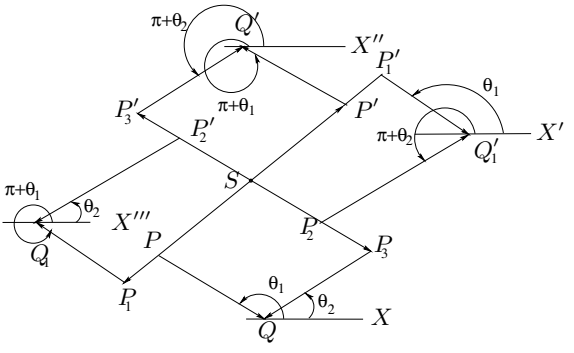


Fig. 6. An example showing four possible positions Q, Q', Q_1, Q_1' of a node Q

These two equations give four solutions as the possible position of the node, which are actually the four intersection points of the eight *fixed_distance_line* corresponding to these two reflected paths. In Fig. 6, we have shown the four possible positions of the sensor node as Q, Q', Q_1 and Q_1', respectively resulting from the solution. All these four points satisfy eqn. 1, but only one of these satisfies the angle constraints (namely, two angles made by the beacons at S and two angles at Q). Corresponding to the line joining S and one reflection point, say P as shown in Fig. 6, two of the solution points are on one side of this line and the remaining two are on the other side. The ambiguity about the actual solution point can be resolved as follows.

Corresponding to each of the four solution points obtained by solving eqn. 1, the values of α and β are substituted back in equation $y - \beta = m_3(x - \alpha)$ and $y - \beta = m_4(x - \alpha)$. The first one of these two equations and $y = m_1x$ are solved to get the coordinates of one reflection point, say P' (see Fig. 5), while the second one and $y = m_2x$ are solved to find the coordinates of the other reflection point, say P'' in Fig. 5. The angles made by the lines $SP', SP'', P'Q$ and $P''Q$ are computed and compared with the actually measured angles $\delta_1, \delta_2, \theta_1$, and θ_2, respectively. It follows from Fig. 6 that only one of the four solution points will satisfy the constraints on these four angles, and that particular solution is selected as the actual position of the sensor node. □

4 Proposed Localization Algorithm

4.1 System Model

We assume that all anchor nodes are equipped with an omnidirectional antenna for sending a beacon and also a directional antenna for the measurement of angle of arrival of a signal from other sensor nodes. An anchor node sends its own position as *id* with the beacon signal. We further assume that other sensor nodes are equipped with directional antennas for the measurement of angle of arrival of the beacon signal from an anchor. An anchor is said to be a *neighbor* of another anchor if it is located within twice the transmission range of the second anchor. Anchor nodes are synchronized with some global clock (possibly through GPS) such that at a time only one anchor sends a beacon to avoid collision with the beacons from neighboring anchors. This ensures that a receiving sensor node receives only one beacon at a time from a particular anchor node. A sensor node receives the beacon from the anchor by using a directional antenna so that it can also measure the angle of arrival of the beacon signal. Because of this directional antenna, the sensor node does not experience a collision even when it receives more than one beacon from an anchor coming through different paths at different angles. We assume that after receiving a beacon, a sensor node transmits back a signal to the anchor, in the same direction in which it has received the beacon from the anchor. This signal carries the *id* of the sensor node and the angle of arrival (AoA) of the beacon in the format, <*id*, AoA>.

4.2 The Algorithm

Based on the above discussions, Algorithm: FindPosition given below finds the position of a sensor node Q using the beacon signals from an anchor node S. Q may receive either the direct signal (LOS communication) from S and/or it may receive one or more reflected signals from S.

Algorithm 1. FindPosition

1: Anchor S sends a beacon with <anchor_id> using omnidirectional antenna.
2: **for** each node Q who hears the beacon **do**
3: Q measures the angles of arrival (θ_i) of all received beacon(s), $i = 1, 2, \cdots$, and transmits back <node_id, θ_i> to S via the same path(s).
4: **end for**
5: S measures the angles of arrival (δ_i) while receiving all replies and computes the corresponding distances (d_i) traveled by the beacon by measuring the ToA. From the θ_i and δ_i values, S determines whether Q received the signal(s) along a direct path and/or reflected paths. After that, S executes the steps 6, 7 and 8 below.
6: If Q received at least one beacon from S through the direct (LOS) path, then S computes the location of Q following case 1 of theorem 1.
7: If Q received at least one pair of beacons through two non-parallel reflectors, then S computes the location of Q following case 2 of theorem 1.
8: For all other cases, S reports the inability to compute the location of Q unambiguously.

5 Simulation

Our proposed algorithm calculates the position of a node correctly if the received signals are either direct or one bound. But in a practical situation, the received signals may be multi-bound (more than one bound). If the position of a node is calculated with multi-bound signals treated as one bound using the proposed algorithm, then error may be introduced. Considering this fact, we have simulated the measurements with randomly deployed nodes and reflectors, and calculated the corresponding errors. Our simulation is restricted to the consideration of up to four bound signals only.

Table 1. Error due to false interpretation of multi-bound signal as one-bound signal

No of runs	1	2	3	4	5	6	7	8	9	10
% of multi-bound signals treated as one bound	27.9	25.2	30.9	26.7	24.8	30.1	33.3	26.1	24.8	23.9
percentage error	67.3	62.4	60.3	66.4	55.1	65.9	66.0	69.0	61.1	59.3

Table 1 shows percentage of the multi-bound signals treated as one bound and the corresponding percentage error. From this table, it appears that more than 70% multi-bound signals can be identified using our technique, and from the multi-bound signals wrongly interpreted as one bound signals, the position of a node can be calculated with a percentage error varying from 55-69%.

6 Conclusions

In this paper we have proposed a deterministic algorithm to find the position of a sensor node based on receiving two reflected signals from only one anchor node. The proposed technique does not need to know the positions of those reflectors. In the presence of a direct path communication, the position of a node can be easily calculated using the same algorithm. In the presence of some parallel reflectors, the position of any node can also be determined if the node receives two reflected signals from the same anchor through any two non-parallel reflectors. In case of a mobile sensor node, the proposed technique is also very useful to locate the position of the mobile node. In this case each anchor has to broadcast the beacon several times at regular time intervals to find the track followed by the mobile sensor node. The time interval to be selected is dependent on the speed of mobility of the nodes. Future research work in this direction includes the extension of our idea to deterministic localization of the sensor nodes where the beacons may suffer multiple reflections before being received by a node.

References

1. Aksu, A., Krisnamurthy, P.: Sub-area localization: A simple calibration free approach. In: Proc. of the 13th ACM Int. Conf. on Modeling, Analysis, and Simulation of Wireless and Mobile Systems (MSWIM 2010) (2010)

2. Ammar, W., ElDawy, A., Youssef, M.: Secure localization in wireless sensor networks: A survey. CoRR abs/1004.3164 (2010)
3. Capkun, S., Hubaux, J.-P.: Secure positioning of wireless devices with application to sensor networks. In: INFOCOM, pp. 1917–1928 (2005)
4. Cheng, X., Thaeler, A., Xue, G., Chen, D.: TPS: A time-based positioning scheme for outdoor wireless sensor networks. In: INFOCOM (2004)
5. Cheng, X., Huang, X., Du, D.-Z.: Ad-hoc wireless networking. Springer (2004)
6. Delaët, S., Mandal, P.S., Rokicki, M.A., Tixeuil, S.: Deterministic secure positioning in wireless sensor networks. Theoretical Computer Science 412(35), 4471–4481 (2011)
7. Ebrahimian, Z., Scholtz, R.A.: Source localization using reflection omission in the near-field. In: IEEE-ACES Conf. on Applied Comput. Electromagnetics (2005)
8. Hofmann-Wellenhof, B., Lichtenegger, H., Collins, J.: Global Positioning System: Theory and Practice, 4th edn. Springer (1997)
9. Hsu, C.-C., Chen, H.-C., Lai, C.-Y.: An improved ultrasonic-based localization using reflection method. In: Proc. of Int. Asia Conf. on Informatics in Control, Automation and Robotics (CAR 2009), pp. 437–440. IEEE Computer Society (February 2009)
10. Jauregui-Ortiz, S., Siller, M., Ramos, F.: Node localization in wsn using trigonometric figures. In: Proc. of IEEE Topical Conf. on Wireless Sensors and Sensor Networks (WiSNet 2011), pp. 65–68 (January 2011)
11. Oberholzer, G., Sommer, P., Wattenhofer, R.: SpiderBat: Augmenting wireless sensor networks with distance and angle information. In: IPSN, pp. 211–222 (2011)
12. Pahlavan, K., Akgul, F.O., Heidari, M., Hatami, A., Elwell, J.M., Tingley, R.D.: Indoor geolocation in the absence of direct path. IEEE Wireless Communications 13(6), 50–58 (2006)
13. Patil, M.M., Shaha, U., Desai, U.B., Merchant, S.N.: Localization in wireless sensor networks using three masters. In: Proc. of the IEEE Int. Conf. on Personal Wireless Commu. (ICPWC 2005), pp. 384–388 (2005)
14. Shih, C.-Y., Marrón, P.J.: COLA: Complexity-reduced trilateration approach for 3D localization in wireless sensor networks. In: SENSORCOMM 2010, pp. 24–32. IEEE Computer Society, Washington, DC (2010)
15. Singelee, D., Preneel, B.: Location verification using secure distance bounding protocols. In: IEEE International Conference on Mobile Adhoc and Sensor Systems, pp. 834–840 (November 2005)
16. Uchiyama, A., Fujii, S., Maeda, K., Umedu, T., Yamaguchi, H., Higashino, T.: Ad-hoc localization in urban district. In: INFOCOM, pp. 2306–2310. IEEE (2007)
17. Zhang, W., Yin, Q., Chen, H., Wang, W., Ohtsuki, T.: Distributed angle estimation for wireless sensor network localization with multipath fading. In: Proc. of IEEE Int. Conf. on Communications (ICC 2011), pp. 1–6. IEEE (June 2011)
18. Zhang, Y., Liu, W., Fang, Y., Wu, D.: Secure localization and authentication in ultra-wideband sensor networks. IEEE Journal on Selected Areas in Communications 24(4), 829–835 (2006)

RNBB: A Reliable Hybrid Broadcasting Algorithm for Ad-Hoc Networks

Ausama Yousef[1], Samer Rishe[1], Andreas Mitschele-Thiel[1], and Abdalkarim Awad[2]

[1] Integrated Communication Systems Group, Ilmenau University of Technology, Helmholtzplatz 5, Germany
{ausama.yousef,samer.rishe,mitsch}@tu-ilmenau.de
[2] Computer Networks and Communication Systems, Dept. of Computer Science, University of Erlangen, Martensstrasse 3, 91058 Erlangen, Germany
abdalkarim.awad@informatik.uni-erlangen.de

Abstract. Broadcasting is considered one of the main challenges in Mobile Ad-hoc Networks (MANETs). Therefore, several algorithms have been proposed, such as Self-Pruning (SP), Counter-Based (CB) and Dominant-Pruning (DP). However, they suffer from low packet delivery in some environments, such as the mobility scenario. This paper presents Reliable Neighbor-Based Broadcasting (RNBB), an efficient broadcast algorithm which can meet the broadcast requirements in MANETs. The design principle is to construct a hybrid scheme between SP and CB broadcast schemes aiming to combine the advantages of the both. To make the solution robust, new ideas such as adaptive broadcast delay and an eavesdropping mechanism are integrated to enable RNBB to mitigate the broadcast storm problem while ensuring highly reliable packet delivery. To evaluate RNBB, a comprehensive comparison with different developed algorithms has been made using ns-2. The experimental results demonstrate that RNBB provides robust performance across a wide range of environments.

Keywords: Ad-hoc networks, broadcasting algorithms, ns-2.

1 Introduction

Mobile Ad-hoc Network (MANET) applications cover many areas where wired communication is not possible, such as vehicle Ad-hoc networks, catastrophe-rescue operations, military operations, wireless sensor and actor networks and wireless personal area networks[1]. In MANETs, the method that delivers a packet from one node to all the nodes (broadcasting) is useful for performing many tasks such as neighbor discovery, alarm propagation, address assignment and route discovery used by several routing protocols, such as Dynamic Source Routing (DSR)[2] and Ad-hoc on Demand Distance Vector (AODV) [3]. Due to the dynamic topology of MANETs and the absence of fixed infrastructure, broadcasting is expected to be performed frequently [4] [5]. However, the broadcast connection is an unreliable mechanism because no acknowledgment is used.

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 150–161, 2012.

To perform the broadcast, simple flooding is considered the basic approach in which each node in MANETs forwards the packet exactly once. This approach is simple, easy to implement, and guarantees high packet delivery. However, it causes serious problems in the network referred to as the broadcast storm problem including packets redundancy, contention and collision [6]. Although several methods for the broadcast process have been proposed, they still show drawbacks of achieving higher packet delivery, fewer redundant packets, or lower latency in some scenarios, such as mobility, dense or sparse nodes scenarios. In this paper we present the RNBB algorithm as a novel efficient broadcast algorithm, whose main feature is to reduce the broadcast redundancy while ensuring high reliability of packet delivery. The design principle is to construct a hybrid scheme between SP and CB broadcast schemes. Nevertheless, the advantage obtained from the combination of these two algorithms is not sufficient to meet all broadcast requirements. Therefore, new ideas for analyzing the network topology, adaptive scheduling of packets, coding the neighbor list and reliable recovery of packet loss (see section 3) have been developed to reduce the broadcast storm problem under a wide variety of MANET topologies while maintaining high broadcast deliverability. We compared RNBB against a number of representative wireless broadcasting protocols using ns-2, which we have chosen explicitly because it has well tuned implementations for several broadcasting algorithms, in addition to detailed simulation models of both MAC and physical layers.

The rest of the paper is organized as follows: In Section 2, we outline relevant related work; In Section 3, an overview of the working principles of our RNBB protocol is presented; Afterwards, the implementation and selected evaluation results are presented in Section 4, which also includes a comparison to several MANET broadcasting protocols to evaluate the broadcasting performance of our RNBB protocol; Finally, Section 5 concludes the paper.

2 Related Work

Broadcasting methods have been categorized into four families according to the network knowledge that the node needs in order to make the decision to broadcast [7] [8]. Simple Flooding, Probability Based, Area Based, and Neighbor Knowledge categories. Apart from Simple Flooding, in which every node receiving the message has to rebroadcast it exactly once, each category aims at reducing the energy and bandwidth usage by minimizing broadcast redundancy.

The probabilistic scheme is similar to Simple Flooding, except that the nodes re-broadcast with a predetermined probability, e.g. GOSSIP algorithms [9]. In networks with high densities, randomly reducing the transmissions saves the network resources without an effect on the broadcast deliverability. However, in sparse networks, the probabilistic scheme cannot ensure high reliability. In Probability Based category, the Counter-Based (CB) scheme [4] is designed, also, for dense networks, wherein, a counter c (initially set to 1) and a timer with a Random Assessment Delay (RAD) (which is randomly chosen between 0 and Tmax) are used. During the RAD, the counter c is incremented by every additional

copy of the message. When the RAD expires, if $c > D$ (where D is a threshold of the received copies), the node drops the message, otherwise it rebroadcasts the message. The idea behind Area Based Methods is to ensure a larger coverage area with every transmission of the broadcast message by analyzing the distances to the source node, wherein, the receiver at a distance bigger than a threshold is able to transmit the message. Because it doesn't consider the nodes density within the area, the broadcast process may be killed before covering the whole network. In Self Deterministic and Dominant Deterministic methods, which belong to the Neighbor Knowledge-based category, neighbor information is an essential part for those methods. Nodes in Self Deterministic methods, e.g. Self-Pruning (SP) [10], take the decision of rebroadcasting or ignoring the message on their own by comparing its neighbors list with the one included in the received message. If the receiver can cover additional nodes, it rebroadcasts the message; otherwise the message will be ignored. This is the simplest Neighbor Based Method with considerable amount of overhead but, also, it can ensure the highest reliability in this category. In contrast, nodes that use Dominant Deterministic methods do not make the decision to rebroadcast or ignore the message on their own. Instead, the sender has to define from its neighbors a list of dominating nodes which are responsible for rebroadcasting the message. To build such neighbors lists Dominant-Pruning [11] needs at least 2-hop neighbor knowledge. Because it is infeasible to get accurate 2-hop neighbors' knowledge in some cases, like mobility, this method may suffer from low packet delivery.

To reach high packet delivery, other methods have been proposed. Some methods, such as efficient scheduling and handshaking, try to ensure that only one node at a time will send the broadcast message. Thus, the problem of packet collisions can be alleviated. The idea presented in [12] follows efficient scheduling methods to assign different transmission times for the neighboring nodes which will forward a broadcast message; however, to achieve this, accurate information of a node's position by using a Global Positioning System (GPS) is required. Of course this method can increase the reliability, but it is infeasible because not all devices are GPS equipped. Moreover, this method doesn't guarantee the successful reception of a message by the responsible nodes. In handshaking approaches, the RTS/CTS mechanism is employed as in [13] [14]. In this approach the number of nodes sending simultaneously a broadcast is kept at minimum. However, the use of this mechanism is not efficient because it leads to high protocol overhead, especially when the broadcast message consists of more than one packet. Other methods, such as repetition and overhearing, allow the node to retransmit a broadcast packet twice or more. Thus, the likelihood of receiving the packet by the neighboring nodes will be higher. In [15] the algorithm applies the repetition method where the packet should be sent several times within a time frame divided into many time slots. However, this mechanism faces serious problems such as high power consumption, protocol overhead and deterioration of the signal quality caused by wireless fading. The algorithm presented in [16], which belongs to the deterministic category, utilizes the overhearing mechanism. In this algorithm there are forwarding and non-forwarding nodes and each

non-forwarding node should be covered by at least two forwarding nodes. If the sender fails to hear from its forwarding nodes during a predefined duration, it has to repeate the transmission once again (for a predefined maximum number of retransmissions). Because this algorithm depends on 2-hop neighbor knowledge it may suffer from inaccurate broadcast decision. Moreover, if the non-forwarding nodes don't get the message due to a congestion, there will be no possibility to announce them.

3 Reliable Neighbor-Based Broadcasting (RNBB)

3.1 Design Principles

The RNBB algorithm belongs to the Probability Based as well as Neighbor Knowledge categories. The main idea of RNBB is to adapt the basic ideas of two methods: Self-Pruning and Counter-Based. By analyzing the network topology and broadcast events, RNBB can define a proper behavior for each node to achieve an efficient forwarding. Additionally, the concept of overhearing is used to enhance the packet delivery ratio. Furthermore, neighbor list coding and adaptive packet scheduling have been integrated to make RNBB more robust and suitable for different scenarios. Therefore, the basic idea of RNBB can be divided into the four following components:

Analysis of the Network Topology. RNBB defines two types of nodes; center and border nodes. A node is defined as a center node in the network, if it has equal or more neighbors than a predefined threshold (D). A node is called border node, if it has fewer neighbors than the threshold. Differentiating between border and center nodes, we define two different behaviors for forwarding. According to the density principle, the selection of the forwarding nodes among the center nodes is based on analyzing the neighbor knowledge and the received messages count. This way, the inaccurate neighbor knowledge due to high density will not influence the performance of the algorithm. In contrast, the forwarding task in the border nodes is defined in RNBB by analyzing the neighbor knowledge which is commonly considered accurate in the border area.

Adaptive Scheduling of Packets. An important design consideration of the RNBB algorithm is the selection of the delay time after which the node will forward the broadcast message. Efficient packet scheduling ensures low likelihood of a collision occurring during multiple transmissions of the message. Moreover, such scheduling will allow the nodes to receive redundant packets and assess whether to rebroadcast or not. In RNBB, each node has to define a Random Assessment Delay (RAD) before any forwarding of a broadcast message. However, a well selection of the RAD is the main issue because it keeps packet collisions at a minimum. Therefore, RNBB uses a special idea which decreases the likelihood of selecting two identical RADs among neighboring nodes. In RNBB, the RAD selection for a forwarding node depends on the following terms:

Algorithm 1. Calculate RAD

Require: Locally stored IP address of all neighbors in set: N
Ensure: Compute RAD
1: $N \leftarrow N \cup \{MyIP\}$;
2: $T \in [T_{min}, T_{max}]$;
3: $T_{step} \leftarrow (T_{max} - T_{min})/NN$;
4: $M \leftarrow Sort(N)$;
5: $P\_Ind \leftarrow the\ order\ of\ MyIp\ in(M)$;
6: $RAD_{min} \leftarrow T_{step} \times P\_Ind + T_{min}$;
7: $RAD_{max} \leftarrow RAD_{min} + T_{step}$;
8: $RAD \leftarrow Random[RAD_{min}, RAD_{max}]$;

- NN: each node maintains NN specifying its number of neighboring nodes. When a node transmits a broadcast message to its neighbors it includes its own NN in the message.
- (P_Ind): indicates the order of the node's IP address in the neighbor list which is obtained by sorting upwards the IP addresses of the actual neighboring nodes.

The basic idea for decreasing the likelihood of selecting two identical RADs is divided into two selection phases; the initial phase and adaptive phase. Algorithm 1 shows the calculation procedure of RAD in initial phase. In this phase, RNBB divides the predefined time range into equal sub-ranges, wherein, the number of these sub-ranges in this node depends on its NN number. Then, the node uses its P_Ind number to select its sub-range from which the RAD should be randomly selected. In the adaptive phase RNBB allows the node which has received a broadcast message to re-adjust the predefined RAD (from the initial phase). The node will do that by analyzing the copies of the broadcast message forwarded by the neighboring nodes. This means that the RNBB uses a reactive algorithm to re-adjust the RAD after every reception of a copy of the message. Depending on a comparison between its NN and the other NN included in the received message, the RAD of this node can be decreased or increased by a suitable value. In RNBB, if its NN is bigger than the received NN, the RAD should be decreased by a value proportional to the difference between the two numbers. When its NN is equal or smaller than the received NN, the RAD is increased by the calculated value. However, increasing or decreasing the RAD should not exceed predefined upper and lower boundaries of the time range. This way the RNBB algorithm ensures that the nodes which have more neighbors forward the message faster than the node with less neighbors. This in turn enables more coverage every time the message is forwarded and allows the nodes to receive redundant packets and assess whether to rebroadcast.

Neighbor List Coding. All algorithms following Neighbor-Based methods, such as the SP algorithm, are based on collecting information about the neighbors of the sender and receiver of a broadcast message. In dense networks, such

Algorithm 2. Half line algorithm

Require: Locally stored IP address of all neighbors in set: $\{A(a_0, a_1, a_2..a_e)\}$
Ensure: Compute $Ncode(sum_x, sum_y, sum_z)$
1: **if** $Odd(e)$ **then**
2: $a_{e+1} \leftarrow a_e$;
3: $A \leftarrow A \cup \{a_{e+1}\}$;
4: **end if**
5: $P : (p_0, p_1, ..p_m), m = e/2, p_i \leftarrow (a_{2\times i}, a_{2\times i+1}), (0 \leq i \leq e)$
6: $F(u, v) \leftarrow$ *middle point between two points u and v*;
7: $mp_o(x_0, y_0) \leftarrow F(\ p_0,\ p_1)$;
8: $mp_k(x_k, y_k) \leftarrow F(mp_{k-1}, p_{k+1}), (0 \leq k \leq i-1)$
9: $sum_x = \sum_{i=0}^{k} x_i$;
10: $sum_y = \sum_{i=0}^{k} y_i$;
11: $sum_z = \sum_{i=0}^{k} \frac{y_i}{x_i}$;

neighbors' information may be too big to fit into a single packet. This in turn leads to additional packet collisions. To solve this problem, RNBB suggests reducing the size of the neighbor information by coding the data which represents the neighbor knowledge when the size of this data is larger than a threshold. For performance considerations, this threshold should not be bigger than half the size of the average broadcast packet size in MANETs, which is preferred to be about 512 bytes [1]. In other words, if the neighbor list is bigger than a threshold, a node, instead of sending a list of its neighbor set, has to send a fixed number of bits (code) representing this information. A new algorithm called Half-line is introduced, which has a very low likelihood of producing the same code (referred to *Ncode* in Algorithm 2) by two neighboring nodes having different neighbors. Upon reception of a broadcast message, each node has to compare the received code with its own code. If the codes are equal, the node receiving the message knows that there is no need to forward the message. This coding approach reduces protocol overhead and collisions in dense networks.

Reliable Recovery of Packet Loss. Although RNBB is designed for reducing the broadcast redundancy (signaling overhead) in MANETs, the reliability for the delivery of a broadcast message is considered an essential design feature. A reliable broadcast operation means that the broadcast packet should be disseminated to all nodes in the network. In MANETs, many issues may prevent some nodes from receiving a broadcast message, such as interference and movement. Therefore, the sender should use a reliable method, such as retransmitting the packet, to increase the delivery ratio of the transmission. In RNBB, the basic idea for reliable broadcasting is based on the overhearing approach. In contrast to the algorithm presented in [16], the reliable reception of a broadcast packet in RNBB is done when the sender receives a number of packet copies from any of its neighbors. So, the sender is not bounded to receive from certain nodes. RNBB suggests that each node which has broadcasted a message has to wait for a predefined timeout referred to as Retransmission Timeout (RTO). During

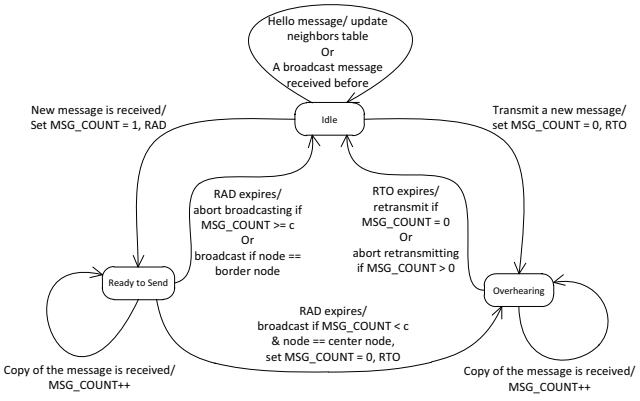


Fig. 1. State transitions for RNBB when a `hello` or `broadcast` message is received

this period the node is waiting to receive a copy of its message. When RTO expires without receiving any copy from any of the neighboring nodes, the node retransmits the message once again (for a maximum of twice retransmissions). To avoid the retransmission of the broadcast message without getting additional coverage, RNBB doesn't allow the border nodes to perform overhearing. Thus, the number of messages can be kept at a minimum while ensuring high reliability of delivering the broadcast message in MANETs.

3.2 Algorithm

In RNBB, a node handling a broadcast message is defined to be in one of three states; "Idle", "Ready to send" and "Overhearing". Figure 1 depicts the state transition diagram of the node during the broadcast process:

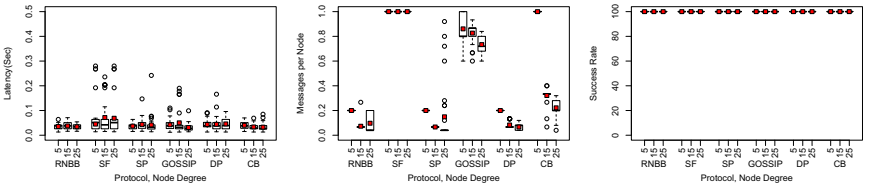
- When a node switches to the RNBB algorithm, it enters the "Idle" state. Upon reception of a hello message, the node updates its own neighbor list, and it remains in the state. Also, the state will not be changed if the node receives a broadcast message which has been received before. However:
  - If the node receives a new broadcast message during the "Idle" state, it adds a new entry to a certain list called the message checklist representing trace information of the received messages. The parameters of each entry are the number of message copies (MSG_COUNT), the timeout (RAD) after which the node will decide to forward the message and a set of uncovered neighbors (UCN) which are constructed from the node's neighbors list (N) excluding the neighbor set included in the broadcast message. After that, the state is changed to "Ready to Send";
  - If the node needs to send a new broadcast message during the time in "Idle" state, it resets the MSG_COUNT and initiates the RTO timeout. Then the node changes the handle process to the "Overhearing" state;

– During "Ready to send" state, the mobile node waits for the expiration of the RAD timeout and it will increase the MSG_COUNT value by 1 on every reception of a copy of the message sent by the neighboring nodes. When the RAD expires, the node checks the MSG_COUNT value which indicates the number of received copies:
  - If the node has received the message equal or more than a pre-defined threshold (D), the node knows that it doesn't need to forward the message and it changes the process status to "Idle" state. However, if the MSG_COUNT is less than D and the node is a border node, then the node has to forward the message;
  - Otherwise, the node broadcasts the message, sets MSG_COUNT=0, initiates the re-transmission timeout (RTO) and changes to "Overhearing";
– During the "Overhearing" state, if the node receives the same message again, it increments the MSG_COUNT by (1). When the RTO expires, the node changes the process status to the "Idle" state. However, the node has to do one of two actions in this case. The node has to retransmit the message if the node has not received the message during the "Overhearing" state. Otherwise, the node aborts the retransmission;

4 Evaluation

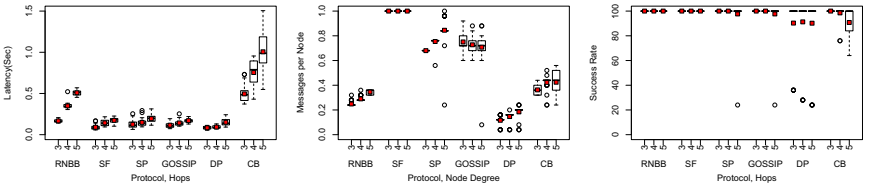
In this section, we evaluate the performance of our proposed algorithm using ns-2 with different network characteristics. We have evaluated our algorithm across different types of scenarios representing a wide variety of network topologies. We compare the performance of RNBB with Simple-Flooding (SF), Self-Pruning (SP) [10], Dominant-Pruning (DP) [11], Gossip3 (GOSSIP) [17] and Counter-Based (CB) [4] algorithms which are representative wireless broadcast protocols with well tuned ns-2 implementations. The simulation experiments are made on different kinds of scenarios, such as static and mobile scenarios. In RNBB the simulative value of (T_{min} & T_{max}) within the initial and adaptive phases are set to (50 & 450ms) and (20 & 50ms) respectively, wherein, RTO is set to 450ms. The threshold value D = 5 has been chosen as optimal value from different experiments applied on the assumed scenarios, a similar value for CB algorithm has been suggested by the experiment applied in [18].

We have run a large number of experiments and the results indicate that RNBB performs well across all experiments, while other protocols tend to perform well on some experiments but poorly on others. For our evaluation, we have selected three basis metrics. First, we analyzed the latency, which describes the time difference between the initiating node sending the broadcast and the last node receiving the broadcast. We measure the success rate as the percentage of nodes that have received the broadcast message to the total number of nodes in the network. The success rate accurately describes the capability of the protocol to ensure correct broadcast delivery. Furthermore, the number of messages broadcasted by each node is evaluated, which indicates how many redundant messages have been sent to inform all the nodes in the network of the new configuration. All results discussed in the following sections are shown as boxplots.



(a) Coverage latency (b) Messages per node (c) Success rate

Fig. 2. Performance of the different algorithms in the static single-hop scenario



(a) Coverage latency (b) Messages per node (c) Success rate

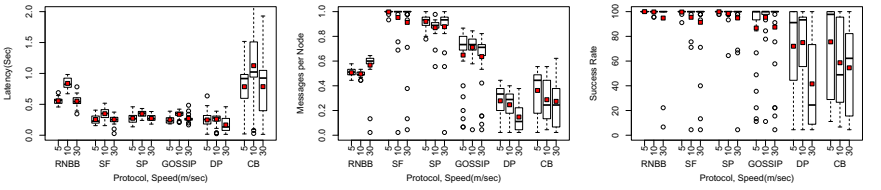
Fig. 3. Performance of the different algorithms in the static multi-hop scenario

The boxplots have been chosen because they are more robust in the presence of outliers than the classical statistics based on the normal distribution. Basically, for each data set, a box is drawn from the first quartile to the third quartile, and the median is marked with a thick line. Additional whiskers extend from the edges of the box towards the minimum and maximum of the data set. Data points outside the range of the box and whiskers are considered outliers and drawn separately. Additionally, the mean value is depicted in the form of a small filled square.

4.1 Static Scenarios

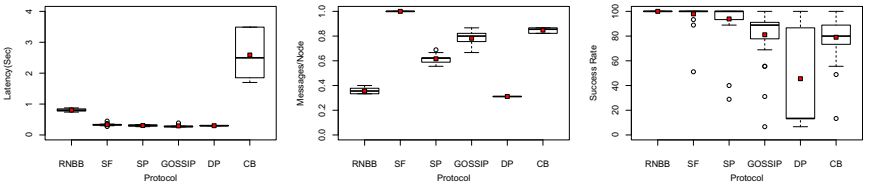
These scenarios are designed to study the effects of dense (single-hop) as well as sparse (multi-hop) networks on the performance of the broadcasting algorithms. In single-hop scenarios, the number of hops is fixed but the network size varies, and in multi-hop scenarios the opposite is true.

Single-Hop Network. The first set of experiments compared the performance of the broadcasting protocols against increasing densities, while keeping all nodes in the communication range of each other. We varied the node degree between 5 and 25. Figure 2 shows that all protocols have a 100% delivery ratios with low delays, which is a consequence of the fact that one broadcast is enough to notify all nodes. Nevertheless, they have different communication overhead, for example Simple-Flooding (SF) has the highest delay due to congestion, which is illustrated by the large increase in messages per node for SF in comparison to the other algorithms.



(a) Coverage latency (b) Messages per node (c) Success rate

Fig. 4. Performance of the different algorithms in the mobility scenario



(a) Coverage latency (b) Messages per node (c) Success rate

Fig. 5. Performance of the different algorithms in the special scenario

Multi-Hop Network. The second set of experiments evaluates the performance of the broadcasting protocols as the number of hops increases while keeping the network size constant. We tested scenarios involving 3, 4 and 5 hops. Figure 3 shows the results for the multi-hop scenario. In comparison to other algorithms, CB and RNBB require more time in comparison to the other algorithms, because they wait for RAD before they rebroadcast the message. The use of RAD leads to increase the latency when the number of hops is increased. In Gossip3, because the RAD has only been used by the nodes that decide not to rebroadcast (probability of aborting broadcast is 25%), its latency is still very low (between 200 to 400 ms) when increasing the number of hops compared to CB and RNBB. Similar to density scenarios, SF and Gossip3 provide the highest redundancy as shown in Figure 3(b). The figure shows that SF and RNBB provide 100% successful packet delivery in all multi-hop scenarios, because in SF every node has to rebroadcast the message once and in RNBB we have used the overhearing mechanism. Nevertheless, RNBB transmits much less messages.

4.2 Mobility Scenarios

We have performed a series of experiments to explore the effect of mobility on the performance of RNBB. The mobility scenarios are following a node mobility pattern generated using the random way point model [19]. The speed of each mobile node is set to the mean values [5 m/s, 10 m/s, and 30 m/s] with a mean pause time of one second. The simulation area of all mobility scenarios is 1000m x 1000m and the simulation time is set to 100 sec. The maximum number of nodes in each scenario is 50 nodes and each of these nodes has a transmission range of

250m. Figure 4 presents the comparison of the proposed broadcast algorithms under different node speeds. The algorithms which use a RAD (CB and RNBB algorithms) in the broadcast process result in a noticeably high latency where CB provides higher latency than RNBB because of the proper selection of the RAD in RNBB. RNBB provides 100% successful packet delivery ratio when the nodes move in low speed (5 m/s). However, this ratio is not perfect at 10 and 30 m/s. RNBB provides high successful packet delivery ratio due to its reliability, where some of the forwarding nodes are able to rebroadcast the message to its neighbors once or more as explained before.

4.3 Special Scenario

The network in this scenario is a combination of multi-hop and density scenarios with the existence of bottleneck nodes. A single node that connects two parts of the network is considered as a bottleneck node because if it fails to send the broadcast message, a portion of the network will not receive the broadcast. Such a scenario explores the efficiency of using a broadcasting algorithm in a heterogeneous topology.

Figure 5 presents the comparison of the proposed broadcast algorithms under the special scenario. As can be expected, CB and RNBB require more time in comparison with other algorithms due to the use of the waiting time (RAD). However, RNBB has lower latency than CB because the RAD value in RNBB is well selected by adapting its value according to the neighbors' knowledge information. Furthermore, RNBB can readjust the RAD whenever the node receives an additional copy of the message. Moreover, the low number of forwarding nodes selected in RNBB in comparison with CB, mitigates the possible contentions and collisions in the network. RNBB provides the highest successful packet delivery ratio (100%) due to the use of the reliability function in which the forwarding nodes in dense areas are able to rebroadcast the message twice if its neighbors don't hear the message from the first broadcast.

5 Conclusion

RNBB is a novel broadcasting protocol, which is designed as a hybrid scheme combining Self Deterministic and Probability Based approaches. RNBB performs well because of the following design components:

- Analysis of the topology: center and border nodes behave differently.
- Packet scheduling: each node schedules its packets autonomously with a very low collision probability.
- Neighbor list coding: a short code contains necessary neighbor information.
- Overhearing: efficient mechanism to rebroadcast the message if necessary.

In this paper, we have evaluated RNBB in the context of Ad-hoc wireless networks. The simulation results show that RNBB is able to achieve low latency, low redundancy and close to perfect packet delivery ratio under a wide variety

of network topologies and in comparison to the other algorithms. This can be seen in the results of the special scenario which is designed to study the impact of different issues affecting broadcasting in MANETs.

References

1. Basagni, S., Conti, M., Giordano, S., Stojmenovi, I.: Mobile Ad Hoc Networking. John Wiley & Sons, Inc., New Jersey (2004) ISBN: 978-0-471-37313-1
2. Johnson, D.B., Maltz, D.A., Broch, J.: DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks. In: Perkins, C.E. (ed.) Ad Hoc Networking, pp. 139–172. Addison-Wesley, Boston (2001)
3. Perkins, C.E., Belding-Royer, E.M., Das, S.R.: Ad Hoc On-Demand Distance Vector (AODV) Routing. RFC 3561 (July 2003)
4. Al-Humoud, S.O., Mackenzie, M.: A Mobility Analysis of Adjusted Counter-Based Broadcast in MANETs. In: 9th Symposium (PGNET), England, pp. 51–62 (2008)
5. Mohammed, A., Ould-Khaoua, M., Mackenzie, L.M.: An Efficient Counter-Based Broadcast Scheme for Mobile Ad Hoc Networks. In: Wolter, K. (ed.) EPEW 2007. LNCS, vol. 4748, pp. 275–283. Springer, Heidelberg (2007)
6. Ni, S.-Y., Tseng, Y.-C., Yuh-Shyan, C.: The Broadcast Storm Problem in a Mobile Ad Hoc Networks. In: 5th ACM/IEEE Conf. (Mobicom), USA, pp. 151–162 (1999)
7. Williams, B., Camp, T.: Comparison of Broadcasting Techniques for Mobile Ad Hoc Networks. In: 3rd ACM Intern. Symp. (Mobihoc), Switzerland, pp. 194–205 (2002)
8. Karthikeyan, N., Palanisamy, V.: Performance Comparison of Broadcasting Methods in Mobile Ad Hoc Network. Int. Journal IJFGCN 2(2), 47–58 (2009)
9. Haas, Z., Halpern, Y., Li, L.: Gossip-Based Ad Hoc Routing. In: 21st IEEE Annual Conference (INFOCOM), USA, pp. 1707–1716 (June 2002)
10. Wu, J., Dai, F.: Broadcasting in Ad Hoc Networks Based on Selfpruning. In: 22nd IEEE Annual Conference (INFOCOM), USA, pp. 2240–2250 (2003)
11. Dai, F., Wu, J.: Distributed Dominant Pruning in Ad Hoc Networks. In: The IEEE International Conference on Communications (ICC), USA, pp. 353–357 (2003)
12. Ovalle-Martínez, F.J., Nayak, A., Stojmenović, I., Carle, J., Simplot-Ryl, D.: Area Based Beaconless Reliable Broadcasting in Sensor Networks. In: Nikoletseas, S.E., Rolim, J.D.P. (eds.) ALGOSENSORS 2006. LNCS, vol. 4240, pp. 140–151. Springer, Heidelberg (2006)
13. Tang, K., Gerla, M.: Mac Reliable Broadcast in Ad Hoc Network. In: The Military Communications Conference (MILCOM 2001), Vienna, pp. 1008–1013 (2001)
14. Bi, Y., Cai, X.L., Shen, X., Zhao, H.: Efficient and Reliable Broadcast in Intervehicle Communication Networks: A cross-layer approach. IEEE Transaction on Vehicular Technology 59(5), 2404–2417 (2010)
15. Farnoud, F., Valaee, S.: Repetition-Based Broadcast in Vehicular Ad Hoc Networks in Rician Channel with Capture. In: IEEE Conf. (INFOCOM), USA, pp. 1–6 (2008)
16. Lou, W., Wu, J.: Toward Broadcast Reliability in Mobile Ad Hoc Networks with Double Coverage. IEEE Trans. on Mobile Computing 6(2), 148–163 (2007)
17. Haas, J.Z., Halpern, Y.J., Li, L.E.: Gossip-Based Ad Hoc Routing. IEEE/ACM Transactions on Networking 14(3), 479–491 (2006)
18. Arango, J., Efrat, A., Ramasubramanian, S.: Retransmission and Backoff Strategies for Wireless Broadcasting. J. Ad Hoc Networks 8(1), 77–95 (2010)
19. Bai, F., Sadagopan, N., Helmy, A.: The IMPORTANT Framework for Analyzing the Impact of Mobility on Performance of Routing for Ad Hoc Networks. J. AdHoc Networks 1(4), 383–403 (2003)

Exploiting Opportunistic Overhearing to Improve Performance of Mutual Exclusion in Wireless Ad Hoc Networks

Ghazale Hosseinabadi and Nitin H. Vaidya

Department of ECE and Coordinated Science Lab.
University of Illinois at Urbana-Champaign
Urbana, IL, 61801, USA
{ghossei2,nhv}@illinois.edu

Abstract. We design two mutual exclusion algorithms for wireless networks. Our mutual exclusion algorithms are distributed token based algorithms which exploit the opportunistic message overhearing in wireless networks. One of the algorithms is based on overhearing of token transmission. In the other algorithm, overhearing of both token and request messages is exploited. The design goal is to decrease the number of transmitted messages and delay per critical section entry using the information obtained from overheard messages.

Keywords: Wireless networks; opportunistic overhearing; mutual exclusion.

1 Introduction

A wireless ad hoc network is a network in which a pair of nodes communicates by sending messages over wireless links. Wireless ad hoc networks have fundamentally different characteristics from wired distributed networks, mainly because the wireless channel is a shared medium and messages sent on the wireless links might be overheard by the neighboring nodes. The information obtained from the overheard messages can be used in order to design distributed algorithms, for wireless networks, with better performance metrics. Although existing distributed algorithms will run correctly on top of wireless ad hoc networks, our contention is that efficiency can be obtained by developing distributed algorithms, which are aware of the shared nature of the wireless channel. In this paper, we present distributed mutual exclusion algorithms for wireless ad hoc networks.

Distributed processes often need to coordinate their activities. If a collection of processes share a resource or collection of resources, then often *Mutual Exclusion (MUTEX)* is required to prevent interference and ensure consistency when accessing the resources. In a *distributed* system, we require a solution to *distributed* mutual exclusion. Consider users who update a text file. A simple means of ensuring that their updates are consistent is to allow them to access

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 162–173, 2012.

the file only one at a time, by requiring the editor to lock the file before updates can be made. A particularly interesting example is where there is no server, and a collection of peer processes must coordinate their access to shared resources amongst themselves.

Mutual Exclusion is a well known problem in distributed systems in which a group of processes require entry into the *critical section (CS)* exclusively, in order to perform some critical operations, such as accessing shared variables in a common store or accessing shared hardware. Mutual exclusion in distributed systems is a fundamental property required to synchronize access to shared resources in order to maintain consistency and integrity. To achieve mutual exclusion, concurrent access to the CS must be synchronized such that at any time only one process can access the CS. The proposed solutions for distributed mutual exclusion are categorized into two classes: token based [1], [2], [3] and permission based [4], [5], [6]. In token based MUTEX algorithms, a unique token is shared among the processors. A processor is allowed to enter the CS only if it holds the token. In a permission based MUTEX algorithm, the processor that requires entry into the CS must first obtain the permissions from a set of processors.

In this paper, we design mutual exclusion algorithms for wireless networks. Most of the existing MUTEX algorithms are designed for typical wired networks [1],[2],[4],[5],[6]. Design of mutual exclusion algorithms for mobile ad hoc networks had received some interest in the past few years [3], [7]. Although the underlying network in these algorithms is wireless, the proposed algorithms are only mobility aware solutions, where the goal is to deal with the problems caused by node mobility, such as link failures and link formations. In this work, we show that the broadcast property of the wireless medium can be exploited in order to improve the performance of the MUTEX algorithms in wireless networks. To the best of our knowledge, this work is the first in which opportunistic overhearing is exploited to improve the performance of MUTEX in wireless networks.

In this work, we present two token based mutual exclusion algorithms that are designed for wireless networks. Network nodes communicate by transmitting unicast messages. Since the channel is wireless, a unicast message transmitted from node i to node j might be overheard by neighbors of node i, for example node k. In this case, node k is not the intended receiver of the message, but it has overheard the message due to the shared nature of the wireless medium. We design our algorithms such that the neighboring nodes that overhear messages can learn more recent information about the current status of the algorithm.

We call our algorithms *Token Overhearing Algorithm (TOA)* and *Token and Request Overhearing Algorithm (TROA)*. *TOA* is based on the MUTEX algorithm designed by Raymond [1]. In Raymond's algorithm, messages are transmitted over a static spanning tree of the network. *TOA* is based on overhearing of the token transmission and the spanning tree maintained by the algorithm changes when token transmission is overheard by the neighboring nodes. *TROA* is based on Trehel-Naimi's algorithm [2]. In Trehel-Naimi's algorithm, when a node requires entry to the CS, the node sends a request message to the last known owner of the token. In *TROA*, overhearing of both request and token

messages are exploited in order to obtain recent information about the latest token holder in the network. The performance metrics that we aim to improve in this work are the number of transmitted messages and delay per CS entry. Our mutual exclusion algorithms satisfy three correctness properties: 1) *Mutual Exclusion*: at most one processor is in the CS at any time; 2) *Deadlock free*: if any processor is waiting for the CS, then in a finite time some processor enters the CS; 3) *Starvation free*: if a processor is waiting for the CS, then in a finite time the processor enters the CS.

The remainder of this paper is organized as follows: We first describe the network model in Section 2. In Section 3, we present *TOA*. *TROA* is described in Section 4. Simulation results are presented in Section 5.

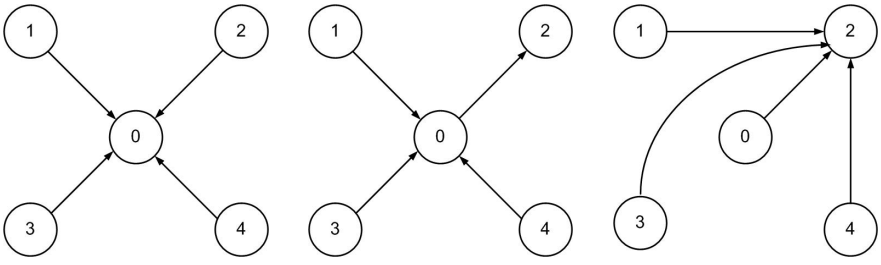
2 Network Model

We consider a network of n nodes, communicating by message passing in a wireless ad hoc network. Each node has a unique identifier, i, $0 \leq i \leq n-1$. Messages transmitted in the network are unicast messages. We assume that lower layers of the network, such as MAC layer and transport layer, ensure reliable delivery of unicast messages. To ensure reliability, retransmission mechanism is used in lower layers in case packets are lost due to noise or interference. Since the network is wireless, a unicast message from node i to node j might be overheard by neighbors of node i, such as node k. We assume that if such an opportunistic overhearing happens, node k does not discard the overheard message; instead it uses the information included in the message. We do not assume that the unicast message of node i to node j is delivered reliably to the neighbors of node i. Instead, the overhearing is opportunistic, meaning that if the neighboring nodes overhear messages not intended for them, they exploit the information included in the messages. We assume that network nodes do not fail and each node is aware of the set of nodes with which it can directly communicate.

3 Token Overhearing Algorithm (TOA)

Token Overhearing Algorithm (TOA) is based on Raymond's algorithm [1]. Raymond designed a distributed token based mutual exclusion algorithm in which requests are sent over a static spanning tree of the network, towards the token holder. The tree is maintained by logical pointers distributed over the nodes and directed to the node holding the token. At each time instance, there is a single directed path from each node to the token holder. When a node has a request for the token, a sequence of request messages are sent on the path between the requesting node and the token holder. The token is sent back over the reverse path to the requesting node. The direction of the links over which the token is transmitted is reversed. In this way, at each time instance, all edges of the tree point towards the token holder.

Similar to Raymond's algorithm, *TOA* uses a spanning tree of the network over which messages are passed. But unlike Raymond's algorithm, the spanning



(a) Initial tree (b) Raymond's algorithm (c) TOA

Fig. 1. Example execution of Raymond's algorithm and *TOA*

tree in *TOA* is dynamic and changes if token transmission is overheard by the neighboring nodes. Sender and receiver of the token are specified in the token message. When token is sent from node i to node j, any other node k that overhears transmission of the token, changes its parent in the tree.

3.1 Example of Algorithm Operation

An example execution of Raymond's algorithm and *TOA* is illustrated in Figure 1. The network is a wireless ad hoc network composed of five nodes, node 0-4. We assume that the network is single-hop, in which all nodes are in the communication range of each other. Figure 1(a) shows the initial spanning tree of the network, where the token holder is 0. We consider a case where 2 requires entry to the CS and sends a request message to 0. We assume that there is no other pending request in the network. When 0 receives the request of 2, it sends the token to 2. Figure 1(b) shows the spanning tree in Raymond's algorithm after the token is sent to 2. At this point, the direction of the edge between 0 and 2 is reversed. In Figure 1(b), nodes 1, 3 and 4 are two hops away from 2. If any of these nodes, for example node 1, requests to enter the CS, two request messages are sent, one request message from 1 to 0 and one from 0 to 2. It then takes two messages to send the token from 2 to 1. So, total of four messages are sent so that 1 can enter the CS.

We now describe how *TOA* performs when the nodes are initially configured as depicted in Figure 1(a) and node 2 requires CS entry. Like Raymond's algorithm, when 0 receives the request of 2, it sends the token to 2. Since all nodes are in the communication range of each other, token transmission from 0 to 2 might be overheard by 1, 3 and 4. We consider the best scenario for our algorithm, in which all nodes 1, 3 and 4 overhear the token transmission. As a result, 1, 3 and 4 point to 2. Figure 1(c) shows the spanning tree in our algorithm when the token is sent to 2. In this figure, nodes 1, 3 and 4 are only one hop away from the token holder, node 2. If any of these nodes requests entry to the CS, only two messages are transmitted, one request message and one token message. This example shows that in single-hop wireless networks and when requests for the token are initiated separate enough in time, *TOA* might perform

better than Raymond's algorithm. The reason is that as a result of messages overhearing, nodes might be aware of the current token holder in which case they send their requests directly to the token holder. In single hop networks, if every node overhears token transmission, *TOA* is optimal and only two messages, one request message and one token message, are transmitted per CS entry. On the other hand, in Raymond's algorithm four messages might be sent for one CS entry, in single-hop networks. The reason is that, although the token holder and the requesting node are in the communication range of each other, they might not communicate directly, rather they exchange messages through the initial root (e.g. node 0 in this example), simply because Raymond's algorithm uses a static spanning tree.

3.2 Overview of Token Overhearing Algorithm (TOA)

Token Overhearing Algorithm (*TOA*) is based on Raymond's algorithm [1], which is a well-known MUTEX algorithm. Due to the lack of space, we do not present the details of Raymond's algorithm here. We just describe our modification to Raymond's algorithm which is *OverhearToken* (procedure 3.2.1). *OverhearToken* is executed when a node k overhears the transmission of `TOKEN` from *sender* to *receiver*. In this case, k is not the intended receiver of the message, but it has overheard the message. $parent_k$ in the tree becomes *receiver* if k and *receiver* are immediate neighbors, otherwise k chooses *sender* as its parent.

3.2.1 *OverhearToken*

```
1: if type is TOKEN then
2:   if receiver is my neighbor then
3:     parent = receiver of the message
4:   else
5:     parent = sender of the message
```

TOA has three correctness properties; safety, deadlock free and lockout free. The proofs of correctness are omitted here due to the lack of space.

4 Token and Request Overhearing Algorithm (TROA)

We design another mutual exclusion algorithm, called *Token and Request Overhearing Algorithm (TROA)* for wireless networks. *TROA* is based on Trehel-Naimi's algorithm [2]. The objective in designing *TROA* is to find a MUTEX algorithm in which overhearing of both token and request messages is exploited in order to improve the performance. We note that *TOA* is only based on overhearing of token transmission.

Trehel-Naimi's algorithm [2] is a token-based algorithm which maintains two data structures: (1) A dynamic tree structure in which the root of the tree is the last node that will hold the token among the current requesting nodes. This tree is called the *last* tree. Each node i has a local variable *last* which points to the

last probable token holder that node i is aware of. (2) A distributed queue which maintains requests for the token that have not been answered yet. This queue is called the *next* queue. Each node i keeps the variable *next* which points to the next node to whom the token will be sent after i releases the CS.

In Trehel-Naimi's algorithm, when a node i requires entry to the CS, it sends a request to its *last* and then changes its *last* to `null`. As a result, i becomes the new root of the *last* tree. When node j receives the request of node i, one of these cases happens: 1) j is not the root of the tree. It forwards the request to its *last* and changes its *last* to i. 2) j is the root of the tree. If j holds the token, but does not use it, it sends the token to i. If j is in the CS or is waiting for the token, j sets its *next* to i. Whenever j exits the CS, it sends the token to $next = i$.

Trehel-Naimi's algorithm is designed for wired networks in which transmitted messages are not overheard by the neighboring nodes. We modify the algorithm to perform better in wireless networks by exploiting the broadcast property of wireless networks. In *TROA*, nodes can learn more recent information about the last token holder in the network by overhearing of messages not intended for them.

4.1 Data Structures, Messages and algorithm procedures

Since *TROA* and Trehel-Naimi's algorithm are different from each other in so many ways, we describe the details of *TROA* in this section. In *TROA*, each node maintains the following data structures:

- *privilege*: *privilege* is `true` if the node holds the token, and `false` otherwise.
- *requestingCS*: when a node initiates request for the token, its *requestingCS* is set to `true`. *requestingCS* becomes `false` when the node releases the CS.
- *last*: when a node wants to enter the CS, it sends a request to its *last*. *last* of a node might change when the node receives or overhears messages.
- *next*: When a node that is waiting for the token receives a request message from another node, it saves the *initiator* of the request message in its *next*. Later, when the node releases the CS, it sends the token to *next*.
- *numCSEntry*: *numCSEntry* of node i denotes how many times CS entry has happened in the network such that node i is aware of.
- *numReceivedRequests*: it denotes how many REQUEST messages are received by a node while the node was waiting for the TOKEN.

numCSEntry and *numReceivedRequests* are used as counters to determine if a node should change its *last* when it overhears messages. We will present more details later, when we describe algorithm procedures.

There are two types of messages in the algorithm, `REQUEST` and `TOKEN`. A `REQUEST` message includes the following information.

- *initiator*: it is the id of the node that has initiated the request for the token.

- *destination*: it denotes the final destination of the message. Since we consider the general case of multi-hop networks, *destination* is not necessarily a neighbor of *initiator*. In this case, the message is routed on the shortest path between *initiator* and *destination*.
- *numberCSEntry*: when a node transmits a message, it writes its *numCSEntry* in *numberCSEntry* part of the message. *numberCSEntry* is used by nodes that overhear the message to determine if their *last* should be changed or not.

A message of type TOKEN includes the following information.

- *destination*: it denotes the final destination of the token.
- *numberCSEntry*: As we explained before, when a node transmits a message, it includes its *numCSEntry* in *numberCSEntry* part of the message.

We now present the procedures of *TROA*.
Initialization: Procedure 4.2.1 is executed at the beginning of the algorithm by every node i to set the initial value of i's data structures.
RequestCS: Procedure 4.2.2 is called when a node wants to enter the CS. If the node holds the token, it enters the CS. otherwise, it sends a REQUEST to its *last*.

4.2.1 *Initialization*

```
1: last = INITIAL-TOKEN-HOLDER
2: next = null
3: requestingCS = false
4: numReceivedRequests = 0
5: numCSEntry = −1
6: if last == myId then
7:    privilege = true
8:    last = null
9:    numCSEntry = 0
10: else
11:   privilege = false
```

4.2.2 *RequestCS*

```
1: requestingCS = true
2: if (privilege == false) then
3:    send REQUEST to last
4:    last = null
5: else
6:    enter CS
```

OverhearRequest : When a REQUEST message from node i to node j is overheard by node k, Procedure 4.2.3 is executed, in which if some conditions hold, node k changes its *last* to *initiator* of the message.

4.2.3 *OverhearRequest*

```
1: if numberCSEntry > numCSEntry+numReceivedRequests+1 and last ! =
   null and requestingCS == false then
2:    last = initiator
3:    numCSEntry = numberCSEntry−1
4:    numReceivedRequests = 0
```

OverhearToken : When node k overhears the transmission of TOKEN from node i to node j, Procedure 4.2.4 is executed. **4.2.4** *OverhearToken*

```
1: if numberCSEntry > numCSEntry + numReceivedRequests and last != null and requestingCS == false then
2:   last = destination
3:   numCSEntry = numberCSEntry
4:   numReceivedRequests = 0
```

ReceiveToken : Procedure 4.2.5 is executed when TOKEN is received at its final destination, *destination*. Intermediate nodes on the path that forward the message do not run this procedure.
ReleaseCS: Procedure 4.2.6 is executed when a node exits the CS.

4.2.5 *ReceiveToken*

```
1: privilege = true
2: numCsEntry = numberCsEntry +1
3: numReceivedRequests = 0
4: enter CS
```

4.2.6 *ReleaseCS*

```
1: requestingCS = false
2: if next != null then
3:   privilege = false
4:   send TOKEN to next
5:   next = null
```

ReceiveRequest : Procedure 4.2.7 is executed when a REQUEST message is received at its final destination, *destination*.

4.2.7 *ReceiveRequest*

```
1: if last==null then
2:   if requestingCS==true then
3:     next=initiator
4:   else
5:     privilege=false
6:     send TOKEN to initiator
7: else
8:   send request to last
9:   numReceivedRequests++
10: last=initiator
```

TROA has three correctness properties; safety, deadlock free and lockout free. The proofs of correctness are omitted here due to the lack of space.

5 Simulations

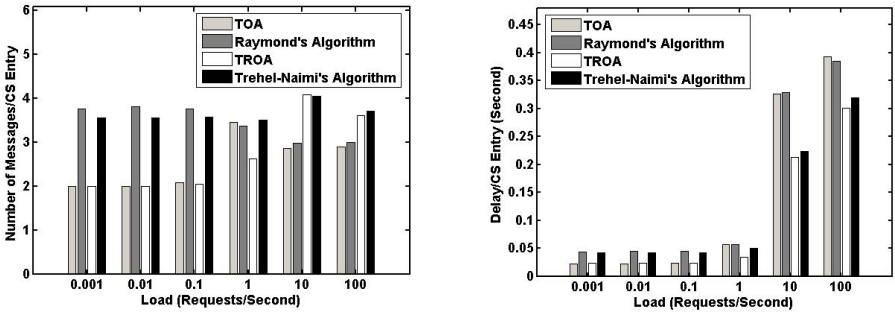
We run simulations to measure the performance of *TOA* and *TROA*. We also simulate Raymond's algorithm and Trehel-Naimi's algorithm to find the improvements obtained by message overhearing. In our simulations, network nodes are placed uniformly at random in a square area. The node closest to the center of the area is chosen as the initial root of the tree. Messages sent in the network are unicast messages. In order to implement message overhearing, we change the 802.11 MAC layer of ns-2. In the current implementation of ns-2, packets that are received in the MAC layer of node i with MAC destination address different

from i's MAC address are dropped. We change ns-2 so that such packets are not dropped, and they are delivered to the application layer of node i. In this work we measure two performance metrics, which we call the cost of the algorithms: 1) Number of messages per CS entry: it is equal to the number of messages transmitted in the network per entry to the CS. 2) Delay per CS entry: the delay is measured as the interval between the time at which a node initiates a request to enter the CS and the time at which the node enters the CS.

Requests for CS entry are assumed to arrive at a node according to a Poisson distribution with rate λ requests/second. When λ is small, no other processor is in the CS when a processor makes a request to enter the CS. In this case, the network is said to be lightly loaded. When λ is large, there is a high demand for entering the CS which results in queueing up of the requests and the network is said to be heavily loaded. The time to execute the CS is 10^{-5} second. Figures 2-4 plot number of messages and delay per entry to the CS against λ in three example networks. λ increases from 10^{-3} to 10^{2} requests/second. Each point in Figures 2-4 is obtained by taking the average of 10 runs of the algorithms. In each run, total number of entry to the CS is $5 * n$, where n is number of nodes in the network. In other words, each point in Figures 2-4 corresponds to the average cost of $50 * n$ entry to the CS.

Figure 2 plots the cost of the algorithms against λ, in a single-hop network. The network is composed of $n = 20$ nodes placed uniformly at random in an area of $100m \times 100m$. Carrier sense range is $250m$. In such a scenario, each node is an immediate neighbor of every other node. We observe that in Figure 2(a), *TOA* outperforms Raymond's algorithm, when λ is small (i.e., under light demand for the token). In single-hop networks and for small λ, approximately 4 messages are transmitted per CS entry in Raymond's algorithm while 2 messages per CS entry are transmitted in *TOA* (as explained in Section 3.1). Figure 2(b) shows that the delay per CS entry is smaller in *TOA* than in Raymond's algorithm, under light demand for the token. Under light demand for the token, when node i makes a request to enter the CS, no other message is transmitted in the network except the messages correspond to the request of node i. In this case, the delay per CS entry is equal to the time required to transmit request and token messages between the requesting node and the token holder, and the wireless channel is available whenever a node wants to transmit a message; i.e. there is no contention in the network.

Raymond's algorithm is designed such that, under heavy demand for the token, constant number of messages (approximately 3) are transmitted per CS entry. Detailed explanation can be found in [1]. In Figure 2(a) we observe what we expected, meaning that under heavy demand approximately 3 messages are transmitted in both Raymond's algorithm and *TOA*. As Figure 2(b) shows, both *TOA* and Raymond's algorithm has the same delay when λ is large, simply because both algorithms transmit the same number of messages per CS entry. Delay increases as λ increases, because at each time instant, there is more than one node requiring access to the channel and so packet of a node might be delayed by other nodes that are using the channel.



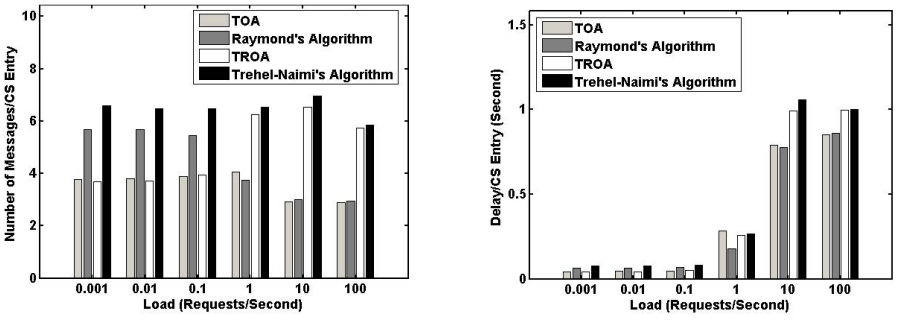
(a) Number of Messages (b) Delay

Fig. 2. 20 nodes placed in 100mx100m

As Figure 2 shows the cost of *TOA* is half of the cost of Raymond's algorithm under light demand. Under heavy demand both algorithms perform approximately the same. We conclude that the cost is decreased by opportunistic overhearing when demand for the token is light.

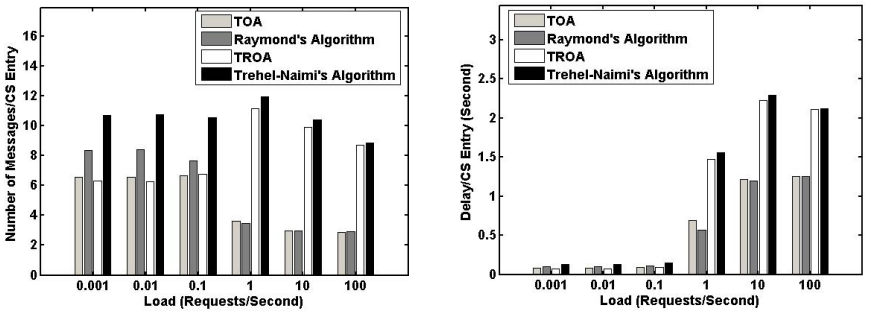
Figure 2 also plots the cost of *TROA* and Trehel-Naimi's Algorithm. As Figure 2(a) shows, in *TROA* two messages are transmitted per CS entry under light demand. The reason is that in *TROA*, nodes send their request to the last token holder, which is known to them because of message overhearing. So, only two messages, one request message and one token message, are transmitted per every CS entry. On the other hand, in Trehel-Naimi's algorithm, a requesting node does not necessarily know which node is the last token holder, since a node does not receive messages exchanged between other nodes. In such a case, a sequence of request messages are transmitted until the request of the requesting node is received by the token holder. We conclude that under light demand, cost of *TROA* is less than the cost of Trehel-Naimi's Algorithm.

Figure 2(a) shows that when demand for the token increases, the number of messages transmitted in *TROA* increases. The reason is that requests for the token from different nodes are initiated close to each other, and so a node might not know the latest status of the algorithm when it initiates a request. For example, we consider a case where node i sends a request to the token holder, node j. If another node k initiates a request before it overhears the request of node i, node k sends its request to node j which is not the last requesting node any more. Node j will forward the request of node k to node i and so one extra request message is transmitted. Figure 2(b) shows that delay per CS entry in *TROA* is always less than Trehel-Naimi's algorithm, when λ is small, simply because fewer messages are transmitted in *TROA*. When λ increases, delay of both *TROA* and Trehel-Naimi's algorithm increases, as a result of contention between nodes on accessing the wireless channel. Considering Figure 2, we conclude that in single hop networks, opportunistic overhearing improves the performance the most when demand for the token is light, and the improvement is about 100%.



(a) Number of Messages (b) Delay

Fig. 3. 40 nodes placed in 500mx500m



(a) Number of Messages (b) Delay

Fig. 4. 60 nodes placed in 800mx800m

Figures 3 and 4 plot the cost of the algorithms in multi-hop networks. Figure 3 plots the cost in a network of 40 nodes placed randomly in an area of $500m \times 500m$. The underlying network in Figure 4 is 60 nodes placed randomly in an area of $800m \times 800m$. Figure 3 shows that in this network topology, under light demand for the token (small λ), the cost of *TOA* is less than the cost of Raymond's algorithm. Comparing *TOA* and Raymond's algorithm in Figure 2 and Figure 3, we observe that the improvement obtained by message overhearing has decreased in Figure 3. The reason is that in a multi-hop network, nodes do not overhear all transmitted messages in the network and so they are not able to learn the latest status of the algorithm, i.e. they might not know which node currently holds the token. As Figure 3 shows and as we explained before, under heavy demand (large λ), Raymond's algorithm and *TOA* has almost the same cost. Figure 3(b) shows that the delay of Raymond's algorithm and *TOA* increases when λ increases. This is because of the contention between nodes in accessing the wireless channel.

As we observe in Figure 4, the cost of *TOA* is still less than the cost of Raymond's algorithms, although improvement percentage has decreased. This

shows that in Raymond's algorithm, as the size of the network increases, the improvement percentage obtained by exploiting message overhearing decreases. As Figures 3 and 4 show, in these networks when λ is small, *TROA* outperforms Trehel-Naimi's algorithm and the improvement is still significant. We conclude that the effect of exploiting message overhearing in different MUTEX algorithms is not always the same; instead it highly depends on the design of the algorithm.

6 Conclusion

We design two distributed token based mutual exclusion algorithms for wireless networks, called *TOA* and *TROA*. Our algorithms exploit the shared nature of the wireless channel in which nodes can overhear the messages not intended for them. We measured the performance of our algorithms as well as Raymond's algorithm and Trehel-Naimi's algorithm through simulations in ns-2, in networks of different sizes and under various rates of the demand for the token. We discussed under what conditions the performance of the considered MUTEX algorithms is improved by exploiting message overhearing.

Acknowledgments. This work was supported in part by Boeing.

References

1. Raymond, K.: A tree-based algorithm for distributed mutual exclusion. ACM Trans. Comput. Syst. 7, 61–77 (1989)
2. Naimi, M., Trehel, M.: A distributed algorithm for mutual exclusion based on data structures and fault tolerance. In: Conference Proceedings of Sixth Annual International Phoenix Conference on Computers and Communications, pp. 33–39 (1987)
3. Walter, J., Welch, J., Vaidya, H.: A mutual exclusion algorithm for ad hoc mobile networks. Wireless Networks 7, 585–600 (2001)
4. Manivannan, D., Singhal, M.: An efficient fault-tolerant mutual exclusion algorithm for distributed systems. In: Proceedings of the International Conference on Parallel and Distributed Computing Systems, pp. 525–530 (1994)
5. Agrawal, D., Abbadi, A.: An efficient and fault-tolerant solution for distributed mutual exclusion. ACM Trans. Comput. Syst. 9, 1–20 (1991)
6. Singhal, M.: A heuristically-aided algorithm for mutual exclusion in distributed systems. IEEE Transactions on Computers 38, 651–662 (1989)
7. Wu, W., Cao, J., Raynal, M.: A Dual-Token-Based Fault Tolerant Mutual Exclusion Algorithm for MANETs. In: Zhang, H., Olariu, S., Cao, J., Johnson, D.B. (eds.) MSN 2007. LNCS, vol. 4864, pp. 572–583. Springer, Heidelberg (2007)
8. Bulgannawar, S., Vaidya, N.: A distributed k-mutual exclusion algorithm. In: ICDCS, pp. 153–160 (1995)

MANET Location Prediction Using Machine Learning Algorithms

Fraser Cadger, Kevin Curran, Jose Santos, and Sandra Moffett

Intelligent Systems Research Centre, School of Computing and Intelligent Systems
Faculty of Computing and Engineering, University of Ulster, Northern Ireland
Cadger-f@email.ulster.ac.uk

Abstract. In mobile ad-hoc networks where users are potentially highly mobile, knowledge of future location and movement can be of great value to routing protocols. To date, most work regarding location prediction has been focused on infrastructure networks and consists of performing classification on a discrete range of cells or access points. Such techniques are unsuitable for infrastructure-free MANETs and although classification algorithms can be used for specific, known areas they are not general or flexible enough for all real-world environments. Unlike previous work, this paper focuses on regression-based machine learning algorithms that are able to predict coordinates as continuous variables. Three popular machine learning techniques have been implemented in MATLAB and tested using data obtained from a variety of mobile simulations in the ns-2 simulator. This paper presents the results of these experiments with the aim of guiding and encouraging development of location-predictive MANET applications.

Keywords: location-prediction, MANET, geographic routing, machine learning, decision tree, neural network, support vector regression.

1 Introduction

Mobile ad-hoc networks (MANETs) are a subcategory of ad-hoc networks; dynamic networks composed of wireless-enabled devices that form a network amongst themselves without the need for any existing infrastructure. MANETs are essentially ad-hoc networks where device mobility is permissible. MANETs are often used in scenarios where a specific task needs to be accomplished or for resource sharing. MANETs are inherently distributed in nature and are therefore an ideal means of allowing multiple heterogeneous devices to pool resources. Due to their ability to form dynamic, temporary networks without the need for direct user involvement and initiation, MANETs offer great potential for future Internet systems as a means of sharing and extending Internet access. Wireless Mesh Networks (WMNs) are another subcategory of ad-hoc networking in which infrastructure nodes form a mesh between themselves to share network access amongst ordinary nodes. However, although Internet access through public WiFi and cellular networks is becoming more commonplace, depending on the area and network Internet access may not be available to all nodes. Therefore the use of ad-hoc networking allows nodes with Internet access to

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 174–185, 2012.

share their coverage with other nodes. Thus extending Internet access to nodes who are incapable of accessing it directly but are able to communicate with other mobile devices.

An interesting area of research from a MANET point of view is geographic routing in which routing decisions are made based on the physical locations of devices (typically by selecting the node closest to the destination) instead of logical addressing systems such as IP. Geographic routing protocols typically only contain information about one-hop neighbors instead of memorizing an entire network topology. This makes geographic routing more resistant to the frequent topology changes that can be expected in mobile networks. Within the field of geographic routing, attempts have been made to apply location prediction algorithms for purposes such as Quality of Service (QoS) management [1-2] and mobility management [3]. In [1-2] basic mobility predictions are used to determine a neighboring node's suitability for providing a desired level of QoS In [1] a metric known as link duration is used to determine how long a link between two nodes will last based on their mobility, while [2] uses a similar approach but instead focuses on determining whether a neighbor is likely to incur a high level of delay and/or jitter based on their mobility pattern. Similarly, mobility management schemes such as [3] can use mobility prediction to mitigate the potentially damaging effects of mobility (i.e. trying to route to a neighbor who is no longer available) through a variety of countermeasures such as adaptive beaconing in which nodes do not send messages periodically but instead send them when their motion changes in a manner likely to affect their neighbors.

As geographic routing is reliant on device location data, it is not surprising that there has been some research into the use of location prediction algorithms that allow devices to predict the future locations of other devices. In addition to the QoS and mobility management examples above, basic geographic routing can be enhanced by allowing nodes to base geographic routing decisions on where neighboring nodes are expected to be at the time of routing rather than where they were according to the last update (which may be some time ago). However, the majority of the location prediction algorithms have been relatively simplistic. In [4] it was shown that geographic routing protocols using location prediction outperformed a geographic routing protocol without location prediction and it was speculated that more accurate location prediction algorithms would lead to further performance increases. Similarly, location prediction technologies can also be of benefit to opportunistic networking protocols by helping make decisions about whether to hold or forward data based on mobility. As well as improving the performance of routing protocols, location prediction performed by MANETs is also important for developing protocols that are explicitly aware of mobility and other elements of their environment. Such protocols are not only beneficial to MANET routing in general but also the area of context-awareness which is part of the larger area of pervasive computing.

As WiFi-equipped mobile devices become more common it is likely that new and novel applications of ad-hoc networking will emerge and that users will increasingly expect connectivity on the move. Sharing of resources such as Internet connection is just one of these examples and in order to become a commonplace reality devices and protocols must be able to handle mobility rather than expect users to remain still. Although the majority of location prediction algorithms found in geographic routing protocols (and other location-aware MANET protocols) have been fairly simplistic,

there have been some attempts at using more advanced location prediction methods in Wireless LANs (WLAN) and cellular networks, which unlike MANETs have dedicated infrastructure to rely on. Techniques from the field of machine learning such as Artificial Neural Networks [5], Hidden Markov Models [6], Bayesian Networks [7], and Support Vector Regression [8] have all been used for predicting future locations. Although these machine learning techniques often result in high-levels of prediction accuracy it is important to recognize that most of them rely heavily on the infrastructure provided by WLANs and cellular networks, which is not always available to MANETs. Most of these formulate the problem of predicting user/device locations in terms of predicting access points or cells rather than fixed geographic coordinates. This allows for the range of 'locations' to be expressed as a known and discrete series of points, which allows the problem to be handled as a classification problem. Although [8] uses a regression algorithm, it is used in the context of predicting GSM infrastructure, which is unsuitable for use in infrastructureless MANETs.

These approaches are therefore difficult to apply to MANET systems that do not have access points or other infrastructure which could be used as discrete locations. If the problem of MANET location prediction is seen as a regression one, then a system could be developed that takes existing location information and outputs predicted future location in the form of continuous coordinates. Such a system would be suitable to use with any positioning or navigation system that used numerical coordinates and thus would be significantly more flexible than a classification based system using relative locations.

The work of [9] is the only work that has applied machine learning to predicting user locations in MANETs. [9] formulates the problem of location prediction as a time series prediction problem to allow for the prediction of node trajectories. The drawback of such an approach is that it relies on the availability of a suitable time series, which depending on how node position data is obtained is not always an option. To date, there have been no attempts to develop a system for predicting continuous coordinates using machine learning techniques. This paper therefore aims to establish whether regression-based machine learning techniques are suitable for the task of predicting mobile device locations in MANETs. To do so, three machine learning algorithms (Decision Trees, Artificial Neural Networks, and Support Vector Regression) have been tested in Matlab using data obtained from ns-2 simulations of two different mobility models and three different network sizes giving a total of six different scenarios. It is intended that the results of these experiments can be used to determine the viability of machine learning in general and the chosen algorithms in particular for predicting device locations in MANETs, and in doing so help motivate and guide the development of protocols that make use of machine learning.

2 Machine Learning Algorithms

2.1 Decision Trees

A decision tree (DT) is a graphical model that displays decisions in the form of a tree containing nodes that correspond to further nodes or decisions (the predicted values). Nodes are often presented in the form of a series of questions and the answer to that question decides whether a particular decision/value is reached, or another node is

traversed. Classification and Regression Tree (CART) are a popular technique for constructing DTs that was introduced by [10]. In CART DT construction, input data is split into smaller categories using splitting rules for classification problems or the squared residuals minimization algorithm for regression trees. The splitting continues until the maximum tree – one in which nodes that do not point to other nodes only point to one class or variable - has been constructed.

CARTs have several advantages, one of the most prominent of these is their ability to be easily understood and interpreted by humans due to their visual nature. Other advantages of CARTs for DTs include their nonparametric nature and speed of computations [11]. A significant disadvantage to DT construction using CARTs include the possible instability of produced DTs where small changes in data values or structure can lead to significant changes in the resulting DT. Another issue with CART DTs include the possibility of unnecessarily large regression trees being created – although the effects of this can be mitigated by using pruning algorithms [11].

2.2 Neural Networks

Artificial Neural Networks which shall hereafter be referred to as neural networks or (NNs) are mathematical models based on real-life biological neural networks that present an adaptive solution to the solving of statistical problems such as classification and regression. Like biological neural networks, NNs are composed of several units known as neurons which accept inputs and perform some form of calculation to produce an output value which is then passed to another layer. The calculation performed by a neuron is determined by the input and the weight of the link between the current neuron and the previous neuron. Connections between neurons go between layers of neurons, with each layer containing several neurons. Interconnections between neurons contain weights and biases, and input to a node is multiplied by both of these to produce the input, which is then multiplied by a transfer function (such as linear or log-sigmoid) to produce the output. The flow of inputs between neurons can either be one-way (as in feedforward networks) or cyclical (recurrent networks).

The learning stage of neural network training is done through altering the weights of the interconnections between neurons. In supervised NNs this is done so as to minimize the error between the output produced by the network and the target output provided by the user. As with the CART DTs an advantage of NNs is that they are easy to use and no considerable knowledge of their operation is required to use basic NNs. Unlike DTs, NNs do not present a structure through which processing can be easily followed and understood by humans. This had led to NNs being termed ‘black box’ systems where data is entered, modified and then output with no way for the user to determine exactly what happened to it. Other advantages of NNs include their ability to detect the existence of complex nonlinear relationships between inputs and targets, their ability to detect all interactions between prediction variables, and their flexibility in the form of a wide range of training algorithms that can produce radically different results with the same architecture [12]. While disadvantages include requiring potentially large amount of computational resources and the possibility of overfitting [12].

2.3 Support Vector Regression

Support Vector Regression (SVR) is the extension of the Support Vector Machine (SVM) model to allow for the prediction of continuous variables instead of discrete classification. Standard SVMs are statistical models for classifying input data into one of two classes using a kernel function which is learned from training data using techniques based on the concept of statistical learning developed by Vladimir Vapnik and implemented in the form of the SVM by Cortes and Vapnik [13]. SVR was proposed by [14] as a means of extending SVM to solve regression problems in which the loss function used to design the SVM differs by ignoring errors that fall below a certain threshold in order to create a function that has an error below the specified level [15]. The main strengths of the SVM approach include good generalization ability (particularly useful for small datasets) and good performance with high-dimension data [16]. The biggest disadvantage is the potentially large size of the completed kernel and the amount of time taken to create it [17].

3 Experimental Configuration

3.1 Obtaining the Data

All of the data for the training and testing sets was obtained from runs of the ns-2 simulator This was done by simulating the GPSR protocol [18] with print statements that printed node location information from routine beacon messages (sent every 0.5 seconds). In ns-2 nodes have the ability to discover their own physical location, although this is often referred to as GPS the coordinates are in Cartesian form. However, these experiments exist to determine the suitability of machine learning algorithms to predict continuous numerical coordinates and although Cartesian coordinates are used, the approach could be extended to another format of coordinates such as GPS or Galileo. Two different mobility models the Random Waypoint Mobility model (RWM) [19] and the Reference Point Group Mobility model (RPGM) [20] were used to provide node movement information. RWM is a relatively simplistic model in which nodes are provided with destinations, and upon reaching those destinations remains at the destination for a set period of time (the pause time) before going to another destination, all movement is individual [19]. Despite its simplicity and claims of inaccuracy, RWM is one of the most frequently used models of mobility in literature. The RPGM focuses on group mobility although individual mobility is still permissible as not all nodes will join groups while others will move individually at some stages and as part of a group at others [20]. The decision to use these two models was based on the desire to experiment with both purely random and also group mobility, to determine how well the prediction algorithm could handle such behaviour.

Mobility in ns-2 is simulated through the use of mobility trace files which contain a set of coordinates that nodes will travel to in the course of the simulation, and the RWM mobility traces were obtained using the ns-2's setdest tool while the RPGM movements were obtained using the BonnMotion tool [21]. For each of the two mobility models scenarios of 10, 50, and 100 nodes were created thus giving a total of

six unique scenarios. All simulations took place on a 1500 by 300 grid, ran for 300 seconds and used pause times of 20s and maximum velocities of 2.5 m/s to try and accurately capture movement by humans on foot. The data obtained from these simulations was in the following format for inputs, while the target data simply comprises the respective x and y coordinates:

Current time - latest x – latest y – latest update time – previous x – previous y – previous update time

3.2 Matlab Configuration

As a stated aim of this work was to use off-the-shelf algorithms in order to allow for general comparison of algorithms, where possible, minimal configuration was done and defaults were preferred. It should be noted that while Matlab implementations of NN and DT are available using the Neural Network and Statistics toolboxes there is no default support for SVR (although SVM classification is available in the Bioninformatics toolbox) and as such a third-party SVM toolbox for Matlab – libsvm developed by [22] was used. It should also be noted that while multivariate output is available as standard for NNs, it is not available in either the official or third party toolboxes and as a compromise separate x and y target vectors were created from the original datasets and ran one after another (i.e. x predictions first, y predictions second).

Although it is acknowledged that doing so would lead to a loss of accuracy as the relationship between the x and y coordinates is lost if they are considered separately, it was the only realistic way of using these algorithms. Similarly, it is important to recognise that even if the implementations of the DT and SVR algorithms used here use only single-variable output this is because of the limitations of the Matlab toolkits, and if desired multivariate versions of the algorithms could be implemented in future work. In recognition of the possible loss of accuracy this could cause the mean square error (MSE) for the combined x and y predictions is calculated as the average, while the duration is calculated as the sum of the two x and y predictions. It should also be noted, that using Matlab defaults the dataset of inputs and targets is split into 70% training, 15% validation, and 15% testing. As there is no such default with DT and SVR the decision to manually split the data into 85% training and 15% testing was taken.

For DTs, the t=classregtree(…) command creates a regression tree using the appropriate input and target training data. The predict(…) command is used to create a vector called predictedLocations that will store the results of the predictions created with the t(…) command. Matlab's wizard-like tools (nnstart and nftool) were used to create a 2-layer feedforward network, with 10 (sigmoid) hidden neurons and 2 linear output neurons. Training is performed using the Levenberg-Marquadt algorithm as the training algorithm and MSE as the measure of performance. For SVR, the first command is model = svmtrain(…) command takes a set of inputs and targets as well as libsvm parameters and creates a model containing parameters, support vectors, etc. for making future predictions. For SVR the main parameter is –s which determines the type of SVM – either Epsilon or Nu, after running both Nu was selected due to its

higher performance. The actual are predictions are performed using the svmpredict(…) command which takes in target and input data and produces vectors of predicted labels (outputs), accuracy, and decision_values.

4 Results

When examining these results it is important to recognize that all simulations were performed on a desktop PC that has significantly more memory and processing power than a typical mobile device found in a MANET would have. Therefore these results should not necessarily be seen as the results that would be obtained for an implementation on actual mobile devices. Similarly, different configurations of the algorithms (i.e. a greater or lesser number of neurons in the NN) would yield different results, which is why standard or 'off-the-shelf' configurations of each algorithm have been used. However, the main purpose of these results is to determine which of the three chosen algorithms achieves the best *general* performance and then focus future development on this algorithm. Therefore these results should serve largely as a comparison of the algorithms against each other and it should not be expected that the exact same results will be obtained from mobile device deployment.

For this reason both accuracy (MSE) and performance (duration) metrics have been chosen for evaluation. MSE has been chosen because unlike other measures of performance such as correct prediction % MSE is an absolute measure of performance and is able to take into account how close a predicted value is to the target rather than simply stating whether it is correct or incorrect. Similarly, duration in seconds is also important as it gives a measure of how long training and prediction with each algorithm will take. Although the results obtained using mobile devices will be different, the duration metric can still serve as an indicator of which algorithms have the lowest or highest overheads.

4.1 Random Waypoint Model

Tables 1 shows the results from the tests of the three location prediction algorithms against scenarios of varying node numbers.

Table 1. Mean Square Error for the prediction algorithms in each RWM scenario

Prediction Algorithm	**10 Nodes**	**50 Nodes**	**100 Nodes**
Decision Tree	182000	1.9	1.31
Neural Network	73	3.31	0.172
Support Vector Regression	3480	219000	25300

From these results it can be observed that with the exception of SVR all of the prediction algorithms' performances improve as the size of the training and testing sets increases. In all but one of these scenarios (50 nodes) NN is the best performer and

this scenario is the same scenario that DT records the best MSE. Although NN is beaten by DT in the 50 node scenario it can be seen as the best overall performer, as it first in two scenarios and second in one, whereas DT is first in one scenario, second in another (100 nodes) and last for 10 nodes. SVR finishes significantly behind the two other algorithms in all scenarios except the first where it comes second to NN. Perhaps the most surprising feature of these results is SVR's performance decreasing as the training set increases. It is generally expected that the larger the training set the more accurate the resulting predictions will be, however the results from SVR show that when a larger training set is used its performance decreases.

While these results would seem to indicate that NN is (at least for mobility based on RWM) the best overall performer, with DT coming second overall it is also important to consider cost, in terms of time taken to complete training and testing. Table 2 shows the total durations for each algorithm in all three RWM scenarios.

Table 2. Duration in seconds for the prediction algorithms in each RWM scenario

Prediction Algorithm	**10 Nodes**	**50 Nodes**	**100 Nodes**
Decision Tree	1.2105	3.8	9.912
Neural Network	7.394	23	7706
Support Vector Regression	1.6756	109.2	452.98

One of the most interesting features of these results is the amount of time taken for NN to complete training and testing for the 100 node scenario. While all of the algorithms exhibited an increase in prediction duration as the number of nodes (and therefore size of training set) was increased, the prediction duration for the 100 node scenario is notable not just because of its size but because of the difference between it and the previous (50 node) scenario. This would seem to suggest that NNs (or at least the implementation of the NN used here) are possibly unsuitable for large datasets. However, it is possible that this is only due to the large training set and that as training is a one-off cost the NN algorithm would not necessarily take such a long time to perform future predictions.

4.2 Reference Point Group Mobility

Tables 3 shows the results of the three machine learning algorithms applied to the three RPGM scenarios.

Table 3. Mean Square Error for the prediction algorithms in each RPGM scenario

Prediction Algorithm	**10 Nodes**	**50 Nodes**	**100 Nodes**
Decision Tree	444.17525	0.108	0.5309
Neural Network	0.357	0.539	0.5049
Support Vector Regression	129000	197000	251000

The results from the RPGM scenarios show a similar pattern to those of the RWM scenarios with NN being the best performer in 2 out 3 scenarios (10 nodes and 100 nodes) and DT coming first in one scenario (the 50 node scenario) which coincidentally contains the same number of nodes as the RWM scenario in which it finished first. However, unlike the results from the RWM simulations SVR finishes last in every scenario, with DT finishing second in the two scenarios where it didn't finish first. Another interesting difference is that unlike the previous results, NN actually experiences a slight dip in performance between the 50 node and 100 node scenarios whereas in the RWM scenarios it always showed an increase in performance between scenarios. Although the dip is only slight (around 0.03) it is still an anomalous result, given the logic that bigger training sets lead to more accurate results.

Comparing the results of the two mobility model simulations it can be seen that DT performs better in all RPGM scenarios than it does RWM scenarios, while NN performs better in 10 and 50 node scenarios for RPGM and 100 nodes for RWM. Like DT, SVM performs best for all RPGM scenarios. If RPGM is an accurate model of group mobility then it could also be considered to be more accurate overall than the RWM as the RPGM allows for single entity as well as group movement while RWM only allows for single entity mobility. Therefore, it is interesting to note that DT and SVM achieve their best results for each number of nodes when using the RPGM model. Similarly, although NN performs best with RWM for the 100 node scenarios, it performs best with RPGM with 10 and 50 nodes.

If the results from RPGM are indeed more accurate then these results are particularly encouraging as it shows that when presented with realistic mobility all three protocols are (generally) able to obtain better prediction accuracy than they are with the less realistic RWM model. This is encouraging as it suggests that these algorithms are suitable for use in real world MANET applications due to their improved performance in realistic vs. unrealistic mobility models. It is worth noting, however, that although SVR performs better with RPGM scenarios its overall performance is often considerably worse than that of NN and DT. Regarding NN and DT it appears that overall NN is the best performer as it comes first in 4 out of 6 scenarios and second in the other two, while DT comes first in 2 scenarios, second in 3 and third in 1. Although NN appears to the best overall performer in terms of accuracy, it is important to recognize that for the RWM scenarios it often performed significantly worse than DT in terms of duration. Another important factor, from a MAENT perspective is duration and Table 4 shows the total durations for each of the protocols in all three MANET scenarios.

Table 4. Duration in seconds for the prediction algorithms in each RPGM scenario

Prediction Algorithm	**10 Nodes**	**50 Nodes**	**100 Nodes**
Decision Tree	1.04	2.85	8.937
Neural Network	2	38	410
Support Vector Regression	2.62	114.67	461.44

It is clear that the best performer for all scenarios is DT which never experiences a duration of over 10 seconds – even with the 100 node dataset. NN outperforms SVR in all RPGM scenarios while in the RWM scenarios it was beaten by SVR in the 50 and 100 node scenarios. Taking the MSE and duration metrics for both mobility models into account it would seem that although NN achieves the best overall level of accuracy, DT clearly completes training and testing in the lowest amount of time and often has an MSE level close to that of NN. This would seem to suggest that DT is the method best suited for use on mobile devices with constrained resources, as DT consistently has the lowest duration but generally still obtains a high level of accuracy in its predictions.

5 Conclusion

This paper addresses the issue of predicting future mobile device locations in MANETs. As it is possible that MANETs will play an important role in the future Internet, there are several reasons for doing so, such as mobility management and location-based networking (i.e. geographic routing). The Introduction briefly discussed previous applications of location prediction to geographic routing and MANETs in general, such as mobility management based on movement or predicting QoS suitability based on mobility pattern. Similarly, opportunistic routing protocols could benefit from location prediction as a means of assessing whether to hold or forward a packet. Using location prediction a device could determine whether either itself or one of its neighbors was moving towards the destination and then either hold the packet or forward the packet to a neighbor and tell it to hold onto the packet. Location prediction could also be applied to energy-aware protocols to determine trade-offs between energy consumption and routing effects. It is hoped that in light of the largely positive results of these experiments that more time and effort will be devoted to research into using location prediction in both geographic and MANET routing in general. However, another more general goal of this work is to increase device/protocol awareness about their environment in order to not only obtain better performance but greater context-awareness in order to develop an infrastructure capable of supporting new applications and uses of mobile technologies.

In this paper three machine learning algorithms (Decision Trees, Neural Networks, and Support Vector Regression) were applied to the purpose of predicting future location from data obtained from ns-2 simulations of 6 different mobility scenarios to determine their suitability for use in future location-aware MANET protocols. The main metrics for comparison were accuracy (MSE) and duration (seconds). In terms of accuracy, the best performer was the NN algorithm followed by DT with SVR finishing last in all but 1 scenario. These results can be contrasted with the results for the duration metric in which DT was the best performed, followed by NN and then SVR. Therefore, although the NN algorithm may seem like the best option given its generally high performance in terms of accuracy, its weaker performance (in contrast to DT) in terms of duration raises questions over its suitability for use in resource-constrained mobile devices. However, it is possible that an alternative configuration that places less demand on resources could be used for such instances. Therefore al-

though DT is the best performer in terms of duration, and overall second to NN in accuracy, the strong prediction accuracy of NN means that suitability is largely guided by application. So that for instances where speed or efficiency are required DT may be more suitable while NN is best suited for scenarios where high accuracy but potentially slower performance is desired.

Regardless of whether the DT or NN algorithm has performed best, it should be clear from this paper that both algorithms are capable of achieving high levels of performance when using MANET datasets and this would suggest that the application of machine learning techniques to MANET location prediction is worthwhile. It is hoped that the results presented in this paper can be of use in the design and implementation of location-predictive MANET protocols, for the dual aims of increasing MANET performance and context-awareness.

Acknowledgements. Fraser Cadger is sponsored by a DEL PhD Studentship from the University of Ulster and Northern Ireland Executive.

References

1. Stojmenovic, Russell, M., Vukojevic, B.: Depth first search and location based localized routing and QoS routing in wireless networks. In: Proceedings of 2000 International Conference on Parallel Processing (2000)
2. Shah, S.H., Nahrstedt, K.: Predictive location-based QoS routing in mobile ad hoc networks. In: IEEE International Conference on Communications, ICC 2002 (2002)
3. Chen, Q., Kanhere, S., Hassan, M., Lan, K.-C.: Adaptive Position Update in Geographic Routing. In: 2006 IEEE International Conference on Communications, pp. 4046–4051 (June 2006)
4. Cadger, F., Curran, K., Santos, J., Moffett, S.: An Analysis of the Effects of Intelligent Location Prediction Algorithms on Greedy Geographic Routing in Mobile Ad-Hoc Networks. In: Proceedings of the 22nd Irish Conference on Artificial Intelligence and Cognitive Science (2011)
5. Capka, J., Boutaba, R.: Mobility Prediction in Wireless Networks Using Neural Networks. In: Vicente, J.B., Hutchison, D. (eds.) MMNS 2004. LNCS, vol. 3271, pp. 320–333. Springer, Heidelberg (2004)
6. Prasad, P.S., Agrawal, P.: Movement Prediction in Wireless Networks Using Mobility Traces. In: 2010 7th IEEE Consumer Communications and Networking Conference (CCNC), pp. 1–5 (2010)
7. Zhang, Y., Hu, J., Dong, J., Yuan, Y., Zhou, J., Shi, J.: Location prediction model based on Bayesian network theory. In: Proceedings of the 28th IEEE Conference on Global Telecommunications, pp. 1049–1054. IEEE Press, Piscataway (2009)
8. Wu, Z.-L., Li, C.-H., Ng, J., Leung, K.: Location Estimation via Support Vector Regression. IEEE Transactions on Mobile Computing 6, 311–321 (2007)
9. Kaaniche, H., Kamoun, F.: Mobility Prediction in Wireless Ad Hoc Networks using Neural Networks. Journal of Telecommunications 2, 95–101 (2010)
10. Breiman, L., Friedman, J., Stone, C.J., Olsen, R.A.: Classification and Regression Trees. Chapman and Hall (1983)
11. Timofeev, R.: Classification and Regression Trees (CART) Theory and Applications. Humboldt University, Berlin (2004)

12. Tu, J.V.: Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes. Journal of Clinical Epidemiology 49, 1225–1231 (1996)
13. Cortes, C., Vapnik, V.: Support-Vector Networks. Machine Learning, 273–297 (1995)
14. Drucker, H., Burges, C.J.C., Kaufman, L., Smola, A., Vapnik, V.: Support Vector Regression Machines (1996)
15. Smola, A.J., Schölkopf, B.: A tutorial on support vector regression. Statistics and Computing 14, 199–222 (2004)
16. Gunn, S.R.: Support Vector Machines for Classification and Regression (1998)
17. Burges, C.J.C.: A Tutorial on Support Vector Machines for Pattern Recognition. Data Mining and Knowledge Discovery 2, 121–167 (1998)
18. Karp, B., Kung, H.: GPSR: greedy perimeter stateless routing for wireless networks. In: Proceedings of the 6th Annual International Conference on Mobile Computing and Networking, pp. 243–254. ACM (2000)
19. Johnson, D.B., Maltz, D.A.: Dynamic source routing in ad hoc wireless networks. In: Mobile Computing, pp. 153–181. Kluwer Academic Publishers (1996)
20. Hong, X., Gerla, M., Pei, G., Chiang, C.-C.: A group mobility model for ad hoc wireless networks. In: Proceedings of the 2nd ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems, pp. 53–60. ACM, New York (1999)
21. Aschenbruck, N., Ernst, R., Gerhards-Padilla, E., Schwamborn, M.: BonnMotion: a mobility scenario generation and analysis tool. In: Proceedings of the 3rd International ICST Conference on Simulation Tools and Techniques, ICST, Brussels, Belgium, pp. 51:1–51:10 (2010)
22. Chang, C.-C., Lin, C.-J.: LIBSVM: a library for support vector machines. Science 2, 1–3 (2001)

Cooperative Sensing-Before-Transmit in Ad-Hoc Multi-hop Cognitive Radio Scenarios

José Marinho[1,2] and Edmundo Monteiro[3]

[1] ISEC, Polytechnic Institute of Coimbra, Portugal
[2] CISUC, University of Coimbra, Portugal
fafe@isec.pt
[3] DEI/CISUC, University of Coimbra, Portugal
edmundo@dei.uc.pt

Abstract. Currently, the radio spectrum is statically allocated and divided between licensed and unlicensed frequencies. This results in its inefficient usage. Cognitive Radio (CR) is a recent paradigm which aims at addressing this inefficiency. It allows unlicensed users to opportunistically access vacant licensed frequency bands, as long as any harmful interference to incumbent users is avoided. In this context, providing appropriate medium access control (MAC) protocols is a core issue. However, the existing proposals omit relevant issues, are too complex, or are based on unrealistic assumptions in practical terms. Therefore, in this paper, we propose and analyze, through simulation, what we intend to be a really practical CR MAC protocol for distributed (i.e., ad-hoc) multi-hop CR scenarios, with high variability in terms of the availability of vacant frequency bands in time and space. Evaluation results confirm that the proposed approach, designated as CoSBT-MAC, can result in relevant benefits.

Keywords: Cognitive Radio, Medium Access Control, Cooperative Sensing, Multi-hop Ad-hoc Networks.

1 Introduction

Currently, it is well-assumed that the radio spectrum is inefficiently used and that a shift towards more flexible regulation policies must be accomplished. Licensed frequency bands are for the exclusive use of designated users, i.e., licensed users (e.g., UHF/VHF TV frequency bands), and unlicensed frequency bands can be freely accessed by any user, following certain rules (e.g., not exceeding a defined limit for transmission power). The latter includes, among other bands, ISM (Industrial, Scientific and Medical) which is shared by a large number of technologies such as high speed wireless local area networks and cordless phones. However, while the unlicensed spectrum bands are becoming more crowded, especially ISM in densely populated areas, licensed frequency bands are often underutilized, creating temporally available spectrum opportunities that are variable in time and space [1].

In this context, Cognitive Radio (CR) has emerged as one of the keys for exploiting the mentioned spectrum opportunities, often designated as spectrum holes or

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 186–197, 2012.

white spaces, opening it to secondary users (i.e., unlicensed users). According to the designated CR overlay approach, secondary users (SU) can opportunistically use these opportunities to increase their performance, but without causing any harmful interference to licensed users, also designated as primary users (PU). The operating spectrum band and other transmission parameters (e.g., transmission power) are dynamically and intelligently chosen by SUs based on spectrum availability. In fact, CR is highly interdisciplinary, being concerned with distinct engineering and computer science disciplines such as signal processing, communication protocols, and machine learning [8]. Therefore, CR issues may span all the layers of the communication protocol stack. However, its basics are mostly limited to the physical (PHY) and medium access control (MAC) layers.

In this paper, we propose and evaluate a novel CR MAC protocol, designated as CoSBT-MAC (Cooperative Sensing-Before-Transmit-based MAC), which targets multi-hop distributed (i.e., ad-hoc) CR networks, with high variability in terms of the availability of vacant frequency bands in time and space. The effective protection of primary users and the increase in performance which is delivered to SUs are the main objectives. Additionally, practicality, low complexity and high scalability are the main guidelines. In this proposal, SUs are considered to be totally autonomous, i.e., to make spectrum decisions based exclusively on observation, learning and cooperation. PUs are also assumed to be completely abstracted from CR-based accesses. The presented work specifically targets a layer-2/MAC protocol. Therefore, other relevant issues, such as learning schemes, remain out of its scope and will be considered in future work.

The remainder of the paper is structured as follows. Section 2 provides readers with enough background in order for them to fully understand this work. It also highlights the main contributions of CoSBT-MAC when compared to the existing ones. Section 3 describes the CoSBT-MAC proposal in detail. Section 4 evaluates it through simulation, in terms of the performance which is delivered to SUs and the achieved protection of PUs. Finally, section 5 draws brief conclusions.

2 Background and Related Work

The objective of this section is twofold: providing enough background about relevant CR issues in the context of our proposal; and identifying the limitations of previous works our CR MAC proposal intends to overcome.

2.1 Background

The architecture of CR networks can either be centralized or distributed, being spectrum decision made by a central entity and by the CR users, respectively. The distributed (i.e., ad-hoc) approach, where decision-making is processed by individual SUs, has the ability to reduce complexity, increase network scalability, and make the support of mobility easier.

Usually, it is considered that in CR networks the access to the spectrum is achieved through a "sense-before-transmit" approach, i.e., a channel is sensed before the transmission of any data packet and another one is searched if it is busy. This is mandatory to reduce the probability of harmful interference to PUs in the type of CR scenarios we target (see section 1). In distributed CR networks, it is up to the SUs to perform sensing locally. However, cooperative sensing schemes, in which SUs share and combine their sensing outputs, increase accuracy and efficiency [3][9]. To support the cooperation between SUs, most CR MAC protocols use a common control channel (CCC) which must be continuously available (i.e., deployed over frequency bands where PU activity is not a concern). Usually, a SU is equipped with an extra radio/transceiver which is dedicated for operations on the CCC.

In wireless networks, there is a so-called hidden node problem when two wireless nodes cannot sense each other, but have overlapped coverage areas. This means that a SU can interfere with a primary system even without being in its coverage area and, therefore, without being able to sense its activity (e.g., in Fig. 1, SU2 cannot sense the activity of PU1, but interferes with its coverage area). In most proposals, such as in the work of Timmers et al. [8], a SU cannot use a specific channel if it is sensed busy by any of the SUs in the network. This is the so-called OR-rule. However, this approach can result in an inefficiency designated as “false spectrum access denial”, i.e., a SU is denied the access to a given channel despite being out of the region of potential interference [10].

Concerning how spectrum access is performed, most current CR MAC proposals are based on random access protocols, i.e., CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) like access, for data and control traffic [4][11]. This approach is easier to implement and, consequently, was chosen for CoSBT-MAC, which aims to be simple and practical.

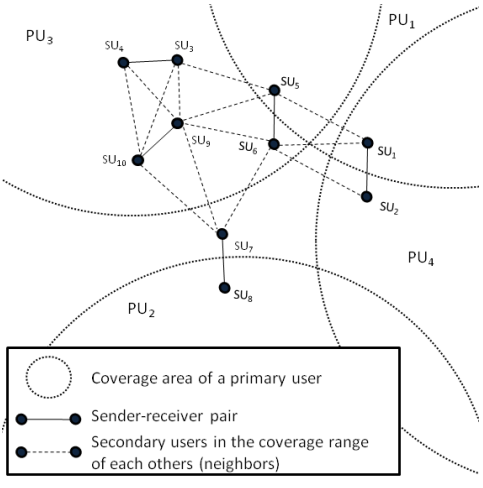


Fig. 1. A scenario with hidden nodes

2.2 Related Work

Several CR MAC proposals, based on random spectrum access, have already been proposed for distributed CR networks [11]. However, on the contrary of CoSBT-MAC, none of these works addresses all the following issues at once: cooperative sensing; balanced usage of opportunities; support of high variability in terms of spectrum opportunities; and protection of hidden PUs. For instance, the balanced usage of spectrum opportunities among SUs is only considered in the works of Yuan et al. [7] and Jia and Zhang [6]. The proposal of Jia and Zhang [6], which is based on the overhearing of control packets sent by neighbors, assumes that every SU has an up to date list of locally available channels. Therefore, the sensing issue is left open. In the work of Choi, Patel and Venkatesan [5], the data channel is unilaterally selected by the receiver and, therefore, it does not address the hidden PU problem. The work of Yuan et al. [7], which targets the usage of spectrum opportunities in the television frequency bands, does not propose any solution aiming at protecting hidden PUs (e.g., through a cooperative spectrum sensing approach). Finally, the work of Timmers et al. [8], which is very complete in terms of addressed issues, applies the OR-rule globally and assumes that sensing and communication can be performed simultaneously. Therefore and as discussed in section 3.3, it is more prone to the mentioned "false spectrum access denial" problem (see section 2.1) than our proposal. Additionally and on the contrary of CoSBT-MAC, the work of Timmers et al. [8] is not compatible with energy detection [3][10], which is the easiest sensing scheme to implement in practical terms (i.e., a channel is considered busy when the strength of the detected signal level is above a certain threshold).

Based on the background and on the description of related work which have just been provided, the next section describes the CoSBT-MAC proposal.

3 CoSBT-MAC Description

CoSBT-MAC strictly follows a "sense-before-transmit approach" (see section 2.1) and includes an innovative cooperative sensing scheme which aims at increasing the protection of PUs through the participation of idle neighbors. Spectrum access is random based and the usage of the spectrum opportunities is intended to be balanced. We also assume that SUs sense transmitting PUs through energy detection. In this proposal, every SU is equipped with two radios, one dedicated for operating the CCC and the other, which is dynamically reconfigurable, for data transmission. Finally, as we target multi-hop ad-hoc CR networks, our proposal also addresses the hidden primary and secondary user problems.

Fig. 2 illustrates the basics of the proposed CoSBT-MAC protocol. It includes two negotiation phases which are based on three-way handshake schemes: spectrum sensing; and spectrum decision. This is what we designate as a "cooperative sense-before-transmit" approach and results in a fully decentralized version of the cooperative sensing algorithm which is formally described in Fig. 3. In CoSBT-MAC, control packets are sent over the CCC, and data over the reconfigurable radio. RTS (Request To Send) packets, which are the first to be sent for each negotiation process

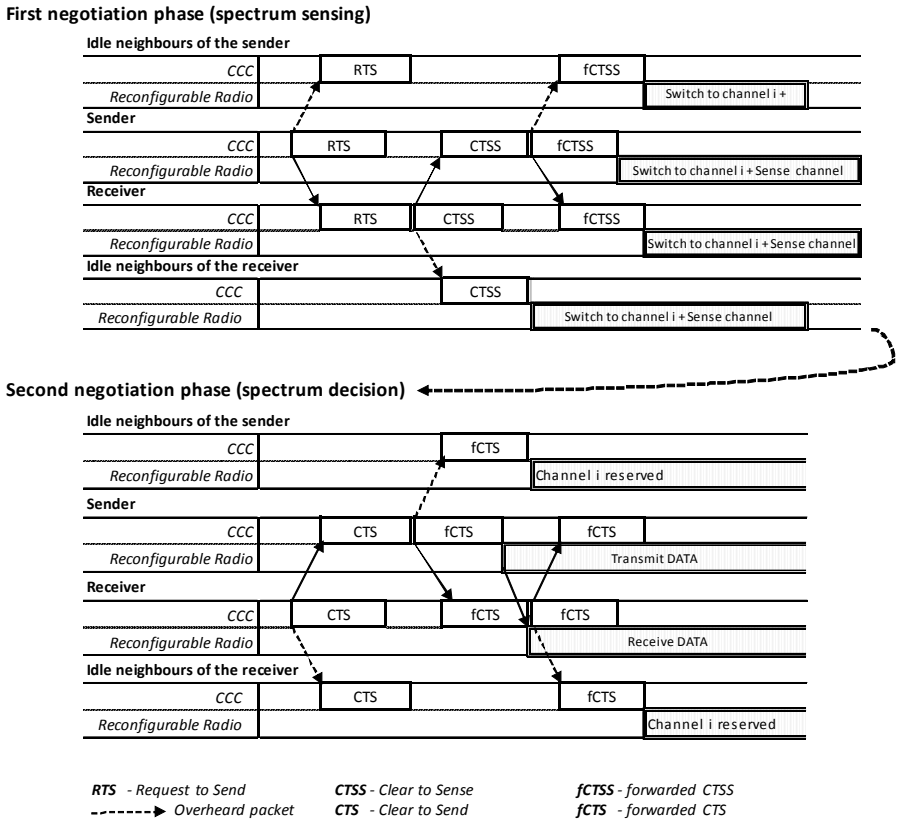


Fig. 2. Interaction between secondary users

(see Fig. 2), are transmitted through a contention scheme similar to IEEE 802.11 DCF (Distributed Coordination Function), which is based on CSMA/CA. However, because the CCC is set as reserved after the successful sending of an RTS packet, no contention is considered for the remainder control packets. In order to reduce the number of collisions on the CCC, random backoffs are also considered. On data channels, as reservation is previously accomplished through negotiation on the CCC (see Fig. 2), access is achieved without any contention.

After this generic introduction, a detailed description of the proposed CoSBT-MAC protocol is provided in the next subsections.

3.1 Sensing Phase

In the initial phase (i.e., the sensing phase), the sender first sends an RTS packet to the intended receiver. This packet includes, such as in the work of Jia and Zhang [6], a list of non-reserved channels as well as the number of data bytes to be transmitted. Every neighbor of the sender which overhears the RTS packet sets the CCC as being busy for a defined amount of time. This reservation scheme, which is based on overheard control packets, is a common practice in the field of wireless communications

Inputs: N, neighbors of the sender-receiver pair;
N_{idle}, idle neighbors of the sender-receiver pair;
C_s, the set of non-reserved channels at the sender;
C_r, the set of non-reserved channels at the receiver;
C_n, the set of non-reserved channels at neighbor n.

Output: A channel c which is free of any activity; or a selection failure.

1. **if** $(C_s \cap C_r) = \emptyset$ **then return** selection failure.
2. c **:=** $f(C_s \cap C_r)$. (*$f(x)$ implements the channel selection scheme*)
3. **if** $\{n \in N \,/\, c \notin C_n\} \neq \emptyset$ **then return** selection failure.
4. **for** the sender, the receiver and all the idle neighbors N_{idle}:
    1. sense c and keep updating the set of non-reserved channels.
5. R **:=** set of obtained sensing results.
6. **if** $\{r \in R \,/\, r = (c \text{ is busy})\} \neq \emptyset$ **then return** selection failure.
7. **if** $c \notin C_s$ **or** $c \notin C_r$ **or** $\{n \in N_{idle} \,/\, c \in C_n\} \neq N_{idle}$ **then return** with selection failure.
8. **return** c.

Fig. 3. Cooperative sensing algorithm

(e.g., see the IEEE 8021.11 standard specification and the work of Jia and Zhang [6]). Then, the receiver selects a channel which is considered as being not reserved at both ends, according to a defined selection scheme (see step 2 in Fig. 3), and replies with a CTSS (Clear to Sense) packet. This packet includes the selected channel and an updated reservation time for the CCC. In order to avoid the hidden node problem in multi-hop networks, the sender forwards the information provided by the CTSS packet through an fCTSS (forwarded CTSS) packet. Neighbors overhear the CTSS and/or fCTSS packet and set the CCC as being busy for the specified amount of time. Besides, idle neighbors get also involved in sensing. This cooperative sensing approach aims at increasing efficiency in terms of detection of any activity on the selected channel and, therefore, addresses the hidden primary and secondary user problems. If, upon the reception of an RTS packet, the receiver and the sender have no common channels set as unreserved, the receiver sends a CTSS packet back with no selected channels. Then, the sender forwards this information through an fCTSS and enters a random backoff period. As handshaking is over, neighbors which overhear these packets also set the CCC as being idle.

Every SU which cooperates in sensing switches to the selected channel and senses it, such as the sender and the receiver do. The longer is the sensing time the higher is the accuracy, but also the resulting overhead. Therefore, a tradeoff must be made [4] and will be evaluated in the next section based on simulation results. Neighbors perform sensing for a shorter time (see Fig. 2) than the sender and the receiver. The reason is as follows. If, after sensing, a neighbor concludes that the targeted channel is not vacant (in terms of primary or secondary traffic), it informs the sender or receiver before they enter the decision phase. This is accomplished through a CTS (Clear To Send) packet with the selected channel set as unavailable. Non-idle neighbors, i.e., which do not participate in sensing, also send a similar CTS packet if they have the intended channel set as reserved. As a SU can operate simultaneously on the CCC and on the data channel, it keeps processing incoming control packets while sensing or transmitting data. If no packets were received from neighbors while sensing, the conclusion is that they all considered the targeted channel to be vacant.

3.2 Decision Phase

In the second phase, based on sensing results and on any relevant information provided by neighbors, the receiver first sends a CTS packet with the appropriate state of the selected channel to the sender. As the other control packets, CTS is overheard by neighbors and processed accordingly. It also includes as the others, except RTS, a CCC reservation time. Then, according to the content of the CTS packet and to its own knowledge (provided by neighbors and local sensing), the sender replies with an fCTS (forwarded CTS) packet with the appropriate information. It also starts transmitting the data packet if the channel is considered available. Due to the hidden secondary user problem, the receiver forwards this fCTS packet in turn. The reception of an fCTS packet also dictates the conclusion of the two-phase negotiation by setting the CCC free (i.e., zero reservation time), and indicates the corresponding reservation time if the selected channel was considered available.

3.3 Additional Remarks

In order to work properly, CoSBT-MAC requires the coverage areas to be the same on both radios, i.e., on the CCC and on data channels. The main reason is that if two SUs are neighbors on the CCC we also need them to be neighbors on data channels, and vice-versa. Therefore, due to strong dependency between the coverage area and the center frequency (i.e., the higher the frequency the shorter the coverage area and vice-versa), the CR MAC entity must also adjust the transmission power, when switching to a new channel. This aims at maintaining the area of coverage constant, i.e., similar to the one of the CCC. In the next section, we set it to 100 meters.

The proposed cooperative sensing approach also results in an implicit clustering scheme. In this case, a cluster is defined by the coverage area of the sender-receiver pair. Therefore, it only includes the corresponding pair and the one-hop idle neighbors. This helps addressing the previously mentioned spatial inefficiency of the OR-rule (see section 2.1). The efficiency of this approach obviously depends on the density of idle neighbors around the sender-receiver pair.

With the CoSBT-MAC proposal, a channel is used by a sender-receiver pair only if it is sensed idle by the pair and its idle neighbors, and if it is not set as reserved by any neighboring SU. Therefore, a balanced usage of opportunities is naturally achieved. Besides, CoSBT-MAC is also compatible with the energy detection approach as silent periods, in terms of secondary activity on a given channel, are created in the neighborhood of the sender-receiver pair during sensing periods. Finally, it can be mentioned that the spatial efficiency, in terms of spectrum usage, is increased with CoSBT-MAC as the same channel can be simultaneously used in distinct areas of the multi-hop network.

4 Evaluation Results

The CoSBT-MAC proposal, which was described in the previous section, was implemented on the OMNET++/MiXiM modeling framework [12][13]. As there was no known support for CR in this framework, we first developed a generic CR simulation framework. It was based on the multi-radio model which is illustrated in Fig. 4.

SUs only use two radios. One of them is dedicated to data transmission and is dynamically reconfigurable both in terms of the center frequency and transmission power. The other radio supports the CCC. The MAC modules, which implement the access algorithms, can dynamically adjust their transmission powers in order to limit their transmission ranges. This value was set to 100 meters. The default simple path loss model of MiXiM [13] was used and sensitivity at the physical layer was set to -84 dBm. The optional scanner module in Fig. 4, which is dedicated to sensing, is not directly used as sensing is performed by the reconfigurable radio. Currently, the cognitive radio engine, which is dedicated to learning and decision making, implements a straightforward uniform random channel selection scheme (only non-reserved channels are considered). Time to adjust the radio parameters when switching to a new channel was set to 10^{-4} seconds, such as in the work of Jia and Zhang [6]. The bit rate, both on the CCC and on the data channels, was set to 2 Mbit/s, and the MTU (Maximum Transfer Unit) to 18432 bits, such as in IEEE 802.11. The other PHY and MAC simulation parameters are not specified due to space limit restrictions. Basically, they are the same which are set by default in MiXiM when the designated Mac80211 module and the simple path loss propagation model are used.

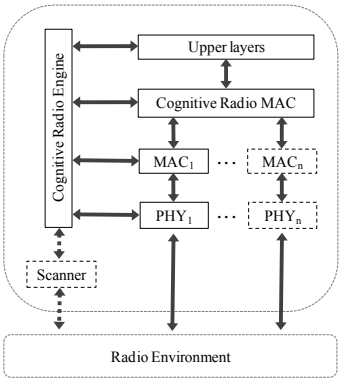


Fig. 4. A secondary user architecture based on a multi-radio model

4.1 Achievable throughput

The performance of CoSBT-MAC is first evaluated in terms of the achievable throughput. Up to six sender-receiver pairs are randomly located in a two dimensional simulation area with 50 meters long sides. Therefore, all the SUs are in the coverage areas of each other, which is the worst possible scenario in terms of contention for the spectrum. Loads up to 3 Mbit/s per sender are considered. Fig. 5, Fig. 6, and Fig. 7 present the obtained results in terms of the average throughput per sender for scenarios with two pairs, four pairs, and more than four pairs, respectively. A comparison is also established with the IEEE 802.11b standard (the RTS/CTS threshold is set to

400 bits) which does not follow any dynamic spectrum access approach, but has lower overhead. We consider the existence of four non-overlapped contiguous channels. PU activity is considered in Fig. 5 and Fig. 6. It is modeled as an alternate stream of idle and busy periods which are exponentially distributed. According to Akyildiz, Lee and Chowdhury [3], this is a common practice. Based on the work of Issariyakul, Pillutla, and Krishnamurthy [14], the parameters of the distributions are set to 0.1 and 0.01 seconds for idle and busy periods, respectively. As mentioned in section 2.1, sensing time dictates the accuracy of local sensing. Therefore, in order to highlight its impact on performance, the experiments are repeated with 0.5 ms and 1.0 ms sensing times. These values were chosen based on the fact that the 802.22 Working Group [2] specifies fast sensing should be under 1 ms per channel.

With two and four sender-receiver pairs, there are enough channels for all of them to transmit in parallel, except when there is any PU activity. From Fig. 5 and Fig. 6, it can be concluded that the CoSBT-MAC proposal brings relevant performance benefits in terms of throughput when compared to IEEE 802.11, despite all the overhead which is introduced by the additional and mandatory sensing phase. It must also be noted that there is a handshaking at least every MTU bits. When PU activity is introduced on the channels (see Fig. 5 and Fig. 6), which is expected to occur in real deployments, results remain concluding. When four sender-receiver pairs are considered (see Fig. 6) instead of just two (see Fig. 5), the performance of CoSBT-MAC slightly decreases when PU activity is considered. This is because the probability that, at a given moment, there are less available channels than sender-receiver pairs willing to transmit is higher. Fig. 7 illustrates scenarios with more sender-receiver pairs than available channels, i.e., with additional contention among secondary users, even when there is no PU activity. Despite the expected degradation of performance, our proposal still outperforms IEEE 802.11 concerning the achievable throughput.

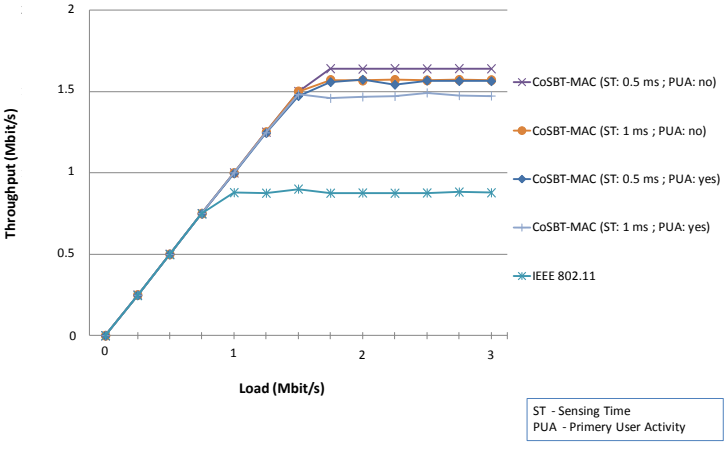


Fig. 5. Average throughput per sender vs. load per sender with two sender-receiver pairs

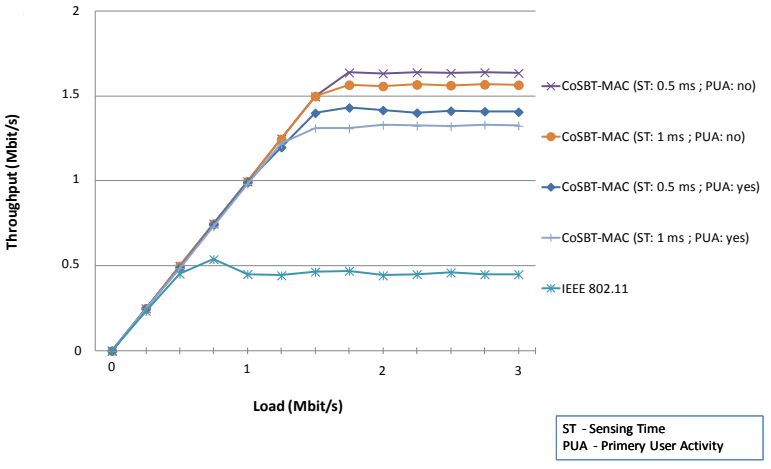


Fig. 6. Average throughput per sender vs. load per sender with four sender-receiver pairs

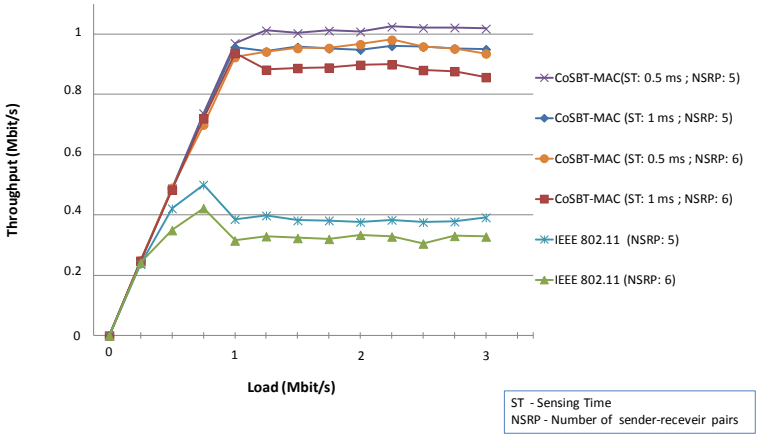


Fig. 7. Average throughput per sender vs. load per sender with five and six sender-receiver pairs, and no primary user activity

4.2 Protection of Primary Users

Table 1 aims at quantifying the contributions which result from the proposed cooperation of idle neighbors in sensing. A comparison is established with the results which are obtained when the cooperation of idle neighbors in sensing is inhibited. The variation in the number of missed PU detections and three different load conditions are considered. With the proposed approach (see section 3), the probability of a SU to be idle is lower for higher loads and, therefore, its chances to help neighbors in sensing

also. Two distinct scenarios are considered: the scenario which is illustrated in Fig. 1; and the same scenario with 100 additional idle neighbors uniformly distributed in the area which includes the five sender-receiver pairs. In Fig. 1, the coordinates of the five sender-receiver pairs and four PUs are fixed. This intends to result in a representative scenario in terms of secondary and primary hidden nodes. Every PU uses a predefined channel and the SUs only have access to this set of four channels (i.e., there are 4 potential non-overlapped channels for 5 sender-receiver pairs). The activity of PUs is modeled as defined in section 4.1, the intermediate 0.5 ms sensing time is considered, and local sensing is assumed to be fully accurate.

Table 1. Contributions of idle neighbors in cooperative sensing

	Variation in the number of missed PU detections[a]	
Load (per sender)	***Scenario in Fig. 1***	***Scenario in Fig. 1 with additional idle neighbors***
1 kbit/s	-39.3%	-53.6%
500 kbit/s	-24.5%	-47,2%
2 Mbit/s	-4.4%	-43,7%

[a] *When compared to the results which are obtained when the cooperation of idle neighbors is inhibited.*

From the presented results, it can be concluded that the proposed cooperative sensing approach effectively contributes for increasing the protection of PUs. Load conditions, the MTU, the characteristics of PU activity, the bit rate, the density of SUs and other factors have an impact on the obtained results, and should be investigated.

5 Conclusions

In this paper, CoSBT-MAC, a CR MAC protocol which follows a "cooperative sensing-before-transmit" approach was proposed and evaluated. It specifically targets fully distributed CR networks which are only based on autonomous SUs, observation, learning, and cooperation. Practicality, low-complexity, scalability, increase of performance which is delivered to SUs, and protection of PUs were the major concerns. Preliminary evaluation results confirm that the proposed approach can result in relevant benefits. The scope of this work was limited to a layer-2 communication protocol issue. However, CoSBT-MAC has great potential for supporting advanced spectrum learning and decision schemes. For instance, as control packets are overheard by neighbors, any node can implicitly and dynamically collect statistics about the success of its neighbors' spectrum decisions and about channel availability. Therefore, no further communication schemes or overheads are needed. On the other hand, as CoSBT-MAC requires the transmission power to be adjusted whenever a channel switch occurs, the interdependency between the carrier frequency and the area of coverage (see section 3.3) can also be exploited in order to address energy efficiency. These issues and others (e.g., energy consumption due to the use of two radios per node, proof of properties such as scalability and low-complexity) will be considered as future work.

References

1. FCC Spectrum Policy Task Force, Report of the spectrum efficiency working group (November 2002)
2. IEEE 802.22 WRAN WG Website (2009), http://www.ieee802.org/22/ (accessed December 2, 2009)
3. Akyildiz, I., Lee, W., Chowdhury, K.: CRAHNs: Cognitive radio ad hoc networks. Ad Hoc Networks 7(5), 810–836 (2009)
4. Cormio, C., Chowdhury, K.: A survey on MAC protocols for cognitive radio networks. Ad Hoc Networks 7(7), 1315–1329 (2009)
5. Choi, N., Patel, M., Venkatesan, S.: A Full duplex multi-channel MAC protocol for multihop cognitive radio networks. In: 1st International Conference on Cognitive Radio Oriented Wireless Networks and Communications, pp. 1–5 (2006)
6. Jia, J., Zhang, Q.: A testbed development framework for cognitive radio networks. In: IEEE International Conference on Communications, ICC 2009, pp. 1–5 (2009)
7. Yuan, Y., Bahl, P., Chandra, R., Chou, P., Ferrell, J., Moscibroda, T., Narlanka, S., Wu, Y.: KNOWS: Cognitive Radio Networks Over White Spaces. In: IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, DySPAN 2007, pp. 416–427 (2007)
8. Timmers, M., Pollin, S., Dejonghe, A., Van der Perre, L., Catthoor, F.: A distributed multichannel MAC protocol for multihop cognitive radio networks. IEEE Transactions on Vehicular Technology 59, 446–459 (2010)
9. Yucek, T., Arslan, H.: A survey of spectrum sensing algorithms for cognitive radio applications. IEEE Communications Surveys & Tutorials 11, 116–130 (2009)
10. Malady, A., da Silva, C.: Clustering methods for distributed spectrum sensing in cognitive radio systems. In: Military Communications Conference, MILCOM 2008, pp. 1–5. IEEE (2008)
11. Marinho, J., Monteiro, E.: Cognitive radio: survey on communication protocols, spectrum decision issues, and future research directions. Wireless Networks (2011), doi:10.1007/s11276-011-0392-1
12. OMNeT++ Network Simulation Framework Website (2010), http://www.omnetpp.org/ (accessed December 2010)
13. MiXiM Website (2010), http://mixim.sourceforge.net/ (accessed December 2010)
14. Issariyakul, T., Pillutla, L., Krishnamurthy, V.: Tuning radio resource in an overlay cognitive radio network for TCP: Greed Isn't Good. IEEE Communications Magazine, 57–63 (2009)

An Evaluation of Vertical Handovers in LTE Networks

Adetola Oredope, Guilherme Frassetto, and Barry Evans

Centre of Communications Systems Research
University of Surrey, Guildford, GU2 7XH, UK
{a.oredope,b.evans}@surrey.ac.uk

Abstract. One of the key requirements of the LTE core network is to provide seamless mobility and session continuity at the IP layer across multiple access networks - 3GPP and non-3GPP. Although this can be achieved in LTE networks using Proxy Mobile IPv6 (PMIPv6), two key limitations still exists - handover latency and high packet loss. In this paper, we provided our proposed, designed and implemented novel protocol known as Proxy Mobile IPv6 Plus (PMIPv6+) aimed at addressing handover latency while reducing signaling traffic in the current standardized PMIPv6 implementations. Our experimental tests and performance analysis shows that PMIPv6+ reduces handover latency by 50% while also reducing the packet loss by almost 95% as compared to legacy PMIPv6.

Keywords: LTE, EPC, SAE, PMIPv6, PMIPv6+.

1 Introduction

In recent years, there has been a rapid increase in the demand for mobile Internet due to the increase of smart phones, new mobile applications and social networks to mention a few. In order to meet these demands, there are currently wide ranges of efforts to evolve the current access, core and service architectures of mobile networks to handle the growing demand for existing and future services. This evolution is driven by the Third Generation Partnership Project (3GPP), a standardisation body formed by groups of telecommunication bodies. The evolved architecture proposed by 3GPP is known as the Long Term Evolution (LTE). LTE is capable of providing higher data rates, compatible with fixed broadband accesses based on the concept of an all-IP network with a new packet core network, the Evolved Packet Core (EPC).

LTE/LTE-Advanced is expected to achieve up to 100Mbps of data throughput, significantly improving the user experience in accessing data services. It also brings enhancements to radio access with reduced delay and latency and high spectral efficiency. In order to achieve this high level of performance, one of the key requirements of the LTE core network is to provide seamless mobility and session continuity at the IP layer across multiple access networks – 3GPP and non-3GPP. As the different access networks have varying requirements for both mobility and session management, the core network has to provide a mechanism to ensure the continuity of IP connectivity during the handover between the access networks, avoiding service interruptions.

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 198–207, 2012.

In this paper, we first outline a wide range of proposed approaches to address this issue, while investigating key technological issues such as handover times, network performance, session management and mobility concepts. Based on the review of the approaches we then highlight the key properties used in defining our proposed and implemented solution – PMIPv6+. Our experimental tests shows that the performance of PMIPv6+ as compared to PMIPv6 reduces the handover latency by 50% while also reducing the packet loss by almost 95% as compared to legacy PMIPv6. Although these are significant improvements to the legacy system, however additional recommendations on how to further optimise the handover process were also suggested in our conclusions.

The rest of the paper is as follows: In Section 2, we discuss a wide range of research efforts aimed at addressing the limitation in PMIPv6 while gathering key requirements for designing our proposed PMIPv6+. Then in Section 3, we provide the requirements and architectural properties of our proposed PMIPv6+ while comparing it to legacy PMIPv6. A wide range of experimental tests to evaluate the performance of PMIPv6+ with corresponding results and analysis are then provided in Section 4. Finally our conclusions and future work are discussed in Section 5.

2 Related Work

Although the protocol of choice in LTE is the PMIPv6 protocol, it does not completely address the handover latency and packet loss issues. During the mobility from one mobile access gateway (MAG) to another, the handover latency relies on the layer 2 attachment/detachment, the proxy binding updates and the network delay. The packets sent to the old MAG when the UE is attached to the new MAG are lost until the local mobile anchor (LMA) updates its routing table with the IP address of the new MAG. To address these issues in PMIPv6, we firstly look at the on going research in the field as discussed below.

Fast Handovers for Proxy Mobile IPv6 (RFC5949) [1] specifies additional procedures to the mobility management in order to reduce the packet loss and the handover latency by ensuring the UE packet flow as soon as it attaches to the new access network during vertical handovers. This extension to the PMIPv6 relies on link layer information sent by the UE meaning that the UE is participating in the mobility decision hence implying in a break of network-based mobility concept. The UE software will need to be developed to provide the required information for the handover. The link layer information is not specified by [1] or [2] and may vary among different AP suppliers. Additionally to that, the nMAG establishes a tunnel with the pMAG to ensure the UE connectivity and then to the LMA. This procedure adds an extra hop to the LMA, by the pMAG-nMAG tunnel, and may increase the latency as all data traffic is transported through this tunnel until the new route with the LMA is established.

An extension of the Fast Handovers for PMIPv6 is proposed in [3] as Enhanced Fast Handovers for PMIPv6 (EFPMIPv6). The main enhancement is in the fact that the nMAG uses the information provided by the HI message to the send the PBU message to LMA and start the establishment of the tunnel with the LMA prior to the

attachment of the UE. The tunnel with the pMAG is still used to recover the stored data in the pMAG [3]. This implementation uses the tunnel between the pMAG and nMAG to send only the buffered data during the handover avoiding the extra delay cost of sending all traffic through this tunnel. But the mechanism chosen still relies on the link layer information sent by the UE and the nMAG starts the proxy binding with the information sent by the pMAG. This may present a risk in case the information provided is not correct or the UE attaches to another MAG.

Another scheme to reduce the packet-loss is proposed by [4], Packet-Lossless PMIPv6 (PL-PMIPv6). In this scheme, the pMAG sends the PBU message in behalf of the nMAG. When the pMAG sends the de-registration PBU to the LMA it also sends the nMAG's PBU to establishes the tunnel between the LMA and the nMAG. The LMA starts sending packets to the nMAG which buffers the packets until the UE is attached to the new AP. The main advantage compared to the previous proposal is that there is no need to establish a second tunnel between pMAG and nMAG and the buffer is set in the nMAG. As the EFPMIPv6, the PL-PMIPv6 is subject to the same risks of wrong prediction information but the fact that the pMAG sends the PBU in behalf of the nMAG adds an extra risk of synchronism between the nMAG and the LMA. When the LMA receives the PBU message from the nMAG it might occur a duplicated entry in the routing table resulting in packet loss.

In [5], a comparative performance analysis demonstrates that the EFPMIPv6 and PL-PMIPv6 have almost the same performance, representing lower latency values compared to the PMIPv6. A mathematical approach was used to compare the handover latency in respect to PMIPv6. The papers [3] and [5] lack an analysis of the packet losses and an evaluation of different transport protocols. The Seamless Handover Scheme for PMIPv6 [6] proposes a scheme where, by the detection of disconnection, the pMAG sends the UE profile to the other MAGs using the Network Discovery (ND) message from IPv6. When the nMAG detects the UE attachment, it establishes a tunnel with the pMAG to get the buffered data packets and sends a PBU message to the LMA with the information collected in the ND messages [6]. This scheme has an advantage compared to the previous ones by not relying on information sent by the UE to establish the tunnel between the pMAG and the nMAG. However, it also represents a security threat as the UE profile is broadcasted in the PMIPv6 domain.

Simultaneous-bindings Proxy Mobile IPv6 (SPMIPv6) [7] proposed an extension to the PMIPv6 protocol that allows simultaneous bindings during the UE handover based on mobility predictions. In this proposal, the pMAG acts as a handover coordinator starting the handover procedure. When the pMAG starts the handover, it sends a Simultaneous PBU (SPBU) to the nMAG. The nMAG sends a PBU message to the LMA with a simultaneous binding flag set to 1, requesting the LMA to bi-cast the packets to both pMAG and nMAG. The nMAG starts to buffer the data packets. Upon the attachment of the UE to the nMAG, it sends a PBU message to the LMA to uni-cast data traffic to nMAG only [7]. Although this proposal relies on handover prediction based on the UE information, it does not need to establish a tunnel between the pMAG and the nMAG and the buffer is done in the nMAG only. However, it requires that the packets to be duplicated and sent to both MAGs. It also introduces new messages, SPBU and SPBA, and modifies the PBU messages by adding an additional flag.

Although the improvements suggested by [1], [3], [4], [6], [7] and the comparison performance demonstrated by [5], these proposal increases the complexity of the PMIPv6 adding more signalling messages between the network elements and rely on predictive information sent by the UE and the access network. The latter insinuates a break of the concept of a network-based protocol. The proposals of an establishment of a direct tunnel between the pMAG and the nMAG may have considered the existence or the desire of direct network connectivity between the MAGs. But in a practical network design, the MAGs might be controlling sets of APs in different access networks with no direct link between those networks but through the main backbone, where the LMA should be located. By this assumption it would more efficient if the LMA function as the main anchor and the only buffer point. Thus, instead of buffering the packets and transferring the packet to the nMAG, the pMAG would send the packets back to the LMA which stores and reorder all the packets until reception of the PBU message of the nMAG.

The analysis and conclusions from these proposals are based on either theoretical values or simulations of TCP/UDP traffic but an analysis of the service level is missing. It is well known that different transport protocols are designed to meet different service requirements as packet loss. TCP uses a re-transmission mechanism to provide reliability with the cost of transmission delay. On the other hand, UDP provides no packet reliability but reduces the transmission delay. File transfer services can tolerate transmission delays in order to receive the correct packets of a file. Voice services have low tolerance to delays and can cope with low packet losses. Therefore, file transfer services use TCP and voice services use RTP/UDP.

3 Protocol Design, Experimental Tests and Analysis

3.1 Proxy Mobile IPv6 Plus (PMIPv6+)

Based on the related work discussed in Section 2, the approach proposed in this paper to tackle the issues presented by PMIPv6 was to deploy a buffering system in the LMA to buffer the input packets, that could not be sent to the UE due to the handover downtime, and re-transmit them to the corresponding MAG, as the UE completes its handover procedure. This enhancement to the PMIPv6 protocol is known as Proxy Mobile IPv6 Plus (PMIPv6+).

In a normal operation of a router, it receives a packet, reads its header to identify the destination address, look for an entry in its routing table that matches the destination address of the packet and forwards the packet to the select network interface. If the destination address does not match any entry, the packet is dropped. Queuing disciplines can be applied to provide priority to packets according to its classification, but they do not store the packets. The router does not keep any record of the packets sent, if a retransmission is needed, it cannot retransmit the packets. The concept of a buffer applied in this paper is to use traffic engineering concepts to delay packets long enough until the handover is completed in order to prevent retransmissions and other signalling overheads. The advantage of this concept is that that handover is transparent to the UE and the delay in packet is seen as a small delay in the network that can be compensated for either by the transport or application layers at the UE.

As the LMA has complete visibility of adjacent MAGs during handovers, it best fitted to have the buffering capabilities. However, various traffic engineering concepts

needs to be studied and explored in other to find the queuing or buffering concept that fits the purpose of the proposed approach. One of the approaches analysed was to manipulate the input queues of the system by redirecting the traffic to an intermediate queue. In the intermediate queue a rule to delay the packets was applied to accommodate the handover period. This rule would be applied to the detection of the IPv6 tunnel disconnection and removed after re-establishing the tunnel. The manipulation of the queues did not work as expected due to the fact that the rules were applied after the routing decision, meaning that the packets were dropped before entering the queue, as there was no route to the destination.

Another approach was to use a network emulation program called netem [8] that can emulate real network conditions such as packet loss, jitter variation, delay, etc. Upon the detection of the IPv6 tunnel disconnection a script started the netem to delay the packets for a period of time large enough to accommodate the handover downtime. After the detection of the IPv6 tunnel connection, the script stopped the netem and the packets were transmitted normally. The netem approach had the same problem faced by the manipulation of the system queues. One advantage compared to the manipulation of the system queues was the fact that its operation was less complex.

As the previous approaches could not process the incoming packets before the routing decision dropped the packets, a new approach was needed in order to intercept the packets before the routing decision. The Linux function ip6tables together with the mangle table can intercept the packet before it reaches the routing function. Therefore, a script was developed to intercept those input packets and send them to a separate queue. Another script written in perl was used to read the queue and apply a delay to the packets. This approach worked fine, but due to the fact that the handover period was varying, it became difficult to estimate a proper delay value to match the handover downtime. If the delay were too short, the packet would be dropped before the handover was completed. If it were too long, the packets would arrive at the destination too late.

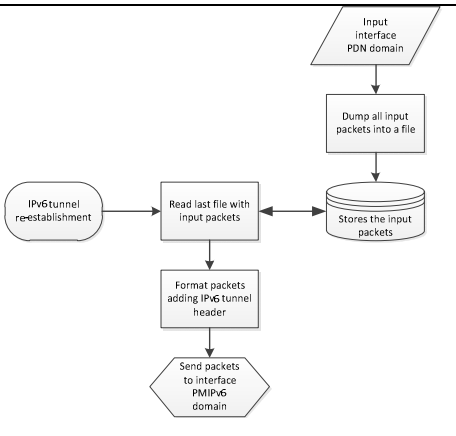


Fig. 1. Algorithm for Buffer Implementation

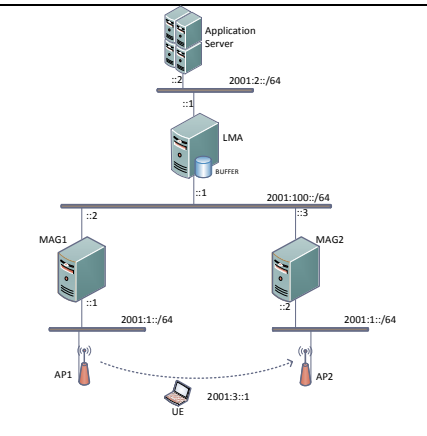


Fig. 2. The PMIPv6+ testbed architecture

Therefore, a new approach should provide the capacity to buffer the incoming packets regardless the routing decision and to send the packets as soon as the IPv6 tunnel is re-established. Using the dumpcap function, all the incoming packets could be stored in .pcap file. A script using scapy [9] was written to read that .pcap file and manipulate the packets in order to add the IPv6 tunnel header. Upon the detection of the tunnel re-establishment, the script would send the packets to the network interface connected to the PMIP domain. The logic of the buffer implementation is depicted in Figure 1. The scapy script was used in the experimental tests to buffer the packets during the UE mobility and re-send the packets to the UE after the handover was completed.

The PMIPv6 development from OpenAirInterface [10] was used to build the test bed depicted in Figure 2. Three servers were configured to work as LMA, MAG1 and MAG2, using virtual machines. The PMIPv6 domain is an IPv6 network and the IPv6 addresses are shown in Figure 2. Two access points, AP1 and AP2, are configured to provide the wireless access link layer to the UE.

In order to demonstrate and evaluate the feasibility of the proposed improvements of the PMIPv6+ the test bed for this project was developed and implemented in order to simulate real life IP mobility conditions within a PMIPv6 domain. The test bed is mainly aimed at providing a controlled platform to experimentally test and analyse IP mobility and session management based on the proposals to PMIPv6 during handovers as the UE from one access network to another.

The performance of the PMIPV6+ system was evaluated by observing the transmission of the ICMPv6 packets, TCP and UDP transport layer packets and video streaming in the application layer. The data were transmitted from the AS to the UE. With those data, an analysis of different network layer could be done providing a more extensive conclusion about the performance of the PMIPv6 protocol. The measurements of the ICMPv6 were carried out using the "ping6" command. The ICMPv6 protocol is commonly used in networks to test the connectivity between network elements and to measure the round trip time (RTT). The transport layer protocols, TCP and UDP, had their performance evaluated using the iperf tool. The iperf tool is used in Linux distribution software to measure the performance of TCP and TCP protocols. It is a very flexible and simple program, allowing setting a sort of different protocol parameters such as IPv4/IPv6, packet size, transmission rate (UDP only), transmission time, etc.

3.2 Experimental Tests and Analysis

Firstly, to validate the test bed system, a wide range of experiments was conducted comprising the attachment of the UE to the AP2 (as shown in Figure 2) followed by a handover from the AP2 to the AP1. The ICMPv6, TCP and UDP protocols were used to validate the correct behavior of the system. In order to validate the system an handover is initiated from AP1 to AP2 while doing a file transfer firstly over TCP them over UDP. The results shown in Figures 3 and 4 shows that during the handover the transmission is interrupted but it restarts after the re-establishment of the wireless link with AP2 and the tunnel between the LMA and the MAG2. The packet sequence number 23 is lost during the handover but the subsequent packets were transmitted normally after the handover.

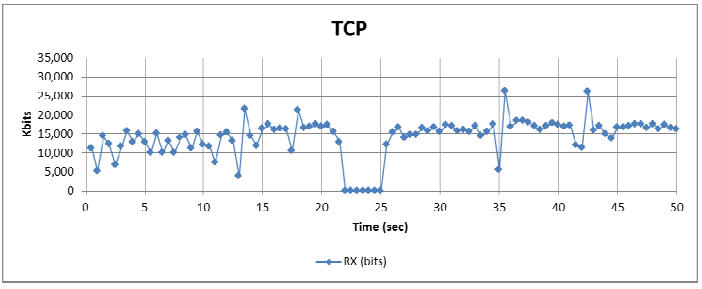


Fig. 3. PMIPv6+ TCP Validation

The same test was repeated for the TCP (Figure 3) and UDP (Figure 4), in different occasions. It can be observed that the TCP protocol downtime is longer compared to the UDP protocol. This is due to the fact that the TCP requested retransmission of the lost packets while the UDP protocol does not use any retransmission or acknowledgement procedure and kept transmitting packets. The ICMPv6, TCP and UDP protocols had different response times during to the handover of the UE between the two access points. This behaviour reflected the differences among those protocols. The ICMPv6 is operates as a request/response protocols, the TCP relies on acknowledgement and re-transmission of packets and the UDP keeps transmitting the packets even if the connection is lost. Despite the difference among those protocols, the tests carried out proved that the system was working properly and the system was ready to conduct further experiments

Once the platform had been validated, the second sets of experiments were then to compare and evaluate the performance of PMIPv6 to PMIPv6+. The tests carried out aimed to collect reference data of the PMIPv6 protocol by the measurement of the TCP performance. Those data are needed to form a fair comparison of the improvements proposed in the PMIPv6+. Figure 5 shows some samples of the experiment for the TCP transmissions. The graph shows the transmission rate versus the time duration of the transmission, from the receiver point of view. The start of the handover is represented by the letter “h” plus (+) or minus (-) the elapsed time, in seconds.

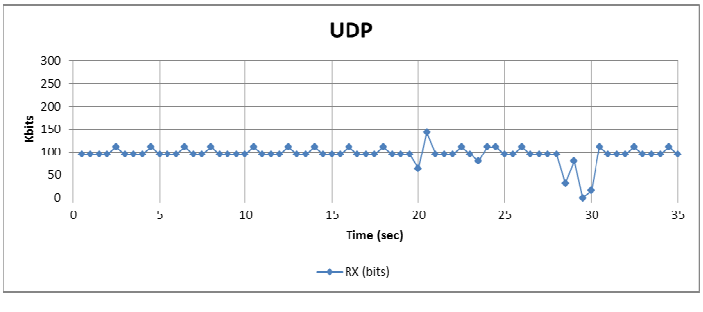


Fig. 4. PMIPv6+ UDP Validation

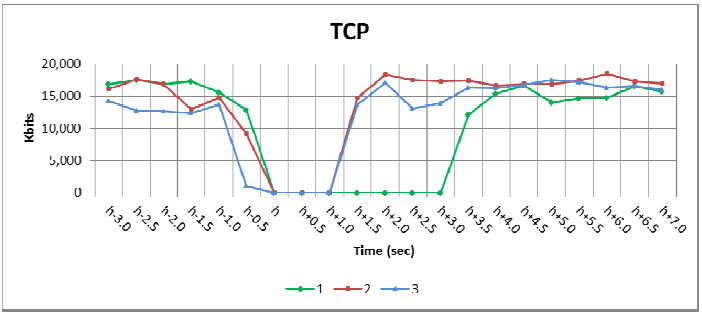


Fig. 5. TCP Performance Evaluation for PMIPv6

Figure 5 show three TCP transmissions, numbers 1, 2 and 3. It can be observed that during the handover the transmissions are interrupted. After the handover the TCP connections are re-established and the TCP protocol started to retransmit the lost packets. The downtime varied from 1 to 3 seconds.

The same experiments were carried out using PMIPv6+ with the LMA was running a script to buffer all the packets received by the AS and forward the packets to active MAG after the handover. The script was based on the proposed algorithm in Section 3.1. Figure 6 indicates that the downtime period varies from 0.5 to 1.0 second. As the TCP protocol in the receiver asks for retransmission of the lost packets, the buffer was already resending some of the lost packets before the transmitter did, reducing the downtime. The buffer in this case acted like anticipating the re-transmission of the TCP packets.

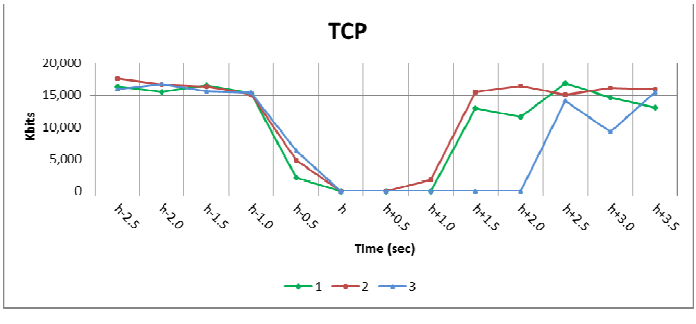


Fig. 6. TCP Performance evaluation for PMIPv6+

As shown in Figure 6, with PMIPv6+, the downtime was reduced compared to using legacy PMIPv6. As a result of the buffered packets in the LMA that where retransmitted to the UE. There was a slight decrease in the TCP downtime around 0.5 to 1 second. In spite of the small improvement when using the buffer, TCP is not used by real-time applications, which are more sensitive to delay and packet loss. The TCP protocol itself can deal with packet losses by negotiating the re-transmission of the lost packets between the receiver and transmitter.

4 Conclusions

In this paper, we has focused on the study of the network session management in LTE networks, looking for ways to enhance the overall performance of the IP mobility, by proposing some optimizations and simple solutions. The work started with a research on the LTE technology and concepts to understand how it deals with the network session followed by a literature review of a wide range of proposals aimed at overcoming the issues of handover latency and signalling overheads related to the PMIPv6 protocol. We have also provided our experiences in developing a novel protocol – PMIPv6+ - aimed at reducing the handover latency and packet loss during handovers from a core network perspective in LTE systems. This has been achieved by investigating ways of optimising the PMIPv6 protocol in which the PMIPv6+ was proposed, designed, developed and tested. However, the solutions still lacks contributions from other affecting factors at the access and application levels that also contributes a wide range of factors to IP mobility.

As shown and discussed in the relate work in Section 2, many proposals still lacks procedures that enables the access network, core network and application exchange key information that facilitates handover decisions in mobile environments. Although this has been attempted in this paper by using system-logging messages to exchange information between the access and core networks, there is presently no standardised way of achieving this and a future protocol to achieve this still need to be investigated.

Another key area that needs to be investigated in the future is the ability for applications to poll key mobility information from the underlying network. That is, the application is aware of a UE mobility status i.e pre-handover, in-handover or post-handover. This will allow the application to dynamically adapt to changes in the mobile node based on its mobility properties.

References

[1] Yokota, H., Chowdhury, K., Koodli, R., Patil, B., Xia, F.: Fast handovers for proxy mobile IPv6. IETF Draft (March 2009) (2010)

[2] Gundavelli, S., Chowdhury, K., Devarapalli, V., Patil, B., Leung, K., et al.: Proxy mobile ipv6 (2008)

[3] Ryu, S., Kim, M., Mun, Y.: Enhanced Fast Handovers for Proxy Mobile IPv6. In: International Conference on Computational Science and Its Applications, ICCSA 2009, pp. 39–43 (2009)

[4] Ryu, S., Kim, G.Y., Kim, B., Mun, Y.: A Scheme to Reduce Packet Loss during PMIPv6 Handover considering Authentication. In: International Conference on Computational Sciences and Its Applications, ICCSA 2008, pp. 47–51 (2008)

[5] Momtaz, A., Khedr, M.E., Tantawy, M.M.: Comparative performance analysis of different Proxy Mobile IPv6 fast handover schemes. In: 2010 International Conference on Information Networking and Automation (ICINA), vol. 1, pp. V1–V452 (2010)

[6] Kang, J.E., Kum, D.W., Li, Y., Cho, Y.Z.: Seamless Handover Scheme for Proxy Mobile IPv6. In: IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, WIMOB 2008, pp. 410–414 (2008)
[7] Bargh, M.S., Hulsebosch, B., Eertink, H., Heijenk, G., Idserda, J., Laganier, J., Prasad, A.R., Zugenmaier, A.: Reducing handover latency in future IP-based wireless networks: Proxy Mobile IPv6 with simultaneous bindings. In: 2008 International Symposium on a World of Wireless, Mobile and Multimedia Networks, WoWMoM 2008, pp. 1–10 (2008)
[8] Hemminger, S.: netem: Network emulator (2004)
[9] Biondi, P.: Scapy (2011)
[10] OpenAir Interface, `http://www.openairinterface.org/` (accessed December 11, 2011)

Connection Cost Based Handover Decision for Offloading Macrocells by Femtocells

Michal Vondra and Zdenek Becvar

Department of Telecommunications Engineering, Faculty of Electrical Engineering
Czech Technical University in Prague
Technicka 2, 166 27 Prague, Czech Republic
michal.vondra@fel.cvut.cz, zdenek.becvar@fel.cvut.cz

Abstract. Femtocells can offload macrocells and reduce a cost of transmitted data in wireless networks. If a connection via the femtocell is of a lower cost than via the macrocell, a time spent by users connected to the femtocells should be maximized. This leads to a reduction of the overall cost of user's connection. Besides, a prolongation of the time spent by users in the femtocells reduces load of the macrocells. Therefore, an extension of handover is presented in this paper. The extension consists in consideration of the connection cost together with user's requirements on a service quality. To that end, a conventional handover decision is modified to achieve higher efficiency in prolongation of the time spent by the users in the femtocells. As the results show, the user who does not require high quality of service spent more time connected to the femtocells and thus the macrocell can be offloaded.

Keywords: femtocell, handover, connection cost, macrocell offloading.

1 Introduction

A concept of home base stations, so-called femtocells was developed to cope with increase in user's demands on throughput. The femtocell is represented by a Femto Access Point (FAP) deployed usually in areas with low level of signal from macrocell (e.g., indoor). The FAPs are typically connected to a network backbone via a wired connection such as xDSL or optical fiber. The FAP can provide three types of user's access: open, closed, and hybrid. All users in the coverage of a FAP can connect to this FAP if it operates in the open access mode. This way, the FAP can offload a Macro Base Station (MBS) by serving several outdoor users. Contrary, the FAP with closed access admits only users listed in so-called Closed Subscriber Group (CSG). This access increases interference to users connected to the MBSs. In the hybrid access mode, a part of capacity is dedicated for the CSG users and the rest of the bandwidth can be shared by other users.

The main purpose of the FAP is to improve indoor coverage for users in the FAP's vicinity or to offload macrocells by serving several outdoor users. Thereby significant increase in throughput is introduced. However, the implementation of the FAPs into the existing network brings several problems that need to be addressed. One of the main tasks is how to handle a handover procedure [1]. A conventional handover decision based on comparison of signals received from a serving and a target station does

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 208–219, 2012.

not take dense deployment and small serving radius of the FAPs into account. A large number of the FAPs in a network increases amount of initiated handovers and decreases Quality of Service (QoS) of users. This effect could be suppressed by common techniques for elimination of redundant handovers such as a hysteresis or a timer [2]. However, these techniques reduce not only amount of handovers, but also a gain in throughput introduced by the FAPs with open or hybrid access [3]. This is due to the small radius of the FAP together with the fact that the conventional techniques always postpone the handover decision. Consequently, the handover to/from the FAP is initiated too late to enable full exploitation of available capacity of the FAPs.

This paper proposes a way how to consider potential lower cost of the connection via the FAP than via the MBS in handover decision. This enables to prolong the time spent by User Equipments (UEs) in the FAP if the FAP provides connection for a lower cost than the MBS. Longer time spent by UEs connected to the FAPs shortens the time when the UE is connected to the MBS. Thus, this approach also offloads MBS and it increases amount of resources available for macrocell users out of femtocells' range. To efficient extension of the time in the FAPs, a modification of the conventional handover decision is described as well.

The rest of the paper is organized as follows. The next section presents related works on handovers in networks with FAPs. In Section 3, modifications of the conventional handover to enable efficient considering of the cost of connection are introduced. In Section 4 and Section 5, the system model and simulation results are presented respectively. The last section gives our conclusions and future work plans.

2 Related Works

Several aspects such as low serving radius or limited backbone connection must be taken into account if the FAPs are deployed. These aspects can lead to an increase in amount of signaling overhead generated due to initiation of large amount of redundant handovers. Therefore, research papers dealing with mobility in a network with the FAPs are usually focused on a reduction of a number of unnecessary handovers.

The possibility of eliminating unnecessary handovers is described, for example, in [4]. The user's speed and a type of service are considered in handover decision algorithm. For users moving with the speed of up to 15 km/h, the handover to the FAP is executed if the signal level of the target FAP exceeds signal level of the serving cell. If the user's speed is in range of 15 km/h and 30 km/h, the type of service is additionally assessed. Handover is executed only if the user is using real-time service. When the user's speed is over 30 km/h, the handover to the FAP is denied.

The idea of the previous paper is further elaborated in [5]. The handover decision is based also on an available bandwidth of the FAP and a category of the user. The UEs are categorized according to their membership in the CSG. A user who is not included in the CSG is connected to the open/hybrid FAP only if three conditions are fulfilled: i) the FAP has available bandwidth, ii) the speed of user is lower than a threshold, and iii) the FAP interferes significantly to the UE connected to the MBS.

In [6], the authors also consider the speed of the user for the decision on handover. Unlike [4] and [5], the speed of users and the cell's configuration influence the setting of time-to-trigger (TTT) parameter.

Another proposal, presented in [7], targets the decrease of number of redundant handovers to the FAP by defining two thresholds, one related to the MBS signal level and the second one related to the FAP signal level. To perform the handover to the FAP, at least one of the following conditions must be fulfilled: i) signal level of the MBS must be lower than the first threshold; or ii) signal level of the FAP must exceed the second threshold. Last, the signal level of the FAP must be above than signal level of the MBS.

Handover for hierarchical macro/femto networks is presented also in [8] and further specified in [9]. The main idea of the proposed algorithm is to combine values of the received signal strength from the serving MBS and the target FAP while considering the large asymmetry in transmit power of both. This mechanism compares the level of signal received from the FAP with absolute threshold value of –72 dB. Besides, the signal of the MBS is compared with combination of signals from the MBS and the FAP. It increases the probability of handover to the FAP if this FAP provides signal above the threshold and if the FAP is deployed far from the MBS. Otherwise, if the threshold is not meet, the conventional handover is performed. The proposed scheme leads to elimination of the handovers if the FAP and MBS are close to each other.

All these proposals are trying to restrict connections of users to the FAPs. However, it leads to a reduction in utilization of the FAPs and the most of UEs stays connected to the MBS. This MBS can easily become overloaded since the FAPs interfere to the UEs connected to the MBS. Hence, those UEs must consume more radio resources to reach required throughput. None of before mentioned methods considers fact that the connection via the FAP can be of a lower cost than the connection through the MBS. Contrary to all above-mentioned proposals, the objective of this paper is to enhance handover decision by consideration of the cost of the connection via the FAPs and the MBSs. Therefore, modifications of the conventional handover with the purpose to increase the time spent connected to the FAPs are presented in this paper. More time spent at the FAP is profitable from an operator as well as from the user's point of view. From the operator side, the advantage is to relieve existing network infrastructure. From the user's perspective, it enables to attain higher transmission rate and/or lower cost of the connection.

3 Handover for Maximization of the Time Spent in Femtocells

The conventional handover decision is based on comparison of the signal level of the target ($\overline{s_t}[k]$) and serving ($\overline{s_s}[k]$) cells. Commonly, a hysteresis margin (Δ_{HM}) can be used in order to mitigate a ping-pong effect (i.e., continuous switching of two neighboring serving stations). The conventional handover is initiated if the next condition is fulfilled:

$$\overline{s_t}[k] > \overline{s_s}[k] + \Delta_{HM} \tag{1}$$

To additional elimination of redundant handovers, a timer (e.g. TTT [10]) can be implemented. The conventional handover algorithm is designed for networks with MBSs only and does not consider specifics of heterogeneous network composed of both MBSs and FAPs.

The conventional handover should be modified to maximize the time spent by users in the FAP and thus to either offload MBSs or reduce the connection cost if the FAP provides lower connection cost than the MBS. The handover decision in the proposed algorithm is based on absolute levels of the Carrier to Interference plus Noise Ratio (CINR), on a trend of the FAP's CINR level (as shown in Fig. 1), and on the acceptable outage for users. The modified algorithm compares the CINR values rather than Received Signal Strength Indicator (RSSI) since the interference significantly influence a quality of a radio channel. Due to consideration of the CINR, the FAP is accessed more effectively at a time when it is able to provide higher throughput.

The proposed handover to the FAP is performed immediately when the CINR level of the FAP (denoted $\overline{s_{FAP}[k]}$) exceeds a threshold $CINR_{T,in}$ as expressed in (2). The $CINR_{T,in}$ is set as a fixed value equal to the minimum level of CINR when the UE can be served by the FAP. In addition to (2), the level of the signal received from the FAP must be rising as well (see (3)). The requirements on the rising signal level provides a certain level of a prediction. Thus, we can assume that the user is moving in a direction to become closer to the FAP. This way, the ping-pong effect is suppressed, the time spent by users in the FAP is maximized, and the UE's outage is not increased.

$$\overline{s_{FAP}[k]} > CINR_{T,in} \tag{2}$$

$$\overline{s_{FAP}[k-1]} < \overline{s_{FAP}[k]} \tag{3}$$

When the UE is leaving the FAP, the handover is initiated according to the absolute CINR level of the FAP as well. Moreover, the trend of the FAP's CINR level and the actual level of the MBS's CINR are also taken into account. The handover from the FAP to the MBS is performed only if the following conditions are fulfilled: i) the CINR level from the FAP is lower than the level $CINR_{T,out}$ as defined in (4); ii) the CINR level of the MBS ($\overline{s_{MBS}[k]}$) exceeds the CINR level of the FAP (see (5)); and iii) the trend of the signal received from the FAP is declining, as expressed in (6).

$$\overline{s_{FAP}[k]} < CINR_{T,out} \tag{4}$$

$$\overline{s_{FAP}[k]} < \overline{s_{MBS}[k]} \tag{5}$$

$$\overline{s_{FAP}[k-1]} > \overline{s_{FAP}[k]} \tag{6}$$

The handover between two FAPs is performed based on the same conditions as in the conventional algorithm (defined in (1)).

To avoid an immediate handover back to the MBS a short timer is considered. During this timer no backward handover can be performed. The imminent handover might occur if the value of $CINR_{T,in}$ is set lower than the value of a threshold for handover from the FAP ($CINR_{T,out}$).

As depicted in Fig. 1, the proposed approach leads to earlier initiation of the handover to the FAP comparing to the conventional handover. Contrary, the connection to the FAP remains for a longer time then in the conventional approach if the user is leaving the FAP. This is since the UE performs the handover only if the FAP is no longer able to satisfy QoS requirements of the UE. Therefore, the threshold $CINR_{T,out}$ must be related to the QoS required by individual UEs. In this paper, the QoS requirements are represented by an outage probability. The outage probability is

expressed as the probability of being in a state when the user cannot transmit data. However, other metrics can be implemented and considered in the same way.

An optimum $CINR_{T,out}$ should be determine with respect to the user's preferences in either cost of the connection or the quality of the connection. The variable threshold $CINR_{T,out}$ enables a consideration of different cost of the connection via the MBS and the FAP. If a user can tolerate lower quality of the connection, it can spent more time connected to the FAPs. Then, an operator benefits from lower load of the MBS. Therefore, the operator can give a benefit, such as discount on cost of services, if the user would accept to stay connected to the FAP for a longer time even if it would lead to minor drop in quality.

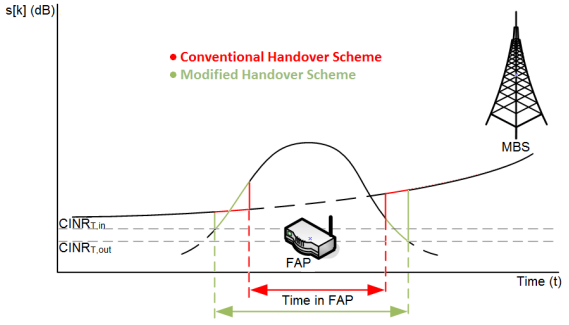


Fig. 1. The principle of the modified handover schemes

4 System Model

First, a model of consideration of the connection cost is introduced. Then, the simulation models are presented.

4.1 Model for Connection Cost

For determining appropriate trade-off between the connection quality and cost, we define three illustrative types of users. Each type represents an example of user's preferences on the outage probability over the connection cost. The first type, *User A*, is aimed primarily on the quality (i.e., low outage) regardless of the connection cost. An example of the *User A* is someone who requires high quality of voice calls. The second type, *User B*, is willing to compromise on the quality requirements for cheaper services. The third type of the user, *User C*, is focused on saving money and does not stress the quality of connection. This user can be seen as someone who uses mainly the services with low requirements on delay, such as e-mail, FTP, or HTTP.

An example of acceptable increase in outage probability over the connection cost ratio for all illustrative types of users is depicted in Fig. 2. The "Cost Ratio" can be expressed as the ratio of the cost of the FAP's connection to the cost of the connection to the MBS. For example, the ratio 1/1 means the same price of the connection via the FAP and the MBS. In this case, the user has preferences for neither FAP nor MBS in term of the cost. On the other hand, the ratio 0/1 corresponds to the situation when the connection via the FAP is for free.

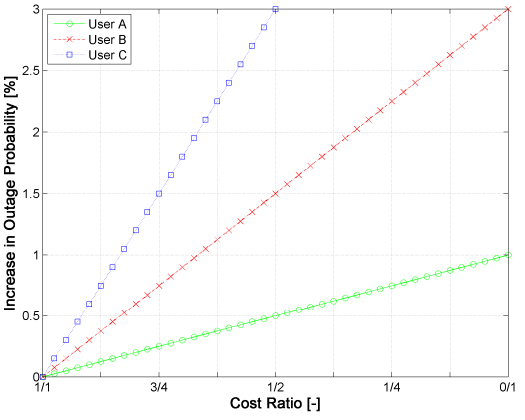


Fig. 2. Acceptable increase in outage for different types of users over ratio of connection cost to FAP and MBS

4.2 Simulation Models

In simulations, twenty-five blocks of flats with the square shape are arranged in a matrix with size of 5 x 5 blocks (see Fig. 3). The size of each block is 100 x 100 meters. Blocks are separated by streets with the width of 10 meters. Each block contains 64 apartments with the size of 10 x 10 meters. The apartments are arranged in two rows around the perimeter.

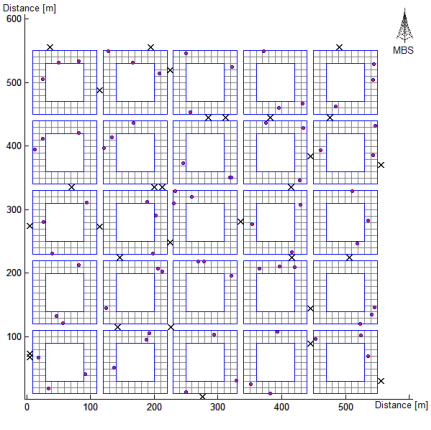


Fig. 3. Example of a random simulation deployment

Three FAPs per a block of flats are randomly placed in random apartments for each drop. All FAPs operate in the open access mode. The MBS is located in the top right corner of the simulation area in distance of approximately 50 meters from the closest flat. Thirty users are randomly deployed in this simulation area. The users are moving along the streets with speed of 1 m/s. Each user covers the distance of 3000 m per a simulation drop, i.e., the duration of each drop is 3000 s of the real time. Indoor users

are not considered in the simulation since the FAP provides signal of sufficient quality to serve the user inside the flat and these users are not supposed to perform handover within the flat.

The quality of signal received by the UE from the FAP is determined according to the ITU-R P.1238 path loss model including wall losses [11]. The Okumura-Hata path loss model for outdoor to outdoor communication [12] is used for derivation of the MBS's signal propagation. For the evaluation of the handover outage probability and the overall outage probability, a CINR Outage Limit ($CINR_{OL}$) is defined. It is the level of the CINR, under which the QoS is not fully guaranteed. It means, the transmission speed and quality of the user's channel are very low. According to [13] and [14], the $CINR_{OL}$ is set to *–3 dB*. The major simulation parameters are summarized in Table 1.

Table 1. Parameters of simulation

Parameters	Value
Frequency / Channel bandwidth	2 GHz / 20 MHz
Transmiting power of MBS / FAP	46 / 15 dBm
Height of macro MBS / FAP / UE	32 / 1 / 1.5 m
External / internal wall loss	10 / 5 dBm
$CINR_{OL}$	–3 dB
$CINR_{T,in}$	–3 dB
Simulation real-time	3 000 s

Three competitive handover algorithms are evaluated for the same movement of users: i) the conventional algorithm based on comparison of RSSI, ii) the conventional algorithm based on comparison of CINR, iii) the Moon's algorithm according to [9] (described in Section 2). These three algorithms are simulated for two levels of hysteresis, i.e., $\Delta_{HM} = 1\ dB$ and $\Delta_{HM} = 4\ dB$. In addition, the modified handover decision proposed in this paper is evaluated and compared with three before mentioned algorithms.

5 Results

The results are split into two subsections. In the first one, the comparison of slightly modified handover decision with the competitive handovers is performed. The second subsection focuses on determination of an optimum threshold for consideration of the connection cost based on the requirements of users.

5.1 Evaluation of the Handover Decision Algorithms

Three parameters are observed and compared: time in the FAP, outage probability, and handover outage probability.

The time spent in the FAP (t_{FAP}) is understood as the average duration of the connection of the UE to the FAP. In other words, it is an average time interval between the handover to the FAP and the handover back to the MBS.

The results of evaluation of t_{FAP} over $CINR_{T,out}$, presented in Fig. 4, show that t_{FAP} is rising with decreasing level of $CINR_{T,out}$ for the proposed handover. Comparing to

the other competitive algorithms, our modification of handover outperforms the Moon's algorithm for all levels of $CINR_{T,out}$. Note that t_{FAP} reached by the Moon's algorithm is nearly independent on hysteresis. Performing handover based on the CINR levels leads to a prolongation of t_{FAP} with increasing hysteresis. However, even for the hysteresis of 4 dB, the proposed scheme achieves higher t_{FAP} if $CINR_{T,out} < -3.4\ dB$. The improvement in t_{FAP} can be reached by the replacement of the conventional CINR based handover decision by the RSSI based one. In this case, the results of the RSSI based handover with hysteresis of 4 dB are the same as results of the proposed algorithm for $CINR_{T,out} = -5.7\ dB$.

According to the results in Fig. 4, it is profitable to either increase the hysteresis for the conventional algorithms or lower $CINR_{T,out}$ for the proposed scheme to increase t_{FAP}. However, higher hysteresis as well as lower threshold $CINR_{T,out}$ can negatively influence the handover outage probability.

The handover outage probability is the ratio of unsuccessful handovers to the overall number of the performed handovers during the simulations. As an unsuccessful handover is understood the handover during which the CINR level drops under the $CINR_{OL}$. According to Fig. 5, the handover outage probability is constant up to $CINR_{T,out} = -2.3\ dB$ for our proposal. Then it rises rapidly and get steady at approximately 50 % of handover outage. This steep increase is caused by the fact that channel quality is not sufficient if the CINR level drops close to the $CINR_{OL}$.

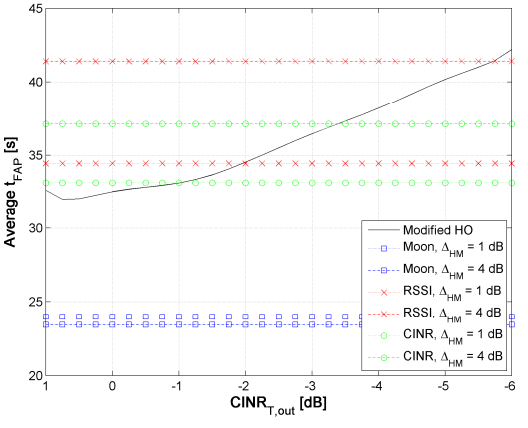


Fig. 4. Average time spent by UEs in connected to the FAP

The handover outage probability comparable with the proposed scheme can be obtained only by the conventional handover based on the CINR with very low hysteresis. Nevertheless, the proposal reaches nearly twice lower handover outage (8 % instead of 15 %). Simultaneously, t_{FAP} is prolonged by 9 % by the proposal as can be observed in Fig. 4. If $CINR_{T,out}$ is above $-2.5\ dB$, a half of handover fails by our proposed procedure. The similar level of handover outage is reached either by the CINR based and the Moon's algorithm with hysteresis of 4 dB. However, in this case, the proposed procedure prolongs t_{FAP} by 14 % and 85 % comparing to the conventional CINR based and the Moon's algorithm respectively for $CINR_{T,out} = -6\ dB$ (see Fig. 4).

Although the RSSI based algorithm shows good results in term of t_{FAP}, the outage is very high even for low hysteresis. It is due to not efficiently chosen times of the handover decision. It means the handover to the FAP and back to the MBS is performed too late comparing to an optimum time instant.

The results for t_{FAP} and handover outage probability can be summarized in two points. First, the proposal can significantly reduce the handover outage probability simultaneously with slight prolongation of t_{FAP}. This is the case when users do not accept high level of the handover outage (outage is reduced from 15 % to 8 % by our proposal). Second, the modified handover algorithm significantly prolongs t_{FAP} and keeps roughly the same handover outage probability if users are willing to tolerate higher level (roughly 50 %) of the handover outage. Therefore, our proposal is profitable for the user who prefers quality as well as for the user who aims low connection cost.

The number of handovers initiated by our modified algorithm is kept at nearly the same level as in case of the conventional handover. The simulation shows only 3 % and 5 % rise in the overall amount of initiated handovers comparing to the CINR and the RSSI based procedure respectively. Comparing to the Moon's procedure, our proposal reduces amount of handovers for approximately 4 %.

Fig. 6 shows the percentage of overall simulation time spent by the UEs in a state of outage. The outage probability is the ratio of the time when the user's requirements are not fulfilled due to the CINR level under the $CINR_{OL}$ to the overall duration of the simulation.

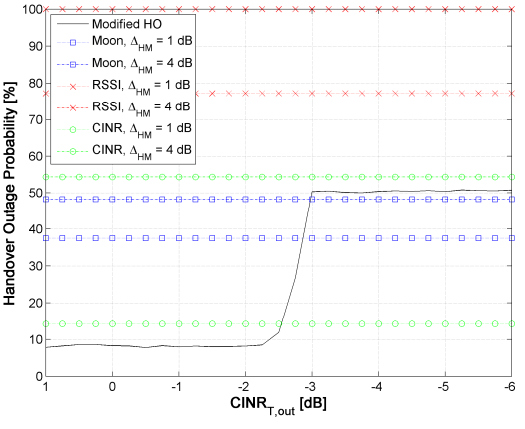


Fig. 5. Handover Outage Probability over the threshold for handover to the MBS

The proposed scheme shows again a constant outage, of roughly 0.2 %, for $CINR_{T,out}$ up to *–2.5 dB*. This outage is the lowest of all evaluated algorithms. Then, the outage probability rises linearly with slope of 1 % per 1 dB for $CINR_{T,out}$ lower than *–2.5 dB*. The handover performed based on the comparison of the CINR reaches very low outage if the hysteresis is set to low value. Nevertheless, the outage is still nearly twice higher than the outage obtained by the proposed handover decision with $CINR_{T,out}$ up to *–2.5 dB*. All other algorithms are outperformed significantly by the proposed one.

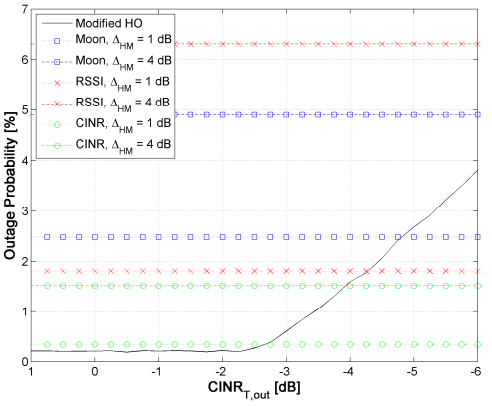


Fig. 6. Outage Probability over the threshold for handover to the MBS

Comparing to all three competitive algorithms, the proposed one provides highest extension of the t_{FAP} with lowest rise in the outage probability. In the proposed handover, the t_{FAP} can be adapted more significantly according to user's requirements on outage probability while the outage rises slowly comparing to other competitive techniques. Therefore, the modified handover algorithm proposed in this paper is more suitable for consideration of the connection cost.

5.2 Optimum CINR Threshold over Connection Cost Ratio

As it is shown in the previous subsection, t_{FAP} rises with lowering $CINR_{T,out}$. However, the outage is also rising with decreasing $CINR_{T,out}$. Therefore, a sort of compromise between t_{FAP} and the outage probability must be found.

Based on the user's requirements and on the connection cost ratio (depicted in Fig. 2), optimum $CINR_{T,out}$ can be determined for each type of users. Fig. 7 shows that the *User C* whose demands on the quality are the lowest can use lower level of $CINR_{T,out}$ (more negative numbers) than other users. The lower threshold results in higher probability of the outage as shown in Fig. 6. However, the *User C* is willing to tolerate an increase in the outage as it prolongs t_{FAP} (see Fig. 4) and thus it reduces the cost of connection.

In contrary, the *User A* prefers high quality regardless of higher connection cost. Therefore, higher threshold must be set to maintain an adequate quality of the connection.

In real networks, the threshold can be derived by an operator from Fig. 7 as a fix value for all users, according to the quality the operator wants to provide. Another option is to let individual users choose their preferences on the quality and cost (as presented in Fig. 2). Then the billing is performed according to the user's selection. It means the operator gives a benefit, e.g., in form of a lower price, to *User C* over *User A* since *User C* consumes fewer resources of MBSs.

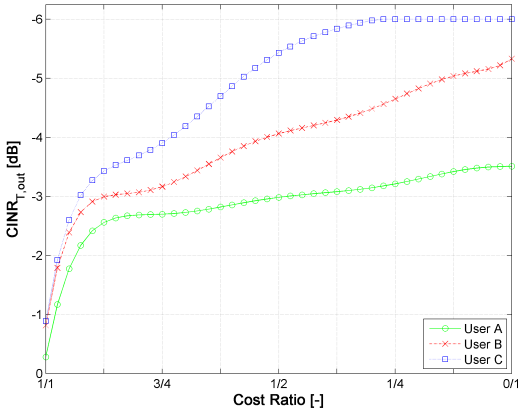


Fig. 7. Threshold for handover to MBS according user's requirements

6 Conclusions

This paper proposes an enhancement of the conventional handover by a consideration of user's requirements on the quality of the connection with respect to the cost of the connection. This way, an operator can give a benefit to the users that are willing to offload its network at the cost of higher outage. The offloading is reached by prolonging the time spend by the UEs connected to the FAPs instead of staying connected to the MBS. To maximize the time spent by UEs connected to the FAP, the conventional handover algorithm is slightly modified. Extension of the time in the FAP is achieved primarily by decreasing the CINR threshold for disconnection from a FAP. This modification kept the number of handovers at nearly the same level as in the case of conventional handovers.

In the future work, we aim on the extension of the metrics to efficiently evaluate users requirements. It means, for example, the throughput, the MBS offloading requirements and other parameters will be taken into account. Simultaneously, the possibility of prediction of the FAP's CINR level will be investigated. This could further prolong the time in FAPs with minimized negative impact on the outage probability.

Acknowledgment. This work has been performed in the framework of the FP7 project FREEDOM IST-248891 STP, which is funded by the European Community. The Authors would like to acknowledge the contributions of their colleagues from FREEDOM Consortium (http://www.ict-freedom.eu).

References

1. Chandrasekhar, V., Andrews, J., Gatherer, A.: Femtocell networks: a survey. IEEE Communications Magazine 46(9), 59–67 (2008)
2. Pollini, G.: Trends in Handover Design. IEEE Communications Magazine 34(3), 82–90 (1996)

3. Zetterberg, K., Scully, N., Turk, J., Jorgušeski, L., Pais, A.: Controllability for self-optimisation of home eNodeBs. In: Workshop COST 2100 SWG 3.1 & FP7-ICT-SOCRATES, Athens, Greece (February 2010)
4. Zhang, H., Wen, X., Wang, B., Zheng, W., Sun, Y.: A Novel Hand-over Mechanism between Femtocell and Macrocell for LTE based Networks. In: Proc. ICCSN 2010 (February 2010)
5. Wu, S.-J.: A New Handover Strategy between Femtocell and Macrocell for LTE-Based Network. In: 4th International Conference on Ubi-Media Computing (U-Media), pp. 203–208 (July 2011)
6. Lee, Y., Shin, B., Lim, J., Hong, D.: Effects of Time-to-Trigger Parameter on Handover Performance in SON-Based LTE Systems. In: 16th Asia-Pacific Conference on Communications (APCC 2010), pp. 492–496 (November 2010)
7. Yang, G., Wang, X., Chen, X.: Handover Control for LTE Femtocell Networks. In: International Conference on Electronics, Communications and Control (ICECC), pp. 2670–2673 (September 2011)
8. Moon, J.-M., Cho, D.-H.: Efficient Handoff Algorithm for Inbound Mobility in Hierarchical Macro/Femto Cell Networks. IEEE Communication Letters 13(10), 755–757 (2009), doi:10.1109/LCOMM.2009.090823
9. Moon, J.-M., Cho, D.-H.: Novel Handoff Decision Algorithm in Hierarchical Macro/Femto-Cell Networks. In: IEEE Wireless Communications and Networking Conference (WCNC 2010), pp. 1–6 (July 2010)
10. Lee, H., Kim, D., Chung, B., Yoon, H.: Adaptive Hysteresis Using Mobility Correlation for Fast Handover. IEEE Communications Letters 12(2), 152–154 (2008)
11. ITU-R P.1238-6 Recommendation. Propagation data and prediction methods for the planning of indoor radiocommunication systems and radio local area networks in the frequency range 900 MHz to 100 GHz (2009)
12. FemtoForum, Interference Management in UMTS Femtocells (2010), `http://www.femtoforum.org/femto/publications.php`
13. Fan, J., Yin, Q., Li, G.Y., Peng, B., Zhu, X.: MCS Selection for Throughput Improvement in Downlink LTE Systems. In: Proceedings of 20th International Conference on Computer Communications and Networks (ICCCN), pp. 1–5 (August 2011)
14. Yu, C., Xiangming, W., Xinqi, L., Wei, Z.: Research on the modulation and coding scheme in LTE TDD wireless network. In: International Conference on Industrial Mechatronics and Automation (ICIMA), pp. 468–471 (July 2009)

Direct Link Assignment Approach for IEEE 802.16 Networks

Chung-Hsien Hsu

Information and Communications Research Laboratories,
Industrial Technology Research Institute
No. 195, Sec. 4, Chung Hsing Rd., Chutung, Hsinchu, Taiwan
stanleyhsu@itri.org.tw

Abstract. The point-to-multipoint (PMP) mode is considered a well-adopted transmission type supported by IEEE 802.16 standard. With the consideration of direct communications among mobile stations (MSs), the required bandwidth and packet latency are reduced. However, inappropriate arrangements of direct communications may introduce additional inter-cell interferences. In order to avoid this situation, a direct link assignment (DLA) approach for each pair of MSs that are expected to conduct direct communication is proposed in this paper. Based on calculated interference region and feasible region as well as predicted motion for each pair, the DLA approach properly arranges the MSs to conduct direct communication. The efficiency of the proposed DLA approach is evaluated via simulations. Simulation studies show that the DLA approach efficiently enhances the performance of user throughput in comparison with the original adaptive point-to-point (APC) approach and the conventional IEEE 802.16 scheme.

Keywords: Direct Link, Interference-aware, IEEE 802.16.

1 Introduction

IEEE 802.16 standards for wireless metropolitan area networks are designed to satisfy various demands for high capacity, high data rate, and advanced multimedia services [1]. In accordance with the specification of IEEE Std 802.16-2009 [2], the point-to-multipoint (PMP) mode is the only solution considered for packet transmission in IEEE 802.16 networks. In the PMP mode, packet transmission is coordinated by the base station (BS), which is responsible for controlling the communication with multiple mobile stations (MSs) in both downlink (DL) and uplink (UL) directions. The inefficiency within the PMP mode occurs while two MSs are intended to conduct packet transmission. It is required for data packets between the MSs to be forwarded by the BS even though the MSs are adjacent with each other. Due to the rerouting process, the communication bandwidth is wasted which consequently increases the packet-rerouting delay.

In order to alleviate the drawbacks resulted from the indirect transmission, a directly communicable mechanism between MSs should be considered in IEEE

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 220–231, 2012.

802.16 networks. Several direct communication approaches have been proposed for different types of networks [3], [4]. The adaptive point-to-point communication (APC) for IEEE 802.16 networks is proposed in [5], which considers the relative locations and channel conditions among the BS and MSs, to switch the transmission operation between direct and indirect manners. Consequently, the communication bandwidth, control overhead, and packet latency are reduced. However, the issues of inter-cell interference was not considered in the approach.

The inappropriate arrangements of direct communications result in inter-cell interferences. For example, in a DL subframe of IEEE 802.16 networks, all the MSs are served as receivers while the transmitter are BSs. In the case that a pair of MSs proceed with direct communication in the DL subframe, the neighbor-cell MSs closing to the pair will suffer the additional inter-cell interference resulted from the direct communication. Consequently, the network performance will go down. In order to enhance the network throughput, a scheduling algorithm for each pair of MSs that are expected to conduct direct communication is proposed. The interference region and feasible region for the pair are studied and calculated. Based on these two types of information as well as a motion predication mechanism, the direct link assignment (DLA) approach is presented to properly arrange opportunities (i.e., in DL subframes or UL subframes) for conducting direct communication. The efficiency of the proposed DLA approach is evaluated and compared via simulations. Simulation results show that the proposed DLA mechanism outperforms the original APC approach and the conventional IEEE 802.16 mechanism in terms of user throughput.

The remainder of the paper is organized as follows. Section 2 briefly describes the considered signal propagation model and mobility model. The interference region and feasible region for a direct communication pair are explained in Section 3. Section 4 describes the proposed DLA approach; while the performance evaluation of the proposed DLA approach is illustrated in Section 5. Section 6 draws the conclusions.

2 Preliminaries

2.1 Signal Propagation Model

The effects of both path loss and shadowing are considered to characterize signal propagation. Since it is difficult to obtain a signal model that characterizes path loss accurately across a range of different environment, the considered path loss model with a simplified formulation is defined as

$$P_L = K + 10\tau \log_{10}(\frac{d}{d_0}) + \psi \tag{1}$$

with

$$K = 20 \log_{10}(\frac{4\pi d_0}{\lambda}), \tag{2}$$

where K is an intercept which is the free-space path loss at reference distance d_0, and λ is the wavelength in meters. The variable τ is the path loss exponent; while d is the distance between the transmitter and receiver. The shadowing component is denoted as ψ, which is a zero-mean Gauss-distributed random variable with variance σ_ψ^2.

2.2 Mobility Model

The GMM model [6] is considered in this work to represent the motion of each MS. Comparing with the random walk mobility model [7] that results in sharp turns and sudden stops, a more realistic motion trajectory can be obtained with the GMM model. Let α_k and V_k denote the moving direction (with respect to the positive x-axis) and velocity of an MS at a discrete time instant t_k respectively. Based on the GMM mode, both the moving direction and velocity can be formulated as [8]

$$\alpha_k = \gamma_{k-1}\alpha_{k-1} + (1-\gamma_{k-1})\bar{\alpha} + \sqrt{(1-\gamma_{k-1}^2)}\Upsilon_{\alpha_{k-1}}, \tag{3}$$

$$V_k = \varrho_{k-1}V_{k-1} + (1-\varrho_{k-1})\bar{V} + \sqrt{(1-\varrho_{k-1}^2)}\Upsilon_{V_{k-1}}, \tag{4}$$

where $\bar{\alpha}$ and $\bar{V}$ represent the asymptotic mean of the moving direction and velocity as $t_k \to \infty$. The $\Upsilon_{\alpha_{k-1}}$ and $\Upsilon_{V_{k-1}}$ are zero-mean Gaussian-distributed random variables; while γ_{k-1} and ϱ_{k-1} are time-varying parameters that represent different levels of randomness as $0 \leq \gamma_{k-1}, \varrho_{k-1} \leq 1$. Two extreme cases correspond to the linear motion (as $\gamma_{k-1} = \varrho_{k-1} = 1$) and Brownian motion (as $\gamma_{k-1} = \varrho_{k-1} = 0$).

3 Feasible Region Analysis

In this section, the feasible region for a direct communication pair is derived from the analyzed interference region. Given a communication pair $\mathbf{L} = \{N_T, N_R\}$, where N_T and N_R denote the transmitter and receiver respectively, the interference region for the pair $\mathbf{L}$ is defined as a region $\mathbf{R_{IR}^L}$ within which N_R will be interfered by an unrelated transmitter $\tilde{N}_T$. On the other hand, the feasible region for the pair $\mathbf{L}$ is defined as a region $\mathbf{R_{FR}^L}$ within which the communication between N_T and N_R can be conducted successfully.

Considering an on-going communication pair $\mathbf{L} = \{N_T, N_R\}$ and the distance between N_T and N_R is $D^{\mathbf{L}}$, the received power of signal transmitted from N_T at N_R is denoted as $P_r^{\mathbf{L}}$, which can be derived from (1) and (2) as

$$P_r^{\mathbf{L}} = P_t^{\mathbf{L}} \left(\frac{\lambda}{4\pi d_0}\right)^2 \left(\frac{d_0}{D^{\mathbf{L}}}\right)^{\tau} 10^{\frac{-\psi^{\mathbf{L}}}{10}}, \tag{5}$$

where $P_t^{\mathbf{L}}$ is the transmitted power of the signal at N_T. Let $P_t^{\tilde{N}_T}$ represent the power of interference signal transmitted by $\tilde{N}_T$ and the distance between $\tilde{N}_T$

and N_R is D^I. Similarly, with (1) and (2), the received power of interference signal at N_R can be obtained as

$$P_r^I = P_t^{\tilde{N}_T} \left(\frac{\lambda}{4\pi d_0}\right)^2 \left(\frac{d_0}{D^I}\right)^\tau 10^{\frac{-\psi^I}{10}}. \tag{6}$$

The signal-to-interference ratio at N_R is given as $\Phi_{N_R} = P_r^{\mathbf{L}}/P_r^I$, which should be grater than a threshold $\mathfrak{T}_\Phi$. Based on (5) and (6), the Φ_{N_R} can be acquired as

$$\Phi_{N_R} = \frac{P_r^{\mathbf{L}}}{P_r^I} = \frac{P_t^{\mathbf{L}}}{P_t^{\tilde{N}_T}} \left(\frac{D^I}{D^{\mathbf{L}}}\right)^\tau 10^{\frac{\psi^I - \psi^{\mathbf{L}}}{10}} \geq \mathfrak{T}_\Phi. \tag{7}$$

The relationship between D^I and $D^{\mathbf{L}}$ is acquired from (7) as

$$D^I \geq D^{\mathbf{L}} \sqrt[\tau]{\frac{\mathfrak{T}_\Phi}{\Theta}}, \tag{8}$$

where

$$\Theta = \frac{P_t^{\mathbf{L}}}{P_t^{\tilde{N}_T}} 10^{\frac{\psi^I - \psi^{\mathbf{L}}}{10}}. \tag{9}$$

Based on (8), the interference region for the pair $\mathbf{L} = \{N_T, N_R\}$ can be derived as a circle region centered at N_R with radius of $\min\{D^I\}$, i.e.,

$$\mathbf{R}_{\mathbf{IR}}^{\mathbf{L}} = \Gamma\left(N_R, D^{\mathbf{L}} \sqrt[\tau]{\frac{\mathfrak{T}_\Phi}{\Theta}}\right), \tag{10}$$

where $\Gamma(\cdot)$ is a function to calculate a circle area. In other words, if the transmitter $\tilde{N}_T$ exists within $\mathbf{R}_{\mathbf{IR}}^{\mathbf{L}}$, N_R will not successively receive the signal transmitted from N_T.

3.1 Feasible Region in DL Subframe

In order to study the feasible region for a direct communication pair in a DL subframe, a simplified scenario is considered as shown in Fig. 1. It is assumed that $\mathbf{L}_2 = \{N_{T_2} = MS_2, N_{R_2} = MS_3\}$ denotes a pair of MSs that are expected to conduct the direct communication served by BS_2, and a conventional communication pair $\mathbf{L}_1 = \{N_{T_1} = BS_1, N_{R_1} = MS_1\}$ is served by the neighboring BS_1. It is noted that direct communication pairs should not intervene the conventional ones. Therefore, the distance between N_{T_1} and N_{R_1} (i.e., $D^{\mathbf{L}_1}$) is assumed to be equal to the radius of the BS's transmission range R. In other words, the strictest case that MS_1 exists on the cell edge of BS_2 is considered. Intuitively, MS_2 should not exist within the interference region of $\mathbf{L}_1$, which can be obtained from (10) as

$$\mathbf{R}_{\mathbf{IR}}^{\mathbf{L}_1} = \Gamma\left(MS_1, R \sqrt[\tau]{\frac{\mathfrak{T}_\Phi}{\Theta}}\right). \tag{11}$$

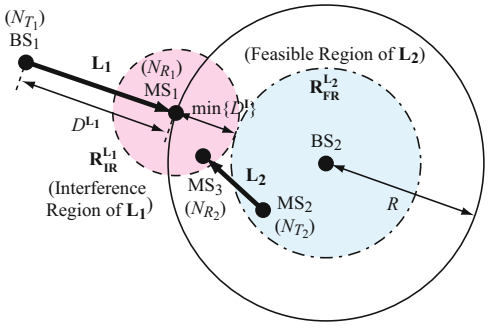


Fig. 1. Schematic diagram of feasible region for direct communication pair $\mathbf{L}_2 = \{N_{T_2}, N_{R_2}\}$ (i.e, MS_2 communicates with MS_3) in DL subframe

As for MS_3, the receiver of $\mathbf{L}_2$, it can be located anywhere within the transmission range of MS_2. Therefore, for the scenario shown in Fig. 1, the feasible region of $\mathbf{L}_2$ is the service range of BS_2 except for the area overlapped by $\mathbf{R}_{\mathbf{IR}}^{\mathbf{L_1}}$.

Based on the aforementioned observations, the feasible region for a direct communication pair in a DL subframe can be inferred as follows:

Corollary 1. *Given a direct communication pair* $\mathbf{L} = \{N_T, N_R\}$ *served by a BS, the feasible region for* $\mathbf{L}$ *in a DL subframe (i.e.,* $\mathbf{R}_{\mathbf{FR,DL}}^{\mathbf{L}}$*) is a circle region centered at the BS with radius of* $R(1 - \sqrt[\tau]{\frac{\Upsilon_\Phi}{\Theta}})$*, where* R *is the radius of BS's service range. If* N_T *is located within* $\mathbf{R}_{\mathbf{FR,DL}}^{\mathbf{L}}$*, the direct communication between* N_T *and* N_R *can be conducted successfully.* *None*

3.2 Feasible Region in UL Subframe

It is similar to the case in the DL subframe, Fig. 2 illustrates a simplified scenario that consists of two communication pairs attached to two neighboring cells, respectively, in a UL subframe. As can be seen that $\mathbf{L}_1 = \{N_{T_1} = MS_1, N_{R_1} = BS_1\}$ is a conventional communication pair served by BS_1. On the other hand, BS_2 serves a pair of MSs that are expected to conduct direct communication, i.e., $\mathbf{L}_2 = \{N_{T_2} = MS_2, N_{R_2} = MS_3\}$. Since MS_1 is a transmitter in the UL subframe, $\mathbf{L}_1$ will never be disturb by $\mathbf{L}_2$, but $\mathbf{L}_1$ may interfere with $\mathbf{L}_2$. In this case, if MS_1 does not exist within the interference region of MS_3, the direct communication will be conducted successfully. Based on this observation, the feasible region for a direct communication pair in a UL subframe can be inferred as follows:

Corollary 2. *Given a direct communication pair* $\mathbf{L} = \{N_T, N_R\}$ *served by a BS, the feasible region for* $\mathbf{L}$ *in a UL subframe (i.e.,* $\mathbf{R}_{\mathbf{FR,UL}}^{\mathbf{L}}$*) is a circle region centered at the BS with radius of* $(R - D^{\mathbf{L}} \sqrt[\tau]{\frac{\Upsilon_\Phi}{\Theta}})$*, where* R *is the radius of BS's service range and* $D^{\mathbf{L}}$ *is the distance between* N_T *and* N_R*. If* N_R *is located*

within $\mathbf{R}^{\mathbf{L}}_{\mathbf{FR,UL}}$*, the direct communication between* N_T *and* N_R *can be conducted successfully.* *None*

It is worthwhile to mention that the feasible region for a direct communication pair is a fixed region in a DL subframe; while in the UL subframe, it is a variable region that is changed according to the distance between the transmitter and receiver of the pair.

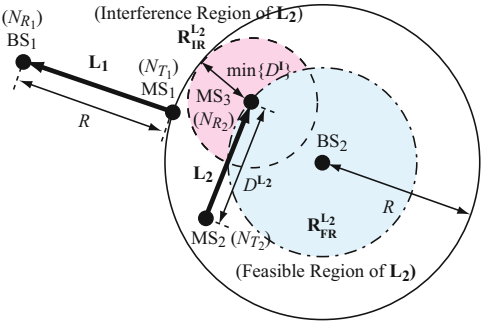


Fig. 2. Schematic diagram of feasible region for direct communication pair $\mathbf{L}_2 = \{N_{T_2}, N_{R_2}\}$ (i.e, MS_2 communicate with MS_3) in UL subframe

4 Direct Link Assignment (DLA) Approach

Since MSs may move around and their actual position information will not always be obtained by the BS, a predication mechanism should be considered to estimate the positions of MSs. As mentioned in Section 2.2, the GMM model is considered in this work. In order to estimate parameters γ_k and ϱ_k at the time instant t_k, both (3) and (4) can be combined and rewritten as

$$\underline{y}_k = \mathbf{H}_k \underline{\gamma}_k + \underline{v}_k, \tag{12}$$

where

$$\underline{y}_k = \begin{bmatrix} \alpha_k \\ V_k \end{bmatrix}, \qquad \mathbf{H}_k = \begin{bmatrix} \alpha_{k-1} & \bar{\alpha} & 0 & 0 \\ 0 & 0 & V_{k-1} & \bar{V} \end{bmatrix},$$

$$\underline{\gamma}_k = \begin{bmatrix} \gamma_{k-1} \\ 1-\gamma_{k-1} \\ \varrho_{k-1} \\ 1-\varrho_{k-1} \end{bmatrix}, \qquad \underline{v}_k = \begin{bmatrix} \sqrt{(1-\gamma^2_{k-1})}\Upsilon_{\alpha_{k-1}} \\ \sqrt{(1-\varrho^2_{k-1})}\Upsilon_{V_{k-1}} \end{bmatrix}.$$

The state vector $\underline{y}_k$ contains the moving direction α_k and velocity V_k of the MS at time instant t_k. $\mathbf{H}_k$ is the design matrix for parameter estimation, while $\underline{\gamma}_k$ represents the state vector for the time-varying parameters γ_{k-1} and ϱ_{k-1} of the

MS at time instant t_k. The vector $\underline{v}_k$ denotes the system noises that are scaled from the random variables $\Upsilon_{\alpha_{k-1}}$ and $\Upsilon_{V_{k-1}}$. Therefore, the parameters γ_k and ϱ_k can be estimated (i.e., $\hat{\underline{\gamma}}_{k+1}$) at the time instance t_k by solving (12) using the recursive least square (RLS) estimation as

$$\hat{\underline{\gamma}}_{k+1} = \hat{\underline{\gamma}}_k - \mathbf{K}_{k+1}(\mathbf{H}_{k+1}\hat{\underline{\gamma}}_k - \underline{y}_{k+1}) \tag{13}$$

with

$$\begin{aligned} \mathbf{K}_{k+1} &= \mathbf{G}_{k+1}\mathbf{H}_{k+1}^T, \\ \mathbf{G}_{k+1} &= \frac{1}{\lambda}\left[\mathbf{G}_k - \frac{\mathbf{G}_k\mathbf{H}_{k+1}^T\mathbf{H}_{k+1}\mathbf{G}_k}{\lambda + \mathbf{H}_{k+1}\mathbf{G}_k\mathbf{H}_{k+1}^T}\right], \end{aligned} \tag{14}$$

where the adjustable parameter λ determines the convergence rate of the RLS method. As the parameters γ and ϱ are estimable with online adaptation, the further position of the MS can be forecasted based on the current state information.

It is assumed that the state information of an MS, including its position $\mathbf{P}_c = \{x_c, y_c\}$, moving direction α_c, and velocity V_c, is obtained at a time instant t_c via its position system. With (3) and (4), the position of the MS at the further time instant t_{c+n} for $n \in \mathbb{N}_1 = \{1, 2, \cdots\}$ can be estimated as

$$\begin{aligned} \hat{x}_{c+n} &= x_{c+n-1} + V_{c+n-1}\triangle t \cos\alpha_{c+n-1}, \\ \hat{y}_{c+n} &= y_{c+n-1} + V_{c+n-1}\triangle t \sin\alpha_{c+n-1}, \end{aligned} \tag{15}$$

where $\triangle t$ is the sampling interval between the time instants t_{c+n} and t_{c+n-1}.

With consideration of the motion predication mechanism, a direct link assignment (DLA) approach is proposed to arrange a proper communication period for a pair of MSs that are expected to conduct direct communication. Based on the aforementioned feasible regions, communication of the pair will be arranged in the DL subframe or UL subframe, or be transferred to the conventional communication manner (i.e, via the BS). Let $\mathbf{L} = \{N_T, N_R\}$ denote the targeted direct communication pair, where N_T and N_R are MSs. Fig. 3 illustrates the flowchart of the DLA mechanism, which consists of nine important steps. The detailed description of these steps are given in the following paragraphs.

Step 1: The availability of direct communication for $\mathbf{L}$ during the DL subframe is determined in this step. Let $\mathbf{P}_m^{N_T}$ represent the position of N_T at a scheduling epoch t_m. Based on Corollary 1, the DL feasible region of $\mathbf{L}$ at t_m is calculated as

$$\mathbf{R}_{\mathbf{FR,DL}}^{\mathbf{L}}(t_m) = \Gamma\left(\mathbf{P}^{N_B}, R(1 - \sqrt[\tau]{\frac{\mathfrak{T}_\Phi}{\Theta}})\right), \tag{16}$$

where $\Gamma(x, y)$ is a function to calculate a circle area centered at x with radius of y; while $\mathbf{P}^{N_B}$ and R denotes the position of the BS (N_B) and its service range

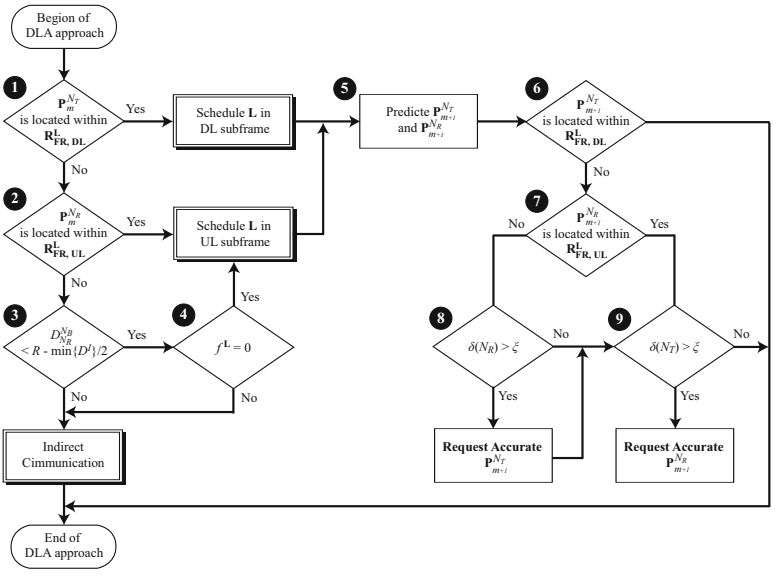


Fig. 3. Flow diagram of DLA Mechanism

respectively. If $\mathbf{P}_m^{N_T}$ is covered by $\mathbf{R}^{\mathbf{L}}_{\mathbf{FR,DL}}(t_m)$, **L** will be scheduled in the DL subframe and the scheduling will be completed; otherwise, it will go to Step 2.

Step 2: This step is utilized to determine if the direct communication for **L** can be conducted in the UL subframe. Based on Corollary 2, the UL feasible region of **L** at t_m is obtained as

$$\mathbf{R}^{\mathbf{L}}_{\mathbf{FR,UL}}(t_m) = \Gamma\left(\mathbf{P}^{N_B}, R - D^{\mathbf{L}}\sqrt[\tau]{\frac{\mathfrak{T}_\Phi}{\Theta}}\right), \tag{17}$$

where $D^{\mathbf{L}}$ is the distance between N_T and N_R. Let $\mathbf{P}_m^{N_R}$ denote the position of N_R at t_m. If $\mathbf{P}_m^{N_T}$ is covered by $\mathbf{R}^{\mathbf{L}}_{\mathbf{FR,UL}}(t_m)$, **L** will be scheduled in the UL subframe and the scheduling will be completed; otherwise, it will go to Step 3.

Step 3: The heuristic scheme assumes that the MSs with conventional communications may exist $\min\{D^I\}/2$ away from the boundary edge of the BS's service range, where $\min\{D^I\}$ is the radius of interference region for **L**. Therefore, the radius of $\mathbf{R}^{\mathbf{L}}_{\mathbf{FR,UL}}$ becomes $R - \min\{D^I\}/2$. In Step 3, the availability of direct communication for **L** is examined again, which checks whether N_B is located within the new $\mathbf{R}^{\mathbf{L}}_{\mathbf{FR,UL}}$, i.e.,

$$D_{N_R}^{N_B} < R - \frac{\min\{D^I\}}{2} = R - \frac{1}{2}D^{\mathbf{L}}\sqrt[\tau]{\frac{\mathfrak{T}_\Phi}{\Theta}}, \tag{18}$$

where $D_{N_R}^{N_B}$ is the distance between N_B and N_R. If $\mathbf{P}_m^{N_R}$ is covered by the new $\mathbf{R}^{\mathbf{L}}_{\mathbf{FR,UL}}$, then go to step 4. Otherwise, **L** will not be conducted either in DL

subframe or UL subframe. The communication between N_T and N_R will be transferred to the conventional communication.

Step 4: The heuristic scheme is utilized to provide another chance of direct communication for **L**. If the chance results in a failed communication between N_T and N_R, the conventional communication scheme will be provided for N_T and N_R. In Step 4, therefore, **L** will be scheduled in DL subframe if $f^{\mathbf{L}} = 0$, which represents that the previous communication of **L** is success. Otherwise, the direct communication for **L** will not be permitted.

Step 5: Estimate the future positions of N_T and N_R based on (15).

Step 6: Based on the estimated position of N_T, the possibility of direct communication for **L** in the next DL subframe is calculated in this step. The DL feasible region for **L** is utilized to determine the calculating result. If the result shows that **L** can be arranged in the next DL subframe, the algorithm will be complete. Otherwise, it will go to Step 7.

Step 7: This step is utilized to predict whether the direct communication between N_T and N_R can be conduct in the next UL subframe. According to the estimated distances of $D^{\mathbf{L}}$ and $D_{N_R}^{N_B}$, the DL feasible region is acquired to predict the availability of direct communication for **L**. If the communication is permitted, the algorithm will execute Step 9. Otherwise, it will go to Step 8.

Step 8: Let $\delta(N_R)$ denote the number of times to predict N_R's position and ξ is a pre-defined threshold value. With the increment of $\delta(N_R)$, the predicted error for N_R's position may also be raised. In this step, therefore, $\delta(N_R)$ is examined. If $\delta(N_R) > \xi$, the BS will request N_R to provide its actual position information. Otherwise, it will go to Step 9.

Step 9: The functionality of this step is similar to that of Step 8, but the targeted MS is N_T instead of N_R. Let $\delta(N_T)$ denote the number of times to predict N_T's position. If $\delta(N_T) > \xi$, the BS will request N_T to provide its actual position information and complete the algorithm. Otherwise, the algorithm will directly be ended.

5 Performance Evaluation

In this section, simulations are conducted to evaluate the performance of the proposed DLA approach in comparison with the original APC approach and the conventional IEEE 802.16 mechanism. A 19 cell-based wrap around topology [9] is considered as the simulation layout. Each cell consists of a centered BS and various number of uniformly distributed MSs. Both the intra-cell and inter-cell traffic flows are considered in the simulation, wherein each MS generates 20 traffic flows with randomly selected destination. The packet arrival process of each flow is assumed to follow a Poisson process with rate $\lambda = 2$ packet/frame. The size of each packet is selected to follow the exponential distribution with the mean value of 200 bytes. Since the scheduling algorithm is not specified in the IEEE 802.16 standard, the deficit round robin (DRR) [10] and weighted round robin (WRR) [11] algorithms are selected as the BS's DL and UL schedulers

respectively. The DRR algorithm is also utilized by the MS to share the UL grants that are provided by the BS for its corresponding UL connections. The simulation is implemented via MATLAB even-driven simulator. Each obtained result is average from 100 simulation runs.

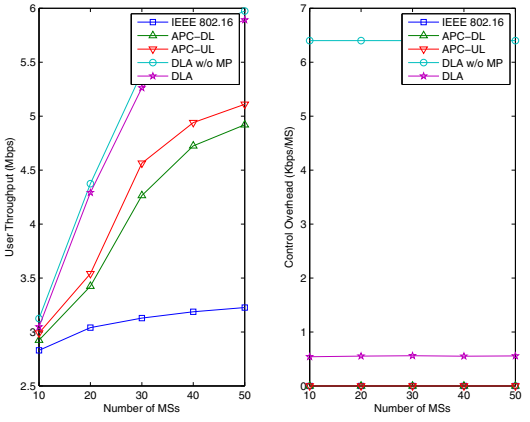


Fig. 4. Performance comparison of user throughput and control overhead versus number of MSs

Fig. 4 illustrates the performance of the user throughput and control overhead under various number of MSs for the compared schemes. The mean velocity of 10 m/s is considered for all MSs in this comparison. As can be expected that the user throughput increases as the number of MSs is augmented for all the schemes. Due to the effect of direct communication conducted among MSs, the performance of the APC-based approaches outperform that of the conventional indirect scheme (denoted as IEEE 802.16). For the original APC approach (i.e.,without the consideration of scheduling algorithm for directly communicable pairs), it is observed that conducting direct transmissions in UL subframes (denoted as APC-UL) has better performance than that in DL subframes (denoted as APC-DL). The reason can be contributed to the potential inter-cell interferences to neighbor-cell MSs that are introduced by conducting direct communication in DL subframes, which can consequently decrease the entire system performance. If the direct transmissions are arranged in UL subframes, the conventionally indirect communication pairs will not be interfered since all the MSs are served as transmitters during those UL subframes. With the consideration of scheduling algorithms for directly communicable pairs in the APC approach (i.e., the DLA w/o MP and DLA schemes), the enhanced performance of user throughput is acquired. It is because that the directly communicable pairs are properly arranged in DL subframes or UL subframes without inducing additional interference. Comparing the DLA w/o MP (i.e., DLA without motion predication)

and DLA approaches, it is observed that the higher user throughput and control overhead are shown in the DLA w/o MP scheme, which can be attributed to the acquirement of MSs position information. In the DLA w/o MP scheme, the exact position information of MSs are acquired via the updates of MSs, which results in large amount of control overhead. On the other hand, the position information of MSs are properly estimated by the BS in the DLA scheme, which efficiently reduces the necessary of position updates by MSs. Consequently, the lower control overhead is shown in the DLA approach in comparison with the DLA w/o MP scheme. Noted that the control overhead means the overhead related to the position updates of MSs. Since the position information of MSs is not utilized in the APC-DL, APC-UL, and IEEE 802.16 approaches, there is no control overhead in these schemes.

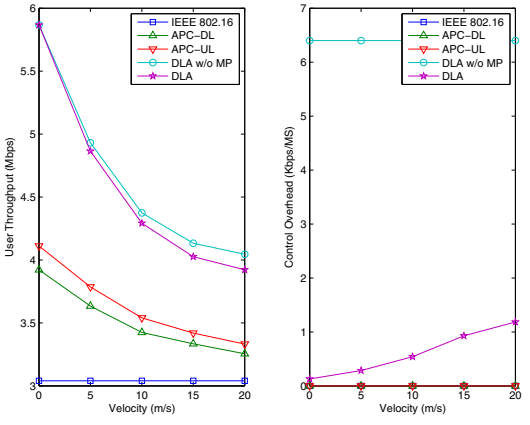


Fig. 5. Performance comparison of user throughput and control overhead versus mean velocity

Performance comparison with an increasing velocity ranging from 0 to 20 m/s is shown in Fig. 5, wherein 20 MSs are considered. As can be expected that the user throughput decreases as the mean velocity of MSs is increased for all the schemes excepted for the IEEE 802.16 approach, which can be attributed to the effect of direct communication. With larger velocity for the MSs, the variation of distances and channel conditions among MSs are changed frequently, which reduces the average life time of direct links. In such a case, the communication operation for the original direct communication pairs are changed from the direct manner to the indirect manner, which results in the decrement of user throughput. Due to the same reasons addressed for Fig. 4, the DLA w/o MP approach has the best performance of user throughput and worst control overhead among the compared schemes.

6 Concluding Remarks

In this paper, a scheduling algorithm for each pair of MSs that are expected to conduct direct communication is proposed. The interference region and feasible region for the pair of MSs are studied and calculated. Based on these two types of information as well as a motion predication mechanism, the direct link assignment (DLA) approach properly arranges the MSs to conduct direct communication in DL subframe or UL subframe without increasing additional interference for other communications. The efficiency of the proposed DLA approach is evaluated and compared via simulations. Simulation studies show that the DLA approach efficiently enhances the performance of user throughput in comparison with the original adaptive point-to-point (APC) approach and the conventional IEEE 802.16 scheme.

References

1. Abichar, Z., Peng, Y., Chang, J.M.: WiMAX: The emergence of wireless broadband. IEEE IT Prof. 8(4), 44–48 (2006)
2. IEEE Standard for Local and metropolitan area networks - Part 16: Air Interface for Broadband Wireless Access Systems, IEEE Standard 802.16-2009 (May 2009)
3. IEEE Standard for Information Technology-Telecommunications and information exchange between systems-Local and metropolitan area networks-Specific requirements-Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, Amendment 7: Extensions to Direct Link Setup (DLS), IEEE Standard 802.11z-2010 (October 2010)
4. Cordeiro, C., Abhyankar, S., Agrawal, D.P.: A dynamic slot assignment scheme for slave-to-slave and multicast-like communication in Bluetooth personal area networks. In: Proc. IEEE Global Telecommunications Conf. (GLOBECOM), San Francisco, CA, pp. 4127–4132 (2003)
5. Hsu, C.-H., Feng, K.-T.: Adaptive point-to-point communication approach for subscriber stations in broadband wireless networks. ACM Wireless Netw. 17(1), 69–86 (2011)
6. Gelb, A.: Applied Optimal Estimation. The MIT Press, Cambridge (1974)
7. Hong, X., Gerla, M., Pei, G., Chiang, C.-C.: A group mobility model for ad hoc wireless networks. In: Proc. ACM Int. Workshop Modeling, Analysis and Simulation of Wireless and Mobility Systems (MSWiM), Seattle, WA, pp. 53–60 (1999)
8. Liang, B., Haas, Z.J.: Predictive distance-based mobility management for multidimensional PCS networks 11(5), 718–732 (2003)
9. IEEE 802.16m Evaluation Methodology Document (EMD), IEEE C802.16m-08/004r5 (January 2009)
10. Shreedhar, M., Varghese, G.: Efficient fair queueing using deficit round robin. IEEE/ACM Trans. Netw. 4(3), 375–385 (1996)
11. Katevenis, M., Sidiropoulos, S., Courcoubetis, C.: Weighted round-robin cell multiplexing in a general-purpose atm switch chip. IEEE J. Sel. Areas Commun. 9(8), 1265–1279 (1991)

Dynamic Contention Resolution in Multiple-Access Channels

Dongxiao Yu[1], Qiang-Sheng Hua[2], Weiguo Dai[2], Yuexuan Wang[2,⋆], and Francis C.M. Lau[1]

[1] Department of Computer Science, The University of Hong Kong, Pokfulam, Hong Kong, P.R. China
[2] Institute for Interdisciplinary Information Sciences, Tsinghua University, Beijing, 100084, P.R. China
amywang@mail.tsinghua.edu.cn

Abstract. Contention resolution over a multiple-access channel can be modeled as a k-selection problem in wireless networks, where a subset k of n network nodes want to broadcast their messages over a shared channel. This paper studies a dynamic version of this problem, which assumes that k messages arrive at an arbitrary set of k nodes (contenders) asynchronously and the message arrival pattern is determined by an online adversary. Under this harsh and more practical assumption, we give a randomized distributed algorithm which can guarantee any contender deliver its message in $O(k + \log^2 n)$ rounds with high probability. Our proposed algorithm neither relies on collision detection, nor a global clock or any knowledge about the contenders, not even its size k. Furthermore, we do not assume the channel can provide any kind of feedback information, which makes our protocol work in simple channels, such as the channels used in wireless sensor networks.

Keywords: Contention Resolution, Multiple-Access Channel, Radio Networks, Distributed Algorithm, Randomized Algorithm.

1 Introduction

Contention resolution is a fundamental operation for both wired and wireless networks, which as a problem has been extensively studied for many years. For a multiple-access channel—a broadcast channel that allows a multitude of users to communicate with each other by sending messages onto the channel—contention resolution can be modeled as a k-selection problem in radio networks [1,5], where each node in a subset k of n network nodes wants to exclusively access a shared communication channel at least once. In terms of message transmissions, this means the k nodes want to broadcast their messages to the single-hop network with n nodes. Due to shared nature of the channel, if two or more users send a message simultaneously, then their messages interfere with each other, and the messages will not be transmitted successfully. The goal of a contention resolution

⋆ Corresponding author.

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 232–243, 2012.

protocol is to minimize the time for the nodes to access the shared channel. The most well-known example of contention resolution is ALOHA which was designed around 40 years ago. In ALOHA, when a sender encounters a collision, the sender will wait a random amount of time and send again. Due to its simplicity, this random back-off idea was later adopted by the current WiFi protocols (IEEE 802.11 protocols).

The static k-selection problem, where the k messages are ready at their respective nodes before the protocol starts, has been intensively studied; the state-of-the-art randomized static k-selection protocol, given in [5], needs $O(k+\log^2 n)$ rounds for all active nodes (nodes having messages to transmit) to successfully transmit their messages on the shared channel with high probability.[1] However, this protocol is uniform, i.e., all active nodes use the same transmission probability in the same communication step, which will not work when the message arrival pattern is arbitrary and a global clock is not available. The reason for this failure is that nodes do not know other nodes' statuses, not even the number of contenders. How to derive an efficient protocol for arbitrary message arrivals without a global clock is still open [5]. Furthermore, although previous work have considered the channel without collision detection—i.e., nodes can not distinguish between the case of no transmission and that of collision (multiple transmissions), they all assume that either the nodes can receive feedback information from the channel if the transmission is not successful, or a node by itself knows whether its transmission is successful, such that the nodes can decide whether it should quit the protocol after a transmission. This assumption is crucial for the correctness of these protocols. However, in some real wireless networks, such as wireless sensor networks, the channel can not provide any feedback. The protocol we introduce in this paper can work for these networks.

1.1 Related Work

The static k-selection problem has been studied since 1970s [2,10,17]. With the availability of collision detection, Martel [16] presented a randomized adaptive protocol with running time of $O(k+\log n)$ in expectation. In [14], it was shown that this protocol can be improved to $O(k+\log\log n)$ in expectation by making use of the expected $O(\log\log n)$ selection protocol in [20]. An $\Omega(\log\log n)$ expected time lower bound was also given in [20] for uniform selection protocols. Without collision detection, the state-of-the-art randomized protocol was presented in [5], which can solve the static k-selection problem in $O(k+\log^2 n)$ rounds with high probability. Given that k is a trivial lower bound, the protocol in [5] is asymptotically optimal for $k \in \Omega(\log^2 n)$. Furthermore, the result in [15] on the lower bound of the expected time needed to get the first message delivered implies an $\Omega(\log n)$ expected time lower bound for randomized k-selection protocols. In a recent paper [6], an $O(k)$ randomized protocol was proposed even without knowing n. However, the error probability of this protocol is $\frac{1}{k^c}$, rather

[1] We say an event succeeds with high probability if the error probability is at most $\frac{1}{n^c}$ for some constant $c > 0$.

than $\frac{1}{n^c}$. The performance of several kinds of randomized backoff k-selection protocols are analyzed in [1]. Apart from above work on static k-selection, the authors of a recent paper [11] showed that the $\Omega(k + \log n)$ lower bound for randomized protocols can be subverted when multiple channels are available. There are also several studies on the k-selection problem with dynamic packet arrivals, e.g., in stochastic model [7] and in the adversarial queuing model [1,3,14]. To the best of our knowledge, there have been no results on randomized protocols for k-selection with arbitrary message arrivals.

As for deterministic solutions for the static k-selection problem, the technique of tree algorithms, which models the protocol as a complete binary tree where the messages are placed at the leaves, has been used to produce adaptive protocols with running time $O(k \log(n/k))$ in [2,10,17]. All these protocols rely on collision detection. A lower bound of $\Omega(k \log_k n)$ is shown in [9] for this class of protocols. For oblivious algorithms, where the sequence of transmissions of a node does not depend on the received messages, by requiring prior knowledge on k and n, Komlòs and Greeberg [13] gave an $O(k \log(n/k))$ protocol without collision detection. The lower bound of $\Omega(k \log(n/k))$ for oblivious protocols was given in [4]. This lower bound also holds for adaptive algorithms without collision detection. If collision detection is unavailable, making use of the explicit selector given in [12], Kowalski [14] presented an oblivious deterministic protocol with $O(kpoly \log n)$ running time. For more details of the contention resolution algorithms in the past four decades, interested readers are referred to an excellent online survey maintained by Goldberg [8].

1.2 Our Result

In this paper, we present the first known randomized distributed protocol for dynamic contention resolution in a multiple-access channel with arbitrary message arrivals. In particular, we show that each node can deliver its message on the shared channel in $O(k + \log^2 n)$ rounds with high probability. When applying to the static scenario, the proposed protocol has the same asymptotical time bound as the state-of-the-art result in [5]. Based on the trivial $\Omega(k)$ lower bound, our protocol is asymptotically optimal if $k \in \Omega(\log^2 n)$. The channel considered in this work is even more practical than the simple multiple-access channel in [1] since we assume that the channel in this paper does not provide any feedback information to nodes. Furthermore, the number of contenders is also unknown to the nodes and there is not any collision detection mechanism being assumed. There are three challenges in this dynamic version of the k-selection problem: The first one is how to let any node know that its transmission has succeeded without feedback from the channel, such that the node knows when to quit the protocol and as a result the contention is reduced; the second one is how to ensure that newly activated nodes would not interfere with other nodes' processing; the third one is how to coordinate the nodes' transmission probabilities such that each node can quickly deliver its message for any arbitrary pattern of message arrivals and in the absence of a global clock. In our protocol, we meet these challenges by electing a leader. Specifically, the leader elected serves three

functions: first, it sends acknowledgement messages to inform nodes of their successful transmissions; second, it periodically transmits a trigger to notify newly activated nodes when they can start executing the protocol; third, it transmits controlling messages to adjust other nodes' transmission probabilities according to the transmission situation of the shared channel.

2 Preliminaries

Communication Model (The Multiple-Access Channel). We consider a single-hop radio network consisting of n nodes, in which each node is potentially reachable from any other node in a communication step. Communication is done in synchronized rounds, which means that all the nodes' clocks tick simultaneously at the same rate. However, we do not assume the existence of a global clock. So in a round, clock values may be different among the nodes. A node sets its clock value to 1 after being activated, and increases it at every round. If exactly one node is transmitting on the shared channel, all nodes receive the message transmitted by this node at the end of the round. A collision occurs when multiple messages are transmitted concurrently, which means that none of these messages can be successfully received by any node. Collision detection is not assumed, i.e., nodes can not tell apart the case of no transmission and that of collision (more than one node transmit concurrently). We further assume that the shared communication channel does not provide any feedback information in any case.

An external mechanism is assumed that generates messages and assigns them to nodes with the purpose of broadcasting them on the channel. For each node, there are three status: *idle*, *active* and *passive*. Before being assigned a message, a node is *idle* and does nothing. It becomes *active* if it is assigned a message and is ready to transmit on the shared channel. After successfully transmitting its message, the node switches its status to *passive*. In a round, active nodes can either transmit or listen, while passive nodes can only listen on the channel. We say that a node is activated in round i if it is assigned a message in round i. Each node has a unique but arbitrary ID, and it knows its own ID and the parameter n. Nodes have no other prior information about the network. Initially, a node does not know any other nodes' statuses, nor the number of nodes that compete with it for the channel.

Problem Definition. Suppose that a subset k of the set of n network nodes are activated by message arrivals. The k-selection problem is to make each of the k nodes deliver its message on the shared communication channel as quickly as possible.

Dynamic Setting. In this work, we consider a dynamic version of the k-selection problem. Messages may arrive at the nodes asynchronously and the arrival pattern is arbitrary, even in the worst case. In particular, we assume that the message arrivals are controlled by an on-line adversary. The adversary knows the protocol, but does not know the future random bits. Obviously, such an

adversary is much stronger than the usually assumed oblivious adversary (which decides the message arrivals off-line).

Complexity Measure. We define the efficiency measurement of dynamic k-selection protocols as follows. We assume that comparing to the communication time consumed, the computation time cost is negligible. So we only care about the time efficiency in terms of communication rounds. Formally, we define the process latency of a node v as the length of the period between its activation time and the completion time (when it completes the message transmission and becomes passive). The time complexity of a k-selection protocol is the maximum value of all nodes' process latencies over any message arrival pattern. When messages arrive at nodes simultaneously, the above defined time complexity is just the same as that for the static k-selection protocols.

Finally, we give some inequalities and a Chernoff bound as follows which will be used in the protocol analysis.

Lemma 1. *([19]) Given a set of probabilities $p_1, \ldots, p_n$ with $\forall i : p_i \in [0, \frac{1}{2}]$, the following inequalities hold:*

$$(1/4)^{\sum_{k=1}^{n} p_k} \leq \prod_{k=1}^{n} (1 - p_k) \leq (1/e)^{\sum_{k=1}^{n} p_k}. \tag{1}$$

Lemma 2. *([19]) For all n, t, with $n \geq 1$ and $|t| \leq n$, it holds that:*

$$e^t (1 - \frac{t^2}{n}) \leq (1 + t/n)^n \leq e^t. \tag{2}$$

Lemma 3. *(Chernoff Bound) Suppose that X is the sum of n independent $\{0, 1\}$-random variables X_i's such that for each i, $Pr(X_i = 1) = p$. Let $\mu = E[X]$. Then for $0 \leq \epsilon \leq 1$,*

$$Pr(X \leq (1 - \epsilon)\mu) \leq e^{-\frac{\epsilon^2 \mu}{2}} \tag{3}$$

3 Algorithm

3.1 Algorithm Description

In this section, we propose our randomized contention resolution algorithm. In the algorithm, due to the assumption that the channel can not provide any feedback, a leader is elected to be responsible for acknowledging successful transmissions. By transmitting controlling messages, the leader also takes the responsibilities of adjusting other nodes' transmission probabilities and informing newly activated nodes when to start executing the algorithm. In particular, the leader and active non-leaders iteratively execute a 3-round scheme which is shown in Algorithm 1. The first round is for active non-leaders to transmit their messages. The second and the third rounds are used for leader's transmissions. In the first

round, each active non-leader transmits its message with a specified transmission probability. If only one node u sends in the first round, the leader v can successfully receive the message. Then in the second round, v sends an acknowledgement message to inform u of the successful transmission. After receiving the acknowledgement message, u adjusts its status as passive and quit the algorithm. To deal with arbitrary arrivals of messages, in the third round, the leader also transmits a controlling message. If v's clock value is not $r_l + i \cdot 3 \cdot 2^{\alpha+6+2^{-\alpha-1}} \log n$ for some integer $i > 0$, where r_l is the last round before v starts executing the 3-round scheme and α is a constant given in Algorithm 1,[2] the controlling message transmitted by v would only carry a trigger which is to inform newly activated nodes to start executing the 3-round scheme from the next round. Otherwise, besides the trigger, the controlling message also contains information on how to adjust the transmission probabilities of other active nodes. Specifically, if v transmitted less than $8 \log n$ *Ack* messages in the past $3 \cdot 2^{\alpha+6+2^{-\alpha-1}} \log n$ rounds, which means that nodes' transmission probabilities are not large enough to get many successful transmissions, v makes all active nodes double their transmission probabilities. Otherwise, it makes all nodes halve the transmission probability. The leader is elected through carrying out the MIS (maximal independent set) algorithm in [19], which can correctly compute a maximal independent set[3] in $O(\log^2 n)$ rounds with high probability. A newly activated node will first wait for at most three rounds. If it did not receive the trigger, which means that the leader has not been elected, then this newly activated node would start executing the MIS algorithm to compete for becoming a leader. Otherwise, it iteratively executes the 3-round scheme after receiving the trigger.

3.2 Analysis

In this section, we prove the correctness and efficiency of the proposed algorithm. Specifically, we show that for any node u, with high probability, it can successfully transmit its message on the shared channel after being activated for at most $O(k + \log^2 n)$ rounds. First, we state the correctness and the efficiency of the MIS algorithm in the following lemma which is proved in [19].

Lemma 4. *([19]) After executing the MIS algorithm for $O(\log^2 n)$ timeslots, a maximal independent set can be correctly computed with probability at least $1 - O(n^{-1})$.*

From the above lemma, a leader can be correctly elected after $O(\log^2 n)$ rounds with high probability. In the following, we assume that the leader is correctly computed; the error probability will be considered in the proof of the main theorem. Let's denote the elected leader as v. Next we give a lemma which

[2] Here we only give a value for α such that the proposed algorithm is correct and has the stated asymptotically running time bound with high probability. Since a different value of α only affects the time complexity of our algorithm by a constant factor, so we do not optimize the value we choose for α.

[3] For single hop networks, the MIS contains only one node.

Algorithm 1. 3-Round Scheme

Initially, $p_u = \frac{2^{-\alpha-1}}{n}$; $\alpha = 1$
3-round scheme for the leader v
1: listen
2: **if** v received a message from a non-leader u
then transmit Ack_u
end if
3: transmit a controlling message
3-round scheme for an active non-leader u
4: **if** u has a message to transmit
then transmit the message with probability p_u
end if
5: listen
if u received Ack_u
then quit the execution of the algorithm
end if
6: listen
if the received controlling message contains information on how to adjust the transmission probability
then adjust the transmission probability accordingly
end if

states that in any round the sum of transmission probabilities of all active nodes is bounded by a constant.

Lemma 5. *Assume that the leader is correctly elected. In any round during the execution of the algorithm, with probability* $1 - \frac{1}{n}$*, the sum of transmission probabilities of active nodes is at most* $2^{-\alpha}$*.*

Proof. Assume that r is the first round that the sum of transmission probabilities of active nodes exceeds $2^{-\alpha}$. Denote r_a as the last round until r in which the leader v transmits a controlling message to adjust other active nodes' transmission probabilities. By the algorithm, the leader v transmits a controlling message to adjust other active nodes' transmission probabilities every $3 \cdot 2^{\alpha+6+2^{-\alpha-1}} \log n$ rounds. So r_a must be in the interval $(r - 3 \cdot 2^{\alpha+6+2^{-\alpha-1}} \log n, r]$. Since r is the first violating round, in any round before r, for the sum of transmission probabilities of active nodes, we have $\sum_u p_u \leq 2^{-\alpha}$. Furthermore, by the algorithm and the definition of r_a, for nodes that are activated after round $r_a - 3$, in any round until r, the sum of transmission probabilities of these nodes is at most $\frac{2^{-\alpha-1}}{n} \times n = 2^{-\alpha-1}$. Then in any round during the interval $I = (r_a - 3 \cdot 2^{\alpha+6+2^{-\alpha-1}} \log n, r_a]$, for the sum of transmission probabilities of nodes that have been activated before $r_a - 2$, we have $\sum_u p_u \geq 2^{-\alpha-2}$, since each such node can only double their transmission probabilities once in round r_a. Before any violating round, there must be such an interval I. Next we show that during I, with probability $1 - n^{-2}$, v transmits at least $8 \log n$ *Ack* messages. Then by the algorithm, v makes all active nodes halve their transmission probabilities in round r_a, which leads to a contradiction.

Claim. During the interval I, with probability $1 - n^{-2}$, the leader v transmits at least $8 \log n$ *Ack* messages.

Proof. For a round r^*, denote the set of active nodes as A_{r^*}. During I, in the first round r_1 of each execution of the 3-round scheme, the probability P_{one} that there is only one node transmitting is

$$\begin{aligned} P_{one} &= \sum_{u \in A_{r_1}} p_u \prod_{w \in A_{r_1} \setminus \{u\}} (1 - p_w) \\ &\geq \sum_{u \in A_{r_1}} p_u \cdot \left(\frac{1}{4}\right)^{\sum_{w \in A_{r_1} \setminus \{u\}} p_w} \\ &\geq \sum_{u \in A_{r_1}} p_u \cdot \left(\frac{1}{4}\right)^{\sum_{w \in A_{r_1}} p_w} \end{aligned} \tag{4}$$

The second inequality is by Lemma 1. Note that the function $f(x) = x\left(\frac{1}{4}\right)^x$ is monotone increasing in the range $[2^{-\alpha-2}, 2^{-\alpha}]$. So we have

$$\begin{aligned} P_{one} &\geq \sum_{u \in A_{r_1}} p_u \cdot \left(\frac{1}{4}\right)^{\sum_{w \in A_{r_1}} p_w} \\ &\geq 2^{-\alpha-2} \cdot \left(\frac{1}{4}\right)^{2^{-\alpha-2}} \end{aligned} \tag{5}$$

By the algorithm and the above equation, during the interval I, in expectation, there are at least $16 \log n$ active non-leader nodes successfully transmitting their messages on the shared channel, since $\frac{1}{3}$ of the rounds in the interval I are used for non-leaders' transmissions. Then using the Chernoff bound in Lemma 3, the probability that v transmits less than $8 \log n$ *Ack* messages during I is at most $e^{-\frac{1}{8} \cdot 16 \log n} = n^{-2}$. □

By the above claim and the algorithm, with probability $1 - n^{-2}$, v makes active nodes halve their transmission probability in round r_a. Thus for all nodes which have been activated before $r_a - 2$, with probability $1 - n^{-2}$, the sum of transmission probabilities is at most $\sum_u p_u \leq \frac{1}{2} \times 2^{-\alpha} = 2^{-\alpha-1}$ in any round during the interval $I' = [r_a, r_a + 3 \cdot 2^{\alpha+6+2^{-\alpha-1}} \log n)$, since v will not transmit another controlling message to adjust the nodes' transmission probabilities during I'. Obviously, r is in I'. Then combining the fact that the sum of transmission probabilities of all nodes that are activated after the round $r_a - 3$ is at most $2^{-\alpha-1}$ in round r, we have $\sum_{u \in A_r} p_u \leq 2^{-\alpha-1} + 2^{-\alpha-1} = 2^{-\alpha}$. So with probability $1 - n^{-2}$, r is not the first violating round.

To complete the proof, we still need to bound the number of potential violating rounds. From the above argument, before each potential violating round, with

probability $1 - n^{-2}$, there are $\Omega(\log n)$ successful transmissions for active non-leader nodes during the corresponding interval I. So there are at most $O(\frac{k}{\log n})$ potential violating rounds and thus with probability at least $1 - n^{-1}$, none of these rounds are the first violating round. This completes the proof. □

Theorem 1. *For any node u, with probability $1 - O(n^{-1})$, it can successfully transmit its message on the shared channel after carrying out the algorithm for $O(k + \log^2 n)$ rounds.*

Proof. By the algorithm, if u is activated before the leader is elected, it needs to take part in the MIS algorithm. By Lemma 4, this process needs at most $O(\log^2 n)$ rounds with probability $1 - O(n^{-1})$. After that, u starts iteratively executing the 3-round scheme. If u is activated after the leader election process, it starts the 3-round scheme execution after waiting for at most three rounds. Next we bound the number of rounds needed for u in executing the 3-round scheme before receiving the Ack_u message from the leader v. From then on, we assume that the leader v is correctly computed, which means that there is only one leader. And we assume that in any round, the sum of transmission probabilities of active nodes is at most $2^{-\alpha}$. The error probability will be considered at last.

For a round, we call it successful if there is only one non-leader node transmitting in it. By the algorithm, when u executes the 3-round scheme, u adjusts its transmission probabilities every $3 \cdot 2^{\alpha+6+2^{-\alpha-1}}$ rounds. So after starting executing the 3-round scheme, for every $3 \cdot 2^{\alpha+6+2^{-\alpha-1}}$ rounds, either a constant of these rounds are successful, which makes u halves its transmission probability, or u doubles its transmission probability once. So after at most $2 \cdot \frac{k-1}{8 \log n} \cdot 3 \cdot 2^{\alpha+6+2^{-\alpha-1}} \log n + 3 \cdot 2^{\alpha+6+2^{-\alpha-1}} \log^2 n$ rounds, u has a constant transmission probability $2^{-\alpha-1}$, since there are at most $k - 1$ contenders when u is active. Next we show that u can successfully transmit its message in the subsequent $3 \cdot 2^{\alpha+6+2^{-\alpha-1}} \log n$ rounds with probability $1 - n^{-2}$. Denote P_{only} as the probability that u is the only transmitting node in a round r. Denote A_r as the set of active nodes in round r. Then we have

$$\begin{aligned} P_{only} &= p_u \prod_{w \in A_r \setminus \{u\}} (1 - p_w) \\ &\geq p_u \cdot \left(\frac{1}{4}\right)^{\sum_{w \in A_r \setminus \{u\}} p_w} \\ &\geq 2^{-\alpha-1} \cdot \left(\frac{1}{4}\right)^{\sum_{w \in A_r} p_w} \end{aligned} \tag{6}$$

By Lemma 5, $\sum_{w \in A_r} p_w \leq 2^{-\alpha}$. Then

$$P_{only} \geq 2^{-\alpha-1} \cdot \left(\frac{1}{4}\right)^{2^{-\alpha}} \tag{7}$$

By the algorithm, u transmits in $\frac{1}{3}$ of the $3 \cdot 2^{\alpha+6+2^{-\alpha-1}} \log n$ rounds. Thus the probability P_{no} that all of these $2^{\alpha+6+2^{-\alpha-1}} \log n$ transmissions are unsuccessful is at most

$$\begin{aligned} P_{no} &\leq (1 - 2^{-\alpha-1} \cdot \left(\frac{1}{4}\right)^{2^{-\alpha}})^{2^{\alpha+6+2^{-\alpha-1}} \log n} \\ &\leq e^{-2^{-\alpha-1} \cdot \left(\frac{1}{4}\right)^{2^{-\alpha}} \cdot 2^{\alpha+6+2^{-\alpha-1}} \log n} \\ &\leq n^{-2} \end{aligned} \tag{8}$$

The second inequality is by Lemma 2. So u will successfully transmit its message in the subsequent $3 \cdot 2^{\alpha+6+2^{-\alpha-1}}$ rounds with probability $1-n^{-2}$. This means that after executing the 3-round scheme for $O(k + \log^2 n)$ rounds, with probability $1 - n^{-2}$, u can successfully transmits its message on the shared channel. This claim is true for any node with probability $1 - n^{-1}$.

Finally, we combine everything together. Based on the above argument, we know that for any node u, with probability $1 - O(n^{-1})$, it takes at most $O(k + \log^2 n)$ rounds in executing the MIS algorithm and the 3-round scheme. Furthermore, note that the above argument is under the assumptions that the leader is correctly computed and the sum of transmission probabilities of active nodes is upper bounded by $2^{-\alpha}$ in any round. By Lemma 4 and Lemma 5, these two assumptions are true with probability $1 - O(n^{-1})$. Thus any node u can successfully transmit its message after being activated for $O(k + \log^2 n)$ rounds with probability $1 - O(n^{-1})$, which completes the proof.

4 Conclusion

In this paper, we solve the dynamic contention resolution problem in the multiple-access channel, also called the dynamic k-selection problem, where the message arrival pattern is determined by an online adversary. To the best of our knowledge, our protocol is the first one considering an arbitrary pattern of message arrivals. We show that the proposed protocol can make each node successfully deliver its message on the shared channel in $O(k + \log^2 n)$ rounds with high probability, which is optimal when $k \in \Omega(\log^2 n)$. Our protocol neither relies on collision detection, nor on a global clock or any knowledge about the number of contenders k. In addition, we do not assume the channel can provide any feedback information which is commonly used in existing contention resolution protocols. Thus our contention resolution protocol can be applied to a variety of wireless networks such as wireless sensor networks without such a function. Interesting future work include how to extend our result to wireless networks without even an estimation of the number of network nodes n. Furthermore, it is also meaningful to analyze the performance of our protocol in the adversarial queueing model [1].

Acknowledgements. The authors thank the anonymous reviewers for their helpful comments. This work was supported in part by the National Basic

Research Program of China Grant 2011CBA00300, 2011CBA00302, the National Natural Science Foundation of China Grant 61103186, 61073174, 61033001, 61061130540, the Hi-Tech research and Development Program of China Grant 2006AA10Z216, and Hong Kong RGC-GRF grants 714009E and 714311.

References

1. Bender, M.A., Farach-Colton, M., He, S., Kuszmaul, B.C., Leiserson, C.E.: Adversarial contention resolution for simple channels. In: Proc. of the 17th Ann. ACM Symp. on Parallel Algorithms and Architectures, SPAA, pp. 325–332 (2005)
2. Capetanakis, J.: Tree algorithms for packet broadcast channels. IEEE Trans. Inf. Theory IT-25(5), 505–515 (1979)
3. Chlebus, B.S., Kowalski, D.R., Rokicki, M.A.: Adversarial queuing on the multiple-access channel. In: Proc. of the 25th ACM Symposium on Principles of Distributed Computing, PODC, pp. 92–101 (2006)
4. Clementi, A., Monti, A., Silvestri, R.: Selective families, superimposed codes, and broadcasting on unknown radio networks. In: Proc. of the 12th Ann. ACM-SIAM Symp. on Discrete Algorithms, SODA, pp. 709–718 (2001)
5. Fernández Anta, A., Mosteiro, M.A.: Contention Resolution in Multiple-Access Channels: k-Selection in Radio Networks. In: Thai, M.T., Sahni, S. (eds.) COCOON 2010. LNCS, vol. 6196, pp. 378–388. Springer, Heidelberg (2010)
6. Fernández Anta, A., Mosteiro, M.A., Muñoz, J.R.: Unbounded Contention Resolution in Multiple-Access Channels. In: Peleg, D. (ed.) Distributed Computing. LNCS, vol. 6950, pp. 225–236. Springer, Heidelberg (2011)
7. Goldberg, L.A., Jerrum, M., Kannan, S., Paterson, M.: A bound on the capacity of backoff and acknowledgment-based protocols. SIAM Journal on Computing 33, 313–331 (2004)
8. Goldberg, L.A.: Design and analysis of contention-resolution protocols, EPSRC Research Grant GR/L60982, http://www.csc.liv.ac.uk/~leslie/contention.html (last modified, October 2006)
9. Greenberg, A., Winograd, S.: A lower bound on the time needed in the worst case to resolve conflicts deterministically in multiple access channels. Journal of the ACM 32, 589–596 (1985)
10. Hayes, J.F.: An adaptive technique for local distribution. IEEE Trans. Comm. COM-26, 1178–1186 (1978)
11. Holzer, S., Pignolet, Y.A., Smula, J., Wattenhofer, R.: Time-Optimal Information Exchange on Multiple Channels. In: Proc. of the 7th ACM SIGACT/SIGMOBILE International Workshop on Foundations of Mobile Computing, FOMC, pp. 69–76 (2011)
12. Indyk, P.: Explicit constructions of selectors and related combinatorial structures, with applications. In: Proc. of the 13th Ann. ACM-SIAM Symp. on Discrete Algorithms, SODA, pp. 697–704 (2002)
13. Komlòs, J., Greenberg, A.: An asymptotically nonadaptive algorithm for conflict resolution in multiple-access channels. IEEE Trans. Inf. Theory 31, 303–306 (1985)
14. Kowalski, D.R.: On selection problem in radio networks. In: Proc. of 24th Ann. ACM Symp. on Principles of Distributed Computing, SPAA, pp. 158–166 (2005)
15. Kushilevitz, E., Mansour, Y.: An $(D \log(N/D))$ lower bound for broadcast in radio networks. SIAM Journal on Computing 27(3), 702–712 (1998)

16. Martel, C.U.: Maximum finding on a multiple access broadcast network. Inf. Process. Lett. 52, 7–13 (1994)
17. Mikhailov, V., Tsybakov, B.S.: Free synchronous packet access in a broadcast channel with feedback. Problemy Peredachi Inform. 14(4), 32–59 (1978)
18. Mitzenmacher, M., Upfal, E.: Probability and Computing: Randomized Algorithms and Probabilistic Analysis. Cambridge University Press (2005)
19. Moscibroda, T., Wattenhofer, R.: Maximal independent sets in radio networks. In: Proc. of 24th Annual ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing, PODC, pp. 148–157 (2005)
20. Willard, D.E.: Log-logarithmic selection resolution protocols in a multiple access channel. SIAM Journal on Computing 15, 468–477 (1986)

An Assessment of Community Finding Algorithms for Community-Based Message Routing in DTNs

Matthew Stabeler, Conrad Lee, and Pádraig Cunningham

School of Computer Science and Informatics
University College Dublin, Dublin 4, Ireland
matthew.stabeler@ucd.ie

Abstract. Previous work has demonstrated that community-finding algorithms can provide useful information for routing algorithms in delay tolerant networks. In this work we investigate which community finding algorithm most effectively informs this routing task. While early community finding algorithms partitioned networks into flat disjoint communities, more recent methods return community structures that can be overlapping and hierarchical. Given this diversity, it seems reasonable to expect that some methods will be better suited to message routing than others. In this paper, we evaluate a number of community finding strategies and find that Link Clustering, which returns overlapping hierarchical clusters, is very effective. We also find that InfoMap performs well – this is somewhat surprising given that InfoMap returns a flat partition of the network, however this may be because the optimization that drives InfoMap is based on flow.

Keywords: Delay Tolerant Networking, Community-Based Routing.

1 Introduction

The study of message routing in *challenged* networks [18,5,11,10] has evolved to cover a variety of situations where traditional point to point networks may be hard, expensive, or otherwise infeasible to create or maintain. The term Delay Tolerant Networks (DTNs) covers many types of networks [2], such as Inter-Vehicular communications, Mobile/Fixed Sensor networks (Animal, Human, Infrastructure etc.), satellite communications, battlefield communication networks, or any other system of interacting devices, objects, beings or things. A DTN protocol is concerned with the delivery of messages between nodes within these dynamic networks in a way that is tolerant to intermittent connections, disconnections and failures. Such systems are often characterised by a sparse network of connections between individuals that change over time.

In previous work, it was demonstrated that community structure can be useful for routing in PSNs [6]. The work introduced a particular routing algorithm, BUBBLE Rap, that exploited the structure returned by a particular community finding algorithm. However, it left two questions unanswered: (1) could one

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 244–256, 2012.

improve the results by modifying the routing algorithm, and (2) which community finding algorithm should be used by the routing algorithm? In this paper, we present the results of experiments designed to address these questions. The contribution of this paper is the comparison of hierarchical, non-hierarchical, overlapping and non-overlapping community-based message routing using 5 community finding algorithms, on networks derived from multiple real-world datasets. Our BubbleH algorithm is described in more details in Section 3 and in [16].

The paper is organized as follows. In the next section we provide an overview of existing research on community-based message routing. Next, we compare the performance of two routing algorithms, BubbleH and BUBBLE Rap. We find that our BubbleH algorithm performs better than BUBBLE Rap when there is hierarchical community structure. Next, we employ various community finding algorithms and try to identify which is most useful for routing purposes. We try out several types of community finding algorithms – including those which return disjoint, overlapping or hierarchical community information. For all experiments, we use several datasets.

2 Message Routing

Research on DTNs has produced many strategies for message routing. Flooding based approaches flood the network with copies of messages between any nodes that meet. Epidemic-like protocols behave similarly, for example Epidemic Routing [17] which transmits messages to other nodes with some probability, but limits messages by hop count to reduce overhead. These are perhaps the most effective, but the large number of message copies generated and sent mean a large overhead in message transmission between nodes. More conservative approaches include Spray and Wait [15], where a limited number of messages are distributed before a phase where nodes keep the messages until meeting the destination. Other approaches involve probabilistic, or opportunistic mechanisms to predict future interactions, such as PROPHET [10] and Context-Aware Adaptive Routing (CAR) [11] which use knowledge about previous contacts to inform predictions about co-locations which are used to decide next-hop routes. Zhang provides a survey of schemes for routing in intermittently connected mobile ad-hoc networks [18], and groups each of the approaches into two distinct categories; deterministic approaches, where the network structure is known over time, and non-deterministic, those where the network structure is not known. In this paper, we are considering deterministic approaches, as the algorithms we are testing assume that the network structure is available to them for community detection. We make the assumption that network structure is available, so that we can concentrate on the underlying operation of the routing algorithm, in a real-world scenario, it is likely that network structure is only known via means of incremental updates as nodes meet each other, this is beyond the scope of this paper, but is of interest for further work.

The effectiveness of a DTN message routing algorithm can be encapsulated in three metrics. Most often the goal is to have high *delivery ratio*, a low overhead transmission *cost* and low *delivery latency* [2]. When comparing the ability of a routing scheme to deliver messages effectively, the relative importance of each metric depends upon the application involved. In the case where a message is being sent between people, for example in a distributed version of the current SMS message system, the expectation may be that the delivery ratio is very high and most important. Also important, is that it should have a low latency, as the message should arrive promptly. Less important is that the cost is as low as possible, but not so low that the message never arrives.

In other situations, delivery ratio and latency may not be as important. Where the content of a message is, for example, a personal status broadcast system, perhaps an infrastructure-less, distributed version of Twitter, then delivery ratio may not be as important. We may only want to have the most up to date status updates – old, out of date messages may be discarded, so *cost* may be our most important metric. For clarity, here we define the above metrics as the following:

Delivery Ratio. (average) is the number of *delivered* messages over the total number of messages sent.

Delivery Latency. (average) is the total time to deliver each *delivered* message. over the total number of *delivered* messages.

Cost. (average) is the total number of transmissions of messages (delivered and undelivered), over the total number of messages sent.

3 Communities for Routing

Here we review two routing algorithms that base their routing decisions on community structure: the well-known BUBBLE Rap algorithm [6] and our own BubbleH [16] algorithm, which extends the basic BUBBLE Rap idea to exploit community hierarchy and the comparative size of communities. Any community finding algorithm could be used by these routing algorithms; which we will discuss in the next section.In this way, we hope to be able compare the benefits of each community finding algorithm in terms of each message routing algorithm.

Each of these routing algorithms use the notion of community structure to inform routing decisions. BUBBLE Rap uses Palla et al's Clique Percolation Method [12] (referred to as k-CLIQUE), to generate overlapping community structure from a graph formed from contacts between nodes. BUBBLE Rap uses the notion of local and global node ranking to make decisions about message passing. Node rank is based on the betweenness centrality of each node in the global network (global rank) and each community the node belongs to (local rank). As with the BUBBLE Rap algorithm, the BubbleH algorithm calculates local rankings based on betweenness centrality, however, it does not use global rank. Instead, it uses the community hierarchy to drive the mobility of messages within the network. When a node encounters another, it considers whether to pass the message on based on how *close* the other node is in the network hierarchy to the destination node. BubbleH has been shown to improve delivery

ratio and at least match delivery cost of BUBBLE Rap for the MIT Reality Mining dataset using the H-GCE community finding algorithm [16]. Here we intend to evaluate its performance against multiple datasets and multiple community finding algorithms.

On node n connection to encountered node p
for all messages m held by n for destination d **do**
  if $p == d$ **then**
    $p \leftarrow m$
  else if $|BC(p,d)| < |BC(n,d)|$ **then**
    $p \leftarrow m$
  else if $BC(p,d) == BC(n,d)$ and
  $LocalRank(p) > LocalRank(n)$ **then**
    $p \leftarrow m$
  else
    n keeps m
  end if
end for

Fig. 1. BubbleH Algorithm, where $BC(p,d)$ is the set of all nodes which represent the smallest community, or *Bridging Community* containing both node p and node d

Figure 1 shows the BubbleH algorithm, the *Bridging Community* ($BC(p,d)$), is the smallest community that contains the node in question, and the recipient, or destination node. To find the *Bridging Community* between two nodes, BubbleH finds all of the communities that the nodes share, and picks the shared community that has the lowest member count. In the case where there are multiple candidate communities (when the community finding algorithm allows overlap), the last candidate community in the list is chosen.

To illustrate the concept, Figure 2 shows a simplified overlapping community structure, and its associated hierarchical structure. If we imagine that node F has a message destined for node B, the smallest community containing both F and B is $c1$, this is the *bridging community* for F and B. On encountering another node,

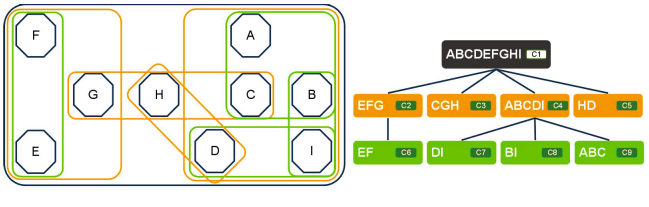


Fig. 2. Simplified example of an overlapping hierarchical community structure. The grouping of nodes in the left section relates to the community hierarchy shown in the right section. Communities are identified as $C1$ to $C9$, nodes are identified as A to G. For clarity, edges between nodes are not shown.

F must consider whether the encountered node has a better *bridging community* than itself. For example, when meeting node C, whose *bridging community* with destination node B is $c9$, it will find that $c9$ has less members than $c1$, and pass the message to C.

4 Community Finding

As described above, any community finding method can be plugged into the routing algorithm. The community finding algorithms that we evaluate here differ fundamentally: whereas InfoMap produces a non-overlapping partition of nodes, the others allow communities to overlap. While Ahn et al.'s link partitioner and Hierarchical Greedy Clique Expansion (H-GCE) create a dendrogram of hierarchical structure, the others are based on a flatter conception of community structure. We now provide a brief description of InfoMap, the Clique Percolation Method, the Link Clustering algorithm, and of H-GCE.

InfoMap. [13] The basic idea behind InfoMap is that a random walker in a network will tend to get stuck in communities, spending relatively large amounts of time within communities and small amounts of time passing between them. This regularity enables compression of a string representation of the random walk sequence: by assigning a unique namespace to each community, short node IDs can be recycled between communities. The extra cost (in terms of information) for the namespace scheme is that each time the random walker switches namespaces, special exit and entry codes must be inserted into the sequence.

The way the random walker moves around communities can be likened to the flow of messages between individuals, there may be a strong current of messages between well connected individuals within communities, and less so to other communities.

Rosvall and Bergstrom show that given a partition, one can efficiently calculate how much compression can be obtained due to community structure with the "map equation." The map equation can then be maximized by any search algorithm. For an explanation of InfoMap, see `www.mapequation.org`

k-CLIQUE (Clique Percolation Method). [12] In 2005 Palla *et al.* introduced their clique percolation method for community finding called CFinder. Clique percolation entails finding k-clique communities which are the union of cliques of size k respecting the constraint that these cliques are adjacent in the sense that they share $k-1$ nodes.

In their paper describing BUBBLE Rap [6], Hui et al use this method for finding communities, which they refer to as the k-CLIQUE method. Hui et. al. remove edges of the original node graph before clustering based on a threshold of connected time for each edge. We also implement this threshold version and tune the threshold weight for each dataset as described in section 4.1.

Link Clustering. [1] With their work on link partitioning Ahn *et al.* take an edge-centric view of communities, defining a community as a set of edges rather than a set of nodes. Their aim is to partition all edges in the graph into non-overlapping communities. Although under this scheme each edge must belong to exactly one community, nodes can belong to multiple communities. The objective function they propose is simply the normalized edge density of each community, where each community is weighted by the fraction of edges present in it. In their algorithm they use single-linkage hierarchical clustering to create a dendrogram, and cut the dendrogram where the objective function is maximized. We adapted the reference implementation of LinkClustering [1] to discard edges under a given threshold, in the same way as with k-CLIQUE.

Random. The random community finding algorithm assigns nodes to communities at random, ensuring that communities do not overlap by more than 50%.

H-GCE. Agglomerative community finding algorithms start with a set of seeds (where a seed is typically a node or an edge), and expands these seeds by adding nodes to them such that a local fitness function is greedily optimized. In [9], Lee et al. showed that cliques provide good seeds for these algorithms, especially in networks where nodes belong to several communities.

In [16] we proposed H-GCE, an agglomerative community finding algorithm that uses cliques as seeds and additionally uses random perturbations to identify which communities are stable. The idea is that "significant" communities will be recoverable even if some noise is added to the graph; this idea has been previously developed in [7,14]. This and other approaches to measuring the significance of communities are outlined in Fortunato's review of community finding methods [4]. For the details of H-GCE, see [16].

4.1 Community Finding Data

To build a network for the community finding analysis we add an undirected edge between individual if they met during the simulation time period and we weight the edge by total amount of time the nodes were connected to each other. Each algorithm deals with edge weights differently; for KCLIQUE, LinkClustering and HGCE, we specify a threshold cut value for the network edges. InfoMap and Random do not take a threshold parameter.

Threshold Selection. We use the the mean, median, 80th percentile and 20th percentile of the connected time for all edges as threshold values. This gives us a robust way to get a good approximation for threshold values. However, this may not be the optimum mechanism for calculating threshold values as it is based on our experience in early experiments. In our results, we show the best results for each metric (delivery ratio, latency, cost etc.), for each routing algorithm to give them a fair chance against each other.

[1] http://barabasilab.neu.edu/projects/linkcommunities/

5 Message Routing Using BubbleH

In our previous work [16], we showed that BubbleH performs well against BUBBLE Rap when using a hierarchical community finding algorithm. In the experiment described here, we use the same Bluetooth proximity traces from the MIT Reality project [3] (described in section 6) to drive contact events within the simulator. The MIT Reality dataset has the most number of connections between between Oct 2004 and Jan 2005, so we chose the period between Nov 2004 and Dec 2004 for community detection and testing, which we refer to as *MIT-NOV*.

Figure 3 shows a visualisation of the MIT-NOV dataset, where edges are weighted by connected time, and nodes are clustered using the InfoMap algorithm (with clusters indicated by node colour).

5.1 BubbleH versus BubbleRAP Evaluation

This simulation tests message routes between all pairs of nodes in the dataset. In the evaluation we consider three different criteria, the relative importance of these criteria may depend on specific circumstances but a reasonable assessment of the order of importance is as follows:

1. **Delivery Ratio:** The proportion of messages successfully delivered.
   Delivery Ratio = total number of delivered messages / total number of messages sent

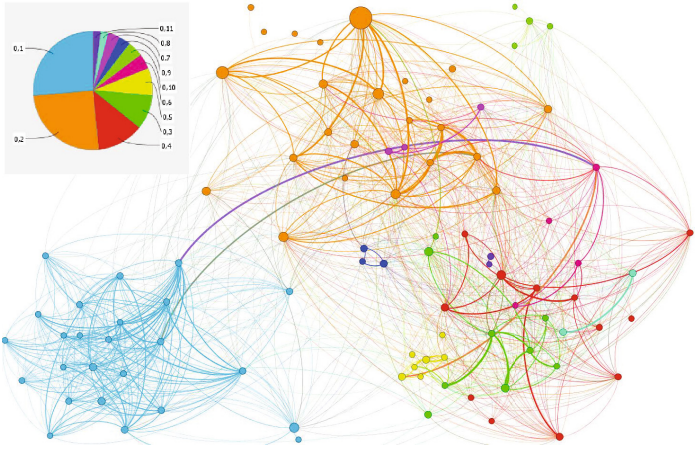


Fig. 3. Visualisation of the *MIT-NOV* dataset, node and edge colour shows the assigned cluster using the InfoMap algorithm. Node size represents betweenness centrality. Edge thickness represents total connected time between nodes. Also shown inset, is the proportion of nodes included in each community. For clarity, this graph does not show edges where the total connected time is less than the median for all edges.

2. **Latency:** This measures the amount of time it takes to deliver messages. *Latency = total time to deliver messages / total delivered Messages*
3. **Cost:** This counts the average number of 'hops' it takes to deliver messages. *Cost = total number of hops to deliver messages / total delivered messages*

Since Latency and Cost are measured only over delivered messages it is important to know the proportion of messages that are successfully delivered. At the same time it is important to remember that the simulation attempts to pass messages between all pairs of nodes so some of these source/destination pairs would never arise in practice.

The importance of Cost as an evaluation criterion is difficult to assess without a specific application context. In some scenarios it may not matter whether a message passes through three or six nodes on its way to the destination. On the other hand a flooding strategy does not appear to be reasonable either.

5.2 Simulation

To evaluate the performance of BubbleH, we have used ContactSim a discrete time event simulator designed specifically for evaluating DTN routing. ContactSim is capable of using contact traces recorded either during real-world experiments or generated synthetically.

Results. In figure 4 we show the results of routing based on the communities found using four community finding algorithms applied to *MIT-NOV*. In addition to KCLIQUE and HGCE, we tested InfoMap and LinkClustering to see if the overlapping and hierarchical nature of LinkClustering would achieve better results that the non-overlapping InfoMap.

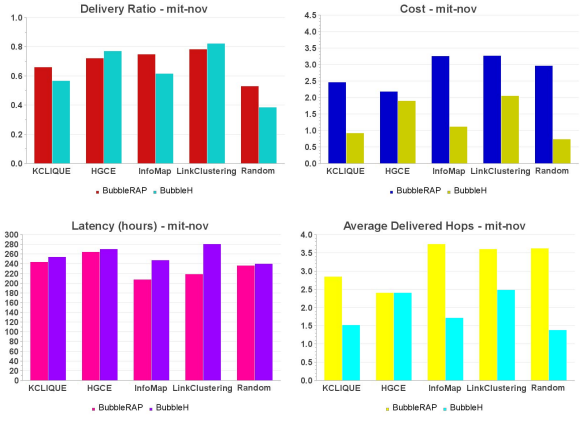


Fig. 4. Result of routing using BubbleH and BubbleRAP on the MIT-NOV dataset, showing the *delivery ratio*, *cost*, *latency* and the number of *delivered hops*

In this case, we see that BubbleH performs well in terms of delivery ratio when using the hierarchical and overlapping community finding algorithms HGCE and LinkClustering. It also does well in terms of cost for all methods. Latency is impacted overall (perhaps because some more difficult messages are being delivered), but the number of hops taken to deliver the messages is lower in most cases, suggesting that BubbleH is more selective than BUBBLE Rap, and 'holds' messages for longer, to ensure a more direct route.

6 Evaluation

In the first experiment described above, we sent messages between *all* pairs of nodes, over the *entire* dataset period. However, we also wanted to test how well our algorithms coped using a more realistic approach. To achieve this, we split the datasets into two distinct periods; a training period, in which the community finding algorithms are allowed to find communities; and a test period, in which the routing algorithm tries to deliver messages. We only send messages between nodes that communicated during the training phase, meaning only selected pairs of nodes were used. We believe this to be a more realistic experimental set-up, as in practice we would not expect all nodes to communicate with every other node. We expect this to result in a higher delivery ratio.

We chose five datasets with which to test our algorithms, here we briefly describe each dataset we use for our simulations. Two of the datasets (Enron and Studivz) don't represent real DTN scenarios but do have the contact characteristics required for the simulation.

MIT Reality. The authors have captured a trace of human contacts over a period of time: communication, proximity, location, and activity information from 100 subjects at MIT over the course of the 2004-2005 academic year. We chose a 4 month section in the most active period (Nov 2004 to Feb 2005); the first month for training, and the remaining three months as the test period. We refer to this as *MIT-SPLIT*.

Enron Email Corpus. This dataset, introduced by Klimt and Yang [8], comprises a large set of emails made public during the legal investigation concerning the Enron corporation. We used the occurrence of an email sent between individuals to represent a physical contact with a duration of 1 second. In this dataset we picked a busy three month period, between April 2001 and July 2001 we refer to this dataset as *ENRON-SPLIT-A*.

Social Sensing. The Social Sensing Study was undertaken by the CLARITY centre for sensor web technologies and took place at University College Dublin and Dublin City University. Mobile phones carried by participants collected data about their interactions, including nearby Bluetooth devices, cell towers and WiFi Access points. We chose a 6 month period for our simulation between Jan 2009 and June 2009, we designated the first two months as the training period. We refer to this dataset as *SOCIAL-SENSING-SPLIT*.

Studivz. This dataset is derived from wall posts from a university social networking site Studivz[2] , we determine contacts occur when a post is made on the 'wall' of another person. Each contact event is given a duration of 1 second per character in the wall post. We chose an active period of three months (from Oct 2007) to use, resulting in a network containing over 28 thousand nodes. To reduce the dataset to a more manageable size, comparable with other datasets we evaluate, we randomly selected three nodes from the aggregate graph of the three month period, and followed the edges to a depth of 2 hops, this resulted in sub-graph of 385 nodes. We refer to this as *STUDIVZ-SPLIT*

Hypertext2009. This dataset was collected during the ACM Hypertext 2009 conference, where the SocioPatterns[3] project deployed the Live Social Semantics application. Conference attendees volunteered to wear radio badges that monitored their face-to-face proximity. The dataset represents the dynamical network of face-to-face proximity of 110 conference attendees over about 2.5 days. We used the first day for training and the remaining time for the test period, we refer to this as *HYPERTEXT2009-SPLIT*.

6.1 Results

Figure 5 shows the resulting *delivery ratio* for BUBBLE Rap and BubbleH, when routing over five datasets, for each community finding algorithm. We notice that for *MIT-SPLIT*, there is not such a pronounced improvement by BubbleH. The story is the same for the other datasets, where there is little difference in *delivery ratio* between BUBBLE Rap and BubbleH, however interestingly, we see that

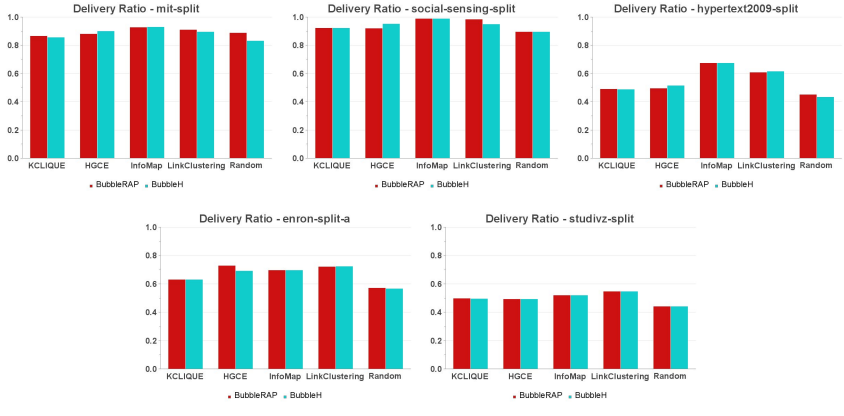


Fig. 5. Delivery Ratio for BUBBLE Rap and BubbleH, for five datasets

[2] http://www.studivz.net/
[3] http://www.sociopatterns.org

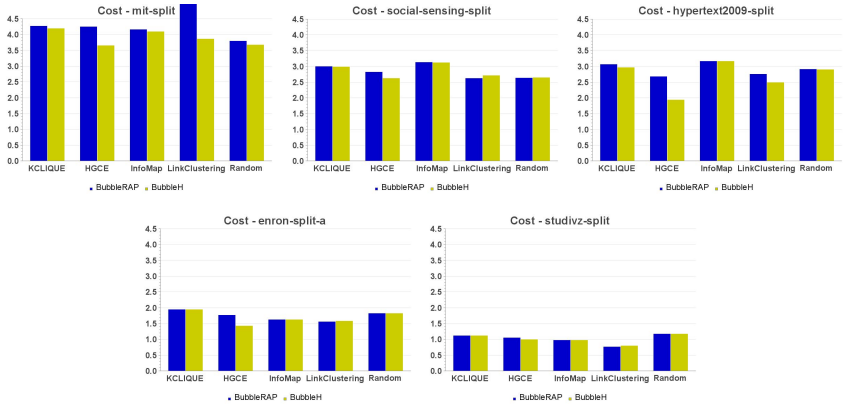


Fig. 6. Cost for BUBBLE Rap and BubbleH, for five datasets

in most cases, InfoMap and LinkClustering perform better than KCLIQUE, H-GCE and Random. The flow based approach of InfoMap and the large number of overlapping community structures of LinkClustering appear to make for better routing decisions.

In terms of transmission *cost*, shown in Figure 6, we see that BubbleH generally performs better than BUBBLE Rap, H-GCE makes distinct improvements in cost for BubbleH in the *MIT-SPLIT* and *HYPERTEXT2009-SPLIT* datasets. The results in terms of *latency*, in Figure 7 are not as clear, in *MIT-SPLIT*, InfoMap and LinkClustering perform well, but BUBBLE Rap and BubbleH have nearly an equal number of wins. The *STUDIVZ-SPLIT* dataset shows a very poor *latency*, and a very low *cost*. This suggests that the network is very sparsely connected, perhaps with many of the nodes only contacting each other very occasionally.

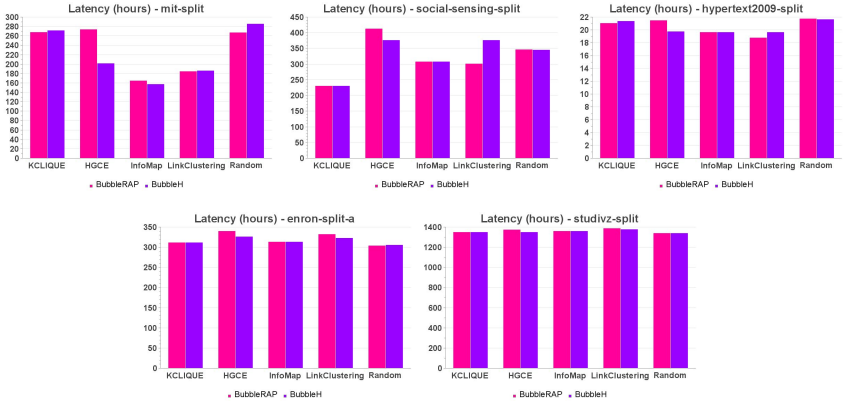


Fig. 7. Latency for BUBBLE Rap and BubbleH, for five datasets

It is interesting that H-GCE performs well in the real-world contact datasets, suggesting that there may be a distinct difference between real-world contacts, and virtual contacts. There is a clear difference in results between the *MIT-NOV* from the first experiment and the *MIT-SPLIT* dataset, this is due to the nature of how the message pairs were selected. In the case of the *SPLIT* datasets, only pairs who communicated during the training phase created messages to be delivered to one another.

7 Conclusion and Future Work

In this paper, we show that our hierarchicy based community messaging algorithm BubbleH, performs better than it's non-hierarchical predecessor, BUBBLE Rap, on five different datasets. We have also evaluated a number of community finding strategies, and have found that LinkClustering with its overlapping hierarchical clustering, and InfoMap, with it's flat partitioning, perform well. InfoMap's success may be due to the optimization that drives it; based on the concept of flow between communities. Perhaps the combination of LinkClustering's highly overlapping and hierarchical approach gives it the edge over the other algorithms.

Future work should explore the extent to which community finding affects the results of routing schemes on different types of dataset, in this way, we may be able to benchmark different community finding approaches, by means of DTN routing performance tests.

A further challenge is the need to incorporate a scheme that includes the dissemination of network structure data in an efficient manner, which will allow nodes to individually calculate community structures, allowing the system to be truly distributed.

Another aspect that has not been fully explored in the literature, is how to use a node's physical location as a driver for routing in communities. Future work should examine new ways to use a persons location to firstly, affect community finding, and secondly, aid routing.

Acknowledgements. This work is supported by a PhD scholarship from the Irish Research Council for Science Engineering and Technology and by Science Foundation Ireland Grant No. 08/SRC/I140 (Clique: Graph and Network Analysis Cluster).

References

1. Ahn, Y., Bagrow, J., Lehmann, S.: Link communities reveal multiscale complexity in networks. Nature 466(7307), 761–764 (2010)
2. Crowcroft, J., Yoneki, E., Hui, P., Henderson, T.: Promoting tolerance for delay tolerant network research. SIGCOMM Comput. Commun. Rev. 38(5), 63–68 (2008)
3. Eagle, N., (Sandy) Pentland, A.: Reality mining: sensing complex social systems. Personal and Ubiquitous Computing 10(4), 255–268 (2006)

4. Fortunato, S.: Community detection in graphs. Physics Reports 486(3-5), 75–174 (2010)
5. Hui, P., Chaintreau, A., Scott, J., Gass, R., Crowcroft, J., Diot, C.: Pocket switched networks and human mobility in conference environments. In: Proceeding of the 2005 ACM SIGCOMM Workshop on Delay-Tolerant Networking, WDTN 2005, pp. 244–251 (2005)
6. Hui, P., Crowcroft, J., Yoneki, E.: Bubble rap: social-based forwarding in delay tolerant networks. In: Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing, MobiHoc 2008, pp. 241–250. ACM, New York (2008)
7. Karrer, B., Levina, E., Newman, M.E.J.: Robustness of community structure in networks. Physical Review E 77(4), 046119 (2008)
8. Klimt, B., Yang, Y.: Introducing the Enron corpus. In: First Conference on Email and Anti-Spam, CEAS (2004)
9. Lee, C., Reid, F., McDaid, A., Hurley, N.: Seeding for pervasively overlapping communities. Phys. Rev. E 83, 066107 (2011)
10. Lindgren, A., Doria, A., Schelén, O.: Probabilistic routing in intermittently connected networks. ACM SIGMOBILE Mobile Computing and Communications Review 7(3), 19 (2003)
11. Musolesi, M., Mascolo, C.: CAR: Context-Aware Adaptive Routing for Delay-Tolerant Mobile Networks. IEEE Transactions on Mobile Computing 8(2), 246–260 (2009)
12. Palla, G., Derényi, I., Farkas, I., Vicsek, T.: Uncovering the overlapping community structure of complex networks in nature and society. Nature 435(7043), 814–818 (2005)
13. Rosvall, M., Axelsson, D., Bergstrom, C.: The map equation. The European Physical Journal-Special Topics 178(1), 13–23 (2009)
14. Rosvall, M., Bergstrom, C.: Mapping change in large networks. arXiv, 1–9 (2008)
15. Spyropoulos, T., Psounis, K., Raghavendra, C.S.: Spray and wait: an efficient routing scheme for intermittently connected mobile networks. In: Proceedings of the 2005 ACM SIGCOMM Workshop on Delay-Tolerant Networking, WDTN 2005, pp. 252–259. ACM, New York (2005)
16. Stabeler, M., Lee, C., Williamson, G., Cunningham, P.: Using Hierarchical Community Structure to Improve Community-Based Message Routing. In: ICWSM 2011 Workshop on Social Mobile Web Workshop, SMW 2011 (2011)
17. Vahdat, A., Becker, D.: Epidemic Routing for Partially Connected Ad Hoc Networks. Tech. Rep. CS-2000-06, Duke (2000)
18. Zhang, Z.: Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: overview and challenges. IEEE Communications Surveys & Tutorials 8(1), 24–37 (2006)

Routing for Opportunistic Networks Based on Probabilistic Erasure Coding

Fani Tsapeli and Vassilis Tsaoussidis

Space Internetworking Center, (SPICE),
ECE Department, Democritus University of Thrace, Xanthi, Greece
{ttsapeli,vtsaousi}@ee.duth.gr

Abstract. We investigate the problem of routing in opportunistic networks and propose a novel routing algorithm, which combines probabilistic routing with erasure coding. Erasure coding generates large amounts of code blocks with fixed overhead; we apply a sophisticated but realistic method to allocate the generated code blocks to nodes that relies on a probabilistic metric for evaluating node potential. Our goal is to enhance further the robustness of the erasure-coding based forwarding in worst-case delays but also in small-delay scenarios. We detail our algorithm and evaluate its performance against other well-known routing algorithms. We exploit scenarios with adequate resource storage as well as scenarios with limited storage capacity. In both cases, our algorithm yields promising results.

Keywords: routing; delay tolerant networks; erasure coding.

1 Introduction

Opportunistic networks exemplify characteristics of delay tolerant networking [1] (DTNs) in environments where contacts appear opportunistically. Common examples of such networks are mobile sensor networks [2], underwater sensor networks [3], pocket switched networks [4], or transportation networks [5], where delay tolerance constitutes an inherent property of limited bandwidth, energy or sparse connectivity. In such networks, traditional routing techniques for ad hoc networks, which assume that an end-to-end path between a source and a destination node always exists, cannot be applied. Routing in opportunistic networks is a particularly challenging problem since it does not simply rely on connectivity to establish optimal routes but rather attempts to exploit windows of connectivity opportunities. Therefore, routing in opportunistic networks becomes a scheduling problem with inherently probabilistic properties.

Routing in opportunistic networks is based on the store-carry-and-forward technique according to which messages are transferred using both transmissions and node physical movement. Thus, node mobility is exploited to address the problem of limited connectivity. A general technique to enhance reliability and reduce delivery delay is the use of a replication scheme according to which identical message copies

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 257–268, 2012.

are spread to the network in parallel. Spreading sufficiently large number of replicas in the network increases the probability that at least one of them will reach its final destination. However, transmission of multiple copies wastes network resources. Thus, there is a trade-off between delivery latency and traffic overhead.

Erasure coding has been proposed as an alternative way to generate redundancy, instead of applying simple replication. According to this technique, each message is encoded into such a number of code blocks that the reconstruction of the original message requires only a specific amount of the generated code blocks. The code blocks are spread throughout a large number of relay nodes, which may carry individually less data than the whole message copy. Utilization of a large number of relays for the transmission of messages renders erasure coding based forwarding techniques more robust to the occasional failures of some relays. However, since it takes longer for the source node to complete the distribution of the code blocks, erasure coding based techniques are associated with prolonged delivery latency.

In this work, we propose a novel erasure coding based forwarding algorithm for heterogeneous networks. We apply an effective policy to allocate the generated code blocks based on the node ability to transfer the blocks further to their final destination. We examine our scheme in conjunction with other well-known replication based forwarding schemes, in scenarios with sparse connectivity and limited buffer resources. We evaluate the performance of the proposed algorithm in terms of delivery latency and delivery ratio.

The rest of the paper is organized as follows. In section 2, we briefly discuss the existing routing approaches for opportunistic networks and emphasize on erasure-coding based methods. In section 3, we present our proposed routing policy. In section 4 we present our evaluation based on selected simulation results and in section 5 we summarize our conclusions and future directions.

2 Related Work

Several approaches have been proposed to deal with the problem of routing in opportunistic delay-tolerant networks. Most of them are based on spreading identical message copies in the network. Epidemic routing [6], a flooding-based method, constitutes a typical example of this category of algorithms. Although epidemic routing achieves the best performance in terms of delivery delay, it imposes significant overhead which renders it impractical in reality. *Controlled Flooding* schemes have been proposed towards this direction, namely to alleviate epidemic routing from the burden of resource exhaustion.

Controlled Flooding schemes cannot guarantee a fixed overhead and, in the worst case, they may act exactly like in epidemic routing. In [11], the *Spray and Wait* forwarding scheme has been proposed to ensure fixed overhead while achieving acceptable delay latency. According to this scheme, a specific number of copies is forwarded to the r first relays that the source node encounters. Each of these relay nodes is only allowed to perform direct transmission to the final destination.

Authors in [12], propose a forwarding algorithm based on erasure coding. According to this algorithm, a message of size M bytes is encoded using an erasure coding

algorithm (e.g. Reed-Solomon coding) with replication factor r, into $N = \frac{Mxr}{b}$ blocks where b is the size of each block in bytes and r is the replication factor. These code blocks are equally split among the first $k \cdot r$ relays. A message can be decoded iff at least *N/r* code blocks reach their final destination. Thus, only k out of $k \cdot r$ relays should deliver their blocks to the final destination in order to have a successful message delivery. This method has the same overhead as in *Spay and Wait* with replication factor r, while k more relays are utilized. It has been proved that the erasure-coding based forwarding scheme achieves better worst-case delay performance than *Spray and Wait.* However, it incurs prolonged delivery latency in typical cases. In an attempt to improve the performance of the erasure-coding technique in small delay performance cases, authors in [13] present a hybrid scheme which incorporates aggressive erasure coding forwarding. This technique has improved performance both in worst delay performance cases and in very small delay performance cases. However, this gain comes at the cost of double overhead.

Both the former erasure-coding based routing schemes assume a simple scenario in which all nodes are equally qualified to act as relays for a message. Considering nodes with different characteristics, a more sophisticated method to allocate code blocks could be investigated. Such an approach is presented in [14], where the performance of different allocation methods, in terms of optimizing the delivery probability in the presence of path failures, is examined.

In [15], erasure-coding based forwarding is deployed along with an estimation-based scheme. According to the authors, the generated code blocks are split to the relays in proportion to their delivery predictability. However, to the best of our knowledge, this method has only been evaluated over reliable networks, where nodes never fail or drop messages. We claim here that a simple proportional allocation of code blocks cannot constitute a viable solution but instead, it may frequently result in a typical simple erasure coding forwarding scheme. However, the estimation concept presented in [15] can be further exploited.

Therefore, in our present work we depart from [15] and investigate an enhanced method for the allocation of code blocks. Our goal is to reduce the delivery delay of simple erasure coding for the small delay performance cases while retaining its improved performance to worst-case delays. Thus, we allow relays that are considered to be *good* – according to our justified criteria – to carry as many blocks as it is required in order to successfully decode a message; however we also exploit nodes with worse delivery predictability as relays for less number of code blocks. We determine both an upper and a minimum threshold to the number of blocks that a node can carry in order to confine the number of relays that will be utilized.

3 Probabilistic Erasure Coding Routing

As it was stated earlier, our goal is to improve the performance of erasure-coding based routing in terms of delivery latency in both small delays and worst-case delays. Thus, we apply an effective and quick method for dispatching the code blocks and we define a proper metric to distinguish between relays with high probability to deliver a

message to its destination (*good* relays) and relays with lower probability (*bad* relays). However, although our proposal prompts to a binary selection process, the probabilistic nature of our scheme along with the diversity of our dispatching methodology, cancels the main drawbacks of binary – and hence possibly faulty – strategy. In this section, we describe comprehensively our proposed routing scheme.

3.1 Proportional Allocation Method

An effective method of allocating code blocks should utilize each contact opportunity and let *good* relays carry more messages while exploiting also *bad* relays by dispatching them a small portion of code blocks. The technique proposed in [15] satisfies this statement and incorporates high reliability since it accounts for the possibility of fault prediction and consequently false characterization of the binary relay state (i.e., *good* vs. *bad*). However, our analysis shows that dispatching a number of blocks to a node in direct proportion to its delivery predictability is not an effective technique.

Let consider the allocation method proposed in [15]. When two nodes meet and one of them has more code blocks than a threshold G it re-dispatches them proportionally to the estimation metrics. The estimation metric is defined as the number of contacts $N_{i,j}$ that node i had with the destination node j within time interval T. Assume that node A has m_A code blocks for a given message destined to node C and node A encounters node B. It will send to node B m_B messages where

$$m_B = m_A \cdot \frac{\tau_{B,C}}{\tau_{A,C} + \tau_{B,C}} \tag{1}$$

and $\tau_{i,j} = N_{i,j}/T$.

Consider a simple scenario where node A has encoded a message with erasure coding and replication factor three. Thus, a destination node E should receive the 33.3% of the generated code blocks in order to decode the message. The threshold G is set to the 33.3% of the code blocks. Assume the following information about the number of contacts that nodes A, B, C and D, respectively, had with node E within the time interval T:

Table 1. Number Of Contacts Of Nodes A, B, C And D With Node E

$N_{A,E}$	$N_{B,E}$	$N_{C,E}$	$N_{D,E}$
10	2	4	10

Now let's assume that node A encounters nodes B, C and D at times t1, t2 and t3 correspondingly and re-dispatches its code blocks according to equation (1). The result of this procedure is depicted in Table 2.

According to our sample allocation, at least two out of four relays are required to succeed in order to decode the message successfully. It should be noticed that, in this example, the result accrued from the proportional allocation method matches the gain of the simple allocation method, where blocks are equally distributed among four relays. Furthermore, since only few relays are utilized, the robustness of the erasure coding technique in worst-case delays may be jeopardized.

Table 2. Blocks Allocation According To Equation (1)

	m_A %	m_B %	m_C %	m_D %
t0	100	0	0	0
t1	83.3	16.7	0	0
t2	59.5	16.7	23.8	0
t3	29.75	16.7	23.8	29.75

3.2 The Proposed Allocation Method

According to our method, each node maintains a table with metrics for each other known node in the network. The metric of node i to node j represents the properness of node i to act as a relay for messages destined to node j. As we detail later, each node will eventually apply predefined rules that elaborate on these metrics in order to characterize whether a relay for a given message is *good* or not.

It is certainly desirable to confine the maximum number of relays that can be utilized – spreading code blocks into too many relays would reduce the performance of our algorithm in terms of delivery latency. Thus, we define both a minimum quantity of code blocks Q_{min} and a maximum number of code blocks Q_{max} that a node is obliged, or allowed, respectively, to carry. Q_{max} is set to 1/r of the generated code blocks and it should be a multiple of Q_{min}. Notice that while Q_{min} is a tunable parameter, Q_{max} is inflexible.

For simplicity, two nodes will not exchange any blocks of a given message if both of them have already code blocks of it. Each node that carries more than Q_{max} code blocks, in each contact opportunity, will dispatch Q_{max} code blocks to the other node. Each node *i* that carries N_i code blocks where $Q_{min} < N_i \leq Q_{max}$, upon encountering node *j*, it will use the rule $R_{split}^{i,j}$ to decide whether it is *good* relay for this message or not. If it is a *bad* relay it will dispatch half of its code blocks to node *j*. If node *i* is *good* relay for a message or if it contains Q_{min} code blocks, thus it cannot split blocks any further, if node *j* is considered to be better relay according to rule $R_{forward}^{i,j}$, it will send all its code blocks to *j*.

According to our method, each node initially receives Q_{max} code blocks and decides afterwards whether it should split or not based on rule R_{split}. However, some relays may experience very sparse connectivity or too limited buffer resources; hence, they will not be able to forward the blocks they have received. Thus, a node *i* with $N_i > Q_{max}$ blocks should use the rule $R_{low}^{j,i}$ to determine whether node j is such a *bad* relay. In that case, it will dispatch to j only the minimum quantity of blocks Q_{min}. Correspondingly, nodes with high delivery probability would be more effective on dispatching their blocks. Thus, each node i with $N_i > Q_{max}$ blocks should apply a rule R_{high} to recognize these advanced relays and dispatch to them more than Q_{max} blocks. The pseudo-code of our model is presented below.

Algorithm The allocation method

Description: Node i which has N_i code blocks of a message encounters node j which does not have any code blocks.

```
1:  if (N_i > Q_max)
2:      if ((R_high^{j,i}) && (! R_high^{i,j}))
3:          N_j = N_i - Q_max
4:      else if ((R_high^{j,i}) && (R_high^{i,j}))
5:          N_j = ⌊N_i / (2 · Q_min)⌋ · Q_min
6:      else if (R_low^{j,i})
7:          N_j = Q_min
8:      else
9:          N_j = min (Q_max, N_i − Q_max)
10: else if ((Q_min < N_i ≤ Q_max) && (R_split^{i,j}))
11:     N_i = ⌊N_i / (2 · Q_min)⌋ · Q_min
12: else if (R_forward^{i,j})
13:     N_j = N_i
14: N_i = N_i − N_j
```

It should be noticed that each node in the DTN will carry $n \cdot Q_{min}$ code blocks, with $\{n \in \mathbb{Z} \mid 0 \leq n \leq Q_{max}/Q_{min}\}$. Thus, the number of relays n_r that will be utilized is confined as: $r \leq n_r \leq r \cdot Q_{max}/Q_{min}$.

3.3 Rules Definition

The effectiveness of our algorithm is partially based on the selection of proper rules; indeed the rules determine the ability of a node to act as relay for a message - and hence, inherently affect the efficacy of the proposed scheme.

We assume that nodes follow a mobility pattern according to which nodes that have been previously encountered it is likely to be encountered again. This assumption does not hold for all types of mobility; however, it is valid for most scenarios since users do have mobility patterns that reflect repetitive behaviors or mobility preferences. Thus, a node can decide about its properness to act as relay for a message based on the history of its encounters. According to our model, each node retains a table with probabilities to meet each other node in the network. We use the same model as PROPHET protocol [10] to compute and update these probabilities.

According to PROPHET, when node a encounters node b, it updates its delivery probability for node b according to equation (2):

$$P_{(a,b)} = P_{(a,b)old} + \left(1 - P_{(a,b)old}\right) \times P_{init} \tag{2}$$

where P_{init} is an initialization constant.

The probabilistic metric is also assumed to have a transitive property according to which node a updates its delivery probability for node c according to equation (3):

$$P_{(a,c)} = P_{(a,c)old} + (1 - P_{(a,c)old}) \times P_{(a,b)} \times P_{(b,c)} \times \beta \tag{3}$$

where β is a constant.

Finally, equation (4) is used to decrease the probability of node a to encounter node b if these two nodes have not been encountered for a period of time:

$$P_{(a,b)} = P_{(a,b)old} \times \gamma^k \tag{4}$$

where γ is an aging constant and k indicates the number of time slots that have elapsed since the last update of the metric.

We should also incorporate into our decision the available storage resources of communicating nodes that act as potential relays. In order to compare nodes i, j in terms of their storage availability we define the *Storage* metric as described in (5):

$$Storage_{(i,j)} = \frac{FreeSpace_i}{\max\left(BufferSize_i, BufferSize_j\right)} \tag{5}$$

Finally, we define the rules $R^{i,j}_{split}$, $R^{i,j}_{forward}$, $R^{i,j}_{low}$ and $R^{i,j}_{high}$, according to which node i dispatches to node j a portion or all of its code blocks that are destined to node d, as follows:

$$R^{i,j}_{split} = \left(P_{(i,d)} < T_{prob}\right) \quad \vee \quad \left(Storage_{(i,j)} < T_{buf}\right) \tag{6}$$

$$R^{i,j}_{forward} = \left(P_{(i,d)} < P_{(j,d)}\right) \ \wedge \left(Storage_{(j,i)} > T_{buf}\right) \tag{7}$$

$$R^{i,j}_{low} = \left(4 \times P_{(i,d)} < T_{prob}\right) \vee \left(Storage_{(i,j)} < T_{buf}\right) \tag{8}$$

$$R^{i,j}_{high} = \left(P_{(i,d)} > 4 \times T_{prob}\right) \wedge \left(Storage_{(i,j)} > T_{buf}\right) \tag{9}$$

where T_{prob} and T_{buf} are predefined thresholds.

4 Simulation Results

We evaluate our *probabilistic erasure coding* scheme (*ProbEC*) in terms of delivery delay by performing a set of simulations. We compare our scheme with other forwarding algorithms such as *spray and wait* (*SnW*) [11] and *erasure coding* (*EC*) [12]. In order to demonstrate the impact of erasure coding alone, we also examine the performance of *ProbEC* when we set $Q_{min} = Q_{max}$, allowing each node to carry exactly the number of code blocks that are required for the message's reconstruction. This

strategy differs from the simple replication algorithm as each node that encounters a node with better delivery predictability, will forward all of its blocks. We name this forwarding policy as *probabilistic replication* (*ProbRep*).

4.1 Simulation Model

We implemented our model in the Opportunistic Network Simulator, ONE [16]. The simulation area size is 3600×3600m^2. Nodes follow a restricted random waypoint movement model similar to that of [15]. We divide the whole area in four equal square subareas A, B, C and D. We randomly set 40 waypoints in each subarea. Each node in the network has a given set of 30 waypoints. Nodes choose randomly a destination from their set of waypoints and a random velocity within an allowed interval. After arriving at the destination they wait for a random time interval and then they pick another destination. We consider four groups of nodes, each one containing 20 mobile nodes. We define different groups of nodes in order to allow nodes belonging to the same group to move within a specific subarea with higher probability. In Table 3 we present the percentage of waypoints within a specific area that nodes have according to the group that they belong to. It becomes apparent that our simulation scenario matches real-life scenarios and is aligned with the probabilistic policy that was incorporated in equations (2), (3) and (4).

Table 3. Waypoints Distribution Per Group Of Nodes

	Waypoints in area A	**Waypoints in area B**	**Waypoints in area C**	**Waypoints in area D**
1st Group	80%	10%	10%	0%
2nd Group	0%	80%	10%	10%
3rd Group	10%	0%	80%	10%
4th Group	10%	10%	0%	80%

Each group of nodes contains nodes with different velocity ranges. More precisely, each group is consisted of 15 slow moving nodes whose velocities range from 0.5m/s to 2m/s and 5 fast moving nodes with velocity range from 4m/s to 6m/s.

According to our simulation model, a message is created every 30 seconds. The source and the destination of the message are randomly selected among the 80 nodes of the network. Hence, the deterministic manner used for generating messages is enhanced by the random location of communication, which in terms results in non-deterministic generation of messages at each communication sub-area. We allow nodes to move within the simulation area without creating any messages for the first 12000 seconds in order to initialize their delivery probability tables. Afterwards, messages are created for the next 46000 seconds. The whole simulation time of each experiment is set to 216000 seconds, a duration that is deemed sufficient for the algorithms to exploit their potential.

Thresholds T_{prob} and T_{buf} have been set to 0.015 and 0.1 correspondingly and parameters P_{init}, β and γ to 0.7, 0.3 and 0.98. The message size is set to 2000 bytes

and the block size 250 bytes. We examine the performance of our algorithm using replication factor 4. Thus, we generate 32 blocks per message. In the following experiments, we set the Q_{max} parameter of *ProbEC* to 8 blocks and the Q_{min} to 2 blocks. Thus, the minimum number of utilized relays is 4 and the maximum 16. Respectively, according to the examined EC scheme, code blocks will be equally split to 16 relays. Although we repeated the experiments with various parameters (e.g., message sizes or rates) we only present here representative results for each scenario.

4.2 Experiments in General Network Scenarios

In this section, we evaluate the performance of our algorithm in scenarios with adequate buffer resources. In this scenario, 5 out of 15 slow moving nodes in each group have 160KB buffer, while all the other nodes have 360KB buffer.

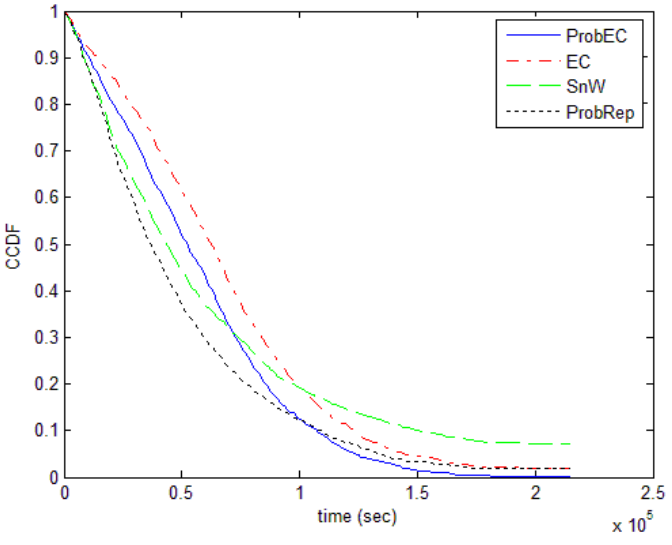


Fig. 1. CCDF of messages delivery latency for ProbEc, EC, SnW and ProbRep routing algorithms in scenario with adequate buffer resources

In Fig. 1 we present the simulation results of data latency distribution in complementary CDF (CCDF) curves for ProbEC, EC, SnW and ProbRep. From Fig. 1, it is clear that ProbEC has improved performance compared with EC both for small and for worst delay performance cases. While SnW still outperforms our algorithm for small delay performance cases, we achieve significant improvement for worst-case delays. We note, however, that small delays do not really pose a challenge; longer delays, instead may annoy users and impact quality of service significantly. ProbRep, acts as an enhanced spray and wait forwarding scheme, which enables nodes to forward their messages if they encounter a node with better delivery predictability: it performs very well both in small and in worst delay performance cases and owes its performance to the strategy of allowing bad relays to re-forward their messages

instead of just waiting until encountering the final destination. However, we observe that ProbEC outperforms the other approaches in worst-case delays; this is expected, since it utilizes more relays. In Table 4 we provide numerical results extracted from the previous curves, in order to allow an easy comparison of the examined forwarding schemes. According to this table, ProbEC manages to deliver the 90% of the generated messages faster than all the other algorithms and overall, it delivers more messages until the end of the experiment.

Table 4. Delivery Latency

Algorithm	Delivery Delay (seconds)				Successfully Delivered Messages
	25% of messages	50% of messages	75% of messages	90% of messages	
ProbEC	27000	52000	79000	106000	99.68%
EC	35000	62000	90000	122000	98.07%
SnW	19000	43000	83000	148000	94.21%
ProbRep	18000	41000	74000	110000	97.94%

4.3 Experiments in Scenarios with Limited Storage Resources

In this experiment, we evaluate the performance of our algorithm in scenarios with limited storage resources. We set the buffer capacity of 5 out of 15 slow moving nodes of each group equal to 20KB, while we retain the buffer size of all other nodes to 360KB.

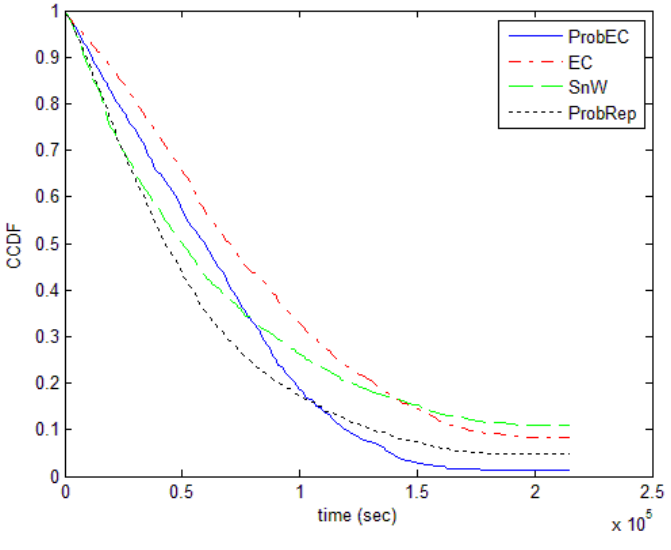


Fig. 2. CCDF of messages delivery latency for ProbEc, EC, SnW and ProbRep routing algorithms in scenario with limited buffer resources

It is clear both from Fig. 2 and Table 5 that while the performance of EC and SnW is decreased significantly compared to their performance in scenarios with adequate

storage resources, ProbEC and ProbRep have only a slight degradation in their performance. That is due to their attitude to forward further their messages when storage is exhausted. ProbEC still outperforms all the other algorithms and manages to deliver the 98.71% of the generated messages even in this extreme scenario.

While in the previous scenario we observe that the performance of the ProbEC scheme, compared to the ProbRep, is slightly improved, in this scenario the impact of adopting erasure coding is more obvious. In this experiment, ProbEC manages to deliver 2.5% more messages than ProbRep while in the previous experiment it delivers 1.74% more messages. Moreover, an interesting observation is that among the examined schemes, EC is the most vulnerable to data loss due to storage exhaustion; in this experiment it delivers 6.43% less messages than in the previous one, while SnW delivers 5.14% less messages correspondingly. EC scheme exemplifies significant performance degradation in experiments with limited storage resources because of its poor small delay performance; long delay in message delivery results in increased need of storage resources.

Table 5. Delivery Latency

Algorithm	Delivery Delay (seconds)				Successfully Delivered Messages
	25% of messages	50% of messages	75% of messages	90% of messages	
ProbEC	29000	59000	89000	118000	98.71%
EC	37000	70000	116000	169000	91.64%
SnW	19000	50000	103000	-	89.07%
ProbRep	20000	42000	82000	129000	96.21%

5 Conclusions and Future Work

We presented a novel routing algorithm for opportunistic networks, which combines probabilistic routing with erasure coding. According to our scheme, a set of rules prescribes the number of code blocks that a node should carry based on its suitability to act as relay for the specific message. Our goal is to exploit the potential of erasure coding scheme to improve the worst-case delay without damaging performance in small delay cases. Our simulation results demonstrate that our algorithm outperforms the simple erasure coding scheme both for small and worst delay performance cases.

The selection of appropriate value for threshold T_{prob} is very important. T_{prob} should vary within the range of node probabilities. Currently, we define T_{prob} statically. However, we plan to investigate the use of an adaptive threshold whose computation will be based on information exchanged by nodes. Finally, it will be interesting to also investigate in more detail the impact of T_{prob} selection.

Acknowledgments. The research leading to these results has received funding from the European Community's Seventh Framework Programme ([FP7/2007-2013_FP7-REGPOT-2010-1, SP4 Capacities, Coordination and Support Actions) under grant agreement **n° 264226** (project title: Space Internetworking Center-SPICE). This paper reflects only the authors views and the Community is not liable for any use that may be made of the information contained therein.

References

1. Delay tolerant networking research group, http://www.dtnrg.org
2. Wang, Y., Wu, H.: Dft-msn: The delay fault tolerant mobile sensor network for pervasive information gathering. In: IEEE Infocom (2006)
3. Heidemann, J., Ye, W., Wills, J., Syed, A., Li, Y.: Research challenges and applications for underwater sensor networking. In: Proceedings of IEEE WCNC (2006)
4. Hui, P., Chaintreau, A., Scott, J., Gass, R., Crowcroft, J., Diot, C.: Pocket switched networks and human mobility in conference environments. In: WDTN 2005: Proceeding of the 2005 ACM SIGCOMM Workshop on Delay-Tolerant Networking (2005)
5. LeBrun, J., Chuah, C., Ghosal, D., Zhang, M.: Knowledge-based opportunistic forwarding in vehicular wireless ad hoc networks. In: Proceedings of the Vehicular Technology Conference, vol. 4, pp. 2289–2293 (2005)
6. Vahdat, A., Becker, D.: Epidemic Routing for Partially-connected Ad hoc Networks. Technical Report CS-2000-06, Duke University (2000)
7. Burgess, J., Gallagher, B., Jensen, D., Levine, B.N.: MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks. In: Proceedings of IEEE Infocom, pp. 1–11 (April 2006)
8. Tseng, Y.-C., Ni, S.-Y., Chen, Y.-S., Sheu, J.-P.: The broadcast storm problem in a mobile ad hoc network. Wireless Networks 8(2/3), 153–167 (2002)
9. Chen, X., Murphy, A.L.: Enabling disconnected transitive communication in mobile ad hoc networks. In: Proceedings of Workshop on Principles of Mobile Computing, Colocated with PODC 2001 (August 2001)
10. Lindgren, A., Doria, A., Schelen, O.: Probabilistic routing in intermittently connected networks. SIGMOBILE Mobile Comput. Commun. Rev. 7(3) (2003)
11. Spyropoulos, T., Psounis, K., Raghavendra, C.: Spray and wait: an efficient routing scheme for intermittently connected mobile networks. In: WDTN 2005: Proceeding of the 2005 ACM SIGCOMM Workshop on Delay-Tolerant Networking, pp. 252–259. ACM Press, New York (2005)
12. Wang, Y., Jain, S., Martonosi, M., Fall, K.: Erasure-coding based routing for opportunistic networks. In: WDTN 2005: Proceeding of the 2005 ACM SIGCOMM Workshop on Delay-Tolerant Networking, pp. 229–236. ACM Press, New York (2005)
13. Chen, L.-J., Yu, C.-H., Sun, T., Chen, Y.-C., Chu, H.H.: A Hybrid Routing Approach for Opportunistic Networks. In: CNANTS 2006: Proceedings of the 2006 SIGCOMM Workshop on Challenged Networks (2006)
14. Jain, S., Demmer, M., Patra, R., Fall, K.: Using Redundancy to Cope with Failures in a Delay Tolerant Network. In: ACM SIGCOMM Workshop on Delay Tolerant Networks (2005)
15. Liao, Y., Tan, K., Zhang, Z., Gao, L.: Estimation based Erasure-Coding Routing in Delay Tolerant Networks. In: IWCMC 2006: Proceedings of the 2006 International Conference on Wireless Communications and Mobile Computing (2006)
16. Keranen, A., Ott, J., Karkkainen, T.: The ONE simulator for DTN protocol evaluation. In: Simutools 2009: Proceedings of the 2nd International Conference on Simulation Tools and Techniques (2009)

On the Performance of Erasure Coding over Space DTNs*

Giorgos Papastergiou, Nikolaos Bezirgiannidis, and Vassilis Tsaoussidis

Space Internetworking Center, Department of Electrical and Computer Engineering,
Democritus University of Thrace, Xanthi, Greece
{gpapaste,nbezirgi,vassilis}@spice-center.org

Abstract. Erasure coding has attracted the attention of space research community due to its potential to present an alternative or complementary solution to ARQ schemes. Typically, erasure coding can enhance reliability and decrease delivery latency when long delays render ARQ-based solutions inefficient. In this paper, we explore the benefits of erasure coding for file transfers over space Delay Tolerant Networks, using a generic end-to-end mechanism built on top of the Bundle Protocol that incorporates LDPC codes along with an ARQ scheme. The results reveal significant insights on the tradeoff among efficient bandwidth exploitation and delivery latency. We quantify the performance gains when optimal erasure coding is applied and investigate in what extent theoretically optimal performance is affected when suboptimal code rates are used. Beyond that, we highlight the ability of erasure coding to provide different QoS to applications, in terms of file delivery latency, by properly tuning the code rate.

Keywords: Delay tolerant networking, Erasure coding, Deep-space communications.

1 Introduction

In this paper, we investigate mechanisms to explore erasure coding techniques in space communications without (i) expending unnecessarily significant bandwidth for error correction overhead that cannot have return in application throughput and (ii) over-investing in Automatic Repeat reQuest (ARQ) techniques when long delays dominate communication performance. Hence, we focus here on mechanisms and experiments that may promote our knowledge on how to dynamically administer the tradeoff between bandwidth and delay in space, with a minimal risk. Given the inherent advantage and significant standardization progress of Delay Tolerant Networking (DTN) in space communications [1], we explore this tradeoff within the space DTN framework.

* *The research leading to these results has received funding from the European Community's Seventh Framework Programme ([FP7/2007-2013_FP7-REGPOT-2010-1, SP4 Capacities, Coordination and Support Actions) under grant agreement* ***n° 264226*** *(project title: Space Internetworking Center-SPICE).* This presentation reflects only the authors views and the Community is not liable for any use that may be made of the information contained therein.

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 269–281, 2012.

Space communication channels are characterized by significant shadowing and fading events, which strongly reduce signal-to-noise ratio and thus introduce bit errors within link frames. In addition, adverse weather conditions can cause long fading events or even link disconnections. In such cases, typical data-link recovery fails, resulting in bursty frame losses, occasionally in the order of tens to thousands of lost frames. Link layer failures are reflected on the upper protocol layers as packet erasures, i.e., missing packets that need to be retransmitted; however, long propagation delays and disruptions render ARQ solutions inefficient, since retransmission latency degrades network performance and extends communication (and thus data delivery) time significantly.

Delay Tolerant Networking [2] incorporates two significant reliability enhancements, that is, *custody transfer* and *storage capability*; their combined impact allows for retransmissions with reduced delay, since the source shifts data and responsibility gradually to intermediate nodes, towards the destination. Hence, lost data will be retransmitted faster, from nodes closer to destination. Inherently, therefore, DTN appears as a natural solution to improve reliability in space. However, shifting data custody towards the destination and thus gradually reducing retransmission delay is not sufficient solution in its own right. Typical erasure coding techniques along with ARQ may also need to be adjusted. Erasure codes [3] have been introduced as complementary mechanisms to ARQ solutions, employing Forward Error Correction (FEC) strategies at higher layers with a clear-cut goal to reduce the number of retransmission rounds. Erasure coding imposes by definition a tradeoff between efficient bandwidth exploitation and faster delivery; however, the question "*when and to what extent does extra and redundant transmission effort translate into better application throughput"* has not yet been adequately addressed. This issue becomes vital in space, where a false strategy to exploit the tradeoff may cause minutes or hours of waiting.

Given that erasure coding is a packet-level FEC technique, it can be incorporated within any protocol of the CCSDS DTN protocol stack that deals with data units (i.e., packets, frames, etc.) and since the DTN architecture is accomplished in a hop-by-hop manner, these solutions will essentially follow the same approach. Although hop-by-hop erasure coding can be tuned for the separate link conditions, it poses several challenges in terms of interoperability and cross-support. On the contrary, the deployment of erasure coding in an end-to-end "transport layer" architecturally placed above the DTN architecture could offer several advantages. From a technical point of view, an end-to-end approach moves complexity towards the ends of the communication system and leads to lower total processing delays, since encoding/decoding processes reside only at the end nodes. From a networking point of view, implementing erasure coding in an upper layer allows for handling packet erasures that are not related to frame losses only, but span across the whole protocol stack (e.g., packet erasures caused by storage congestion or erroneous route calculations at the bundle layer). Furthermore, this approach allows applications to identify certain QoS requirements to the "transport" service below (e.g., delay constraints, packet sizes, etc.); these requirements are associated to the complete end-to-end path and code rate is adapted accordingly.

We note that a performance comparison between hop-by-hop and end-to-end approaches is out of the scope of this paper. Here, we evaluate the performance gains of erasure coding against typical ARQ solutions and investigate the dynamics of the associated trade-offs. Although our contribution to exploiting these dynamics includes a novel generic end-to-end erasure coding protocol, which is placed architecturally between the application and the Bundle Protocol [4], in this initial study a single hop topology is considered. This experimental protocol incorporates erasure coding operating on a per-packet basis, based on block Low Density Parity Check (LDPC) codes [5], and also packet-oriented retransmission of encoding packets whenever decoding is unsuccessful. The proposed coding strategy differs from typical Type-II Hybrid ARQ strategies in that the lost encoding packets are retransmitted and no additional recovery packets are generated. Real-time experiments are conducted using our DTN Testbed [6, 7] and file transfers of different sizes are considered.

The conclusions presented in this work quantify but also qualify this tradeoff in the context of:

a) the optimal gain of erasure coding for file transferring
b) the impact of over- / under-estimation of channel packet erasure rate (PER) on file delivery time and the associated waste of bandwidth resources
c) the capability of an end-to-end "transport layer" erasure coding service to administer QoS in terms of delivery latency

The remainder of the paper is organized as follows: In Section 2 we discuss the related work and we highlight our perspective within this context. In Section 3 we briefly describe the proposed erasure coding experimental protocol. In Section 4 we elaborate on the experimental methodology, metrics and evaluation cases. We present the results of our experimental analysis in Section 5, and finally, in Section 6 we conclude the paper and provide some directions for future research.

2 Related Work

Appropriate positioning of erasure codes within the CCSDS protocol stack is an issue that triggered an interesting debate [8] in the CCSDS community. For example, erasure codes may be implemented as application layer solutions [9], as mechanisms of CFDP [10, 11] or at the DTN Bundle Protocol extensions [12]. Furthermore, authors in [12] compare two alternative approaches to support packet-level FEC: CFDP and DTN bundle protocol extensions. However, they do not conclude in favor of one or another.

Other approaches to incorporate erasure coding in space communications that however do not follow CCSDS architecture also exist. RCP-Planet [13] is an end-to-end rate control protocol for Interplanetary Internet (IPN) that targets the delivery of real-time application data. The protocol incorporates a probing rate control scheme to cope with link congestion and error rate, in conjunction with a packet-level FEC that is based on Tornado codes. The term "real-time" is rather a euphemism for channels with high propagation delays. However it reflects the time constraints of space applications. In [14] the authors propose Uni-DTN, a non-CCSDS DTN convergence layer protocol for unidirectional transport to provide scalability for both unicast and

multicast distribution of DTN bundles. Although some ideas included in [14] are promising, they do not target space environments and therefore, comparisons cannot be made easily.

Although simulation results across different space environments (either near-Earth/Cislunar [10] or deep-space [11]) are available for some of the aforementioned approaches, they are mainly scenario-oriented and do not highlight the specific tradeoffs of erasure coding versus ARQ schemes. For example, evaluation results presented in [10, 11] are strictly confined within specific and predetermined QoS parameters (delivery time / loss probability) and classes of space data traffic and hence, conclusions can only be confined within this specific context.

On the contrary, we attempt to go beyond the scenario-confined conclusions and investigate the tradeoff *per se*; that is to focus on the benefits of erasure coding along with ARQ schemes for future space internetworking architecture, in a way that scenario-independent conclusions about its performance can be drawn. In order to be able to generalize our conclusions, we apply a generic ARQ scheme that incorporates LDPC codes.

In this work, we consider deep-space communication links, where the performance of typical ARQ solutions is highly affected. Our results reveal interesting dynamics that can constitute the basis for further investigations and guide the design of efficient protocol solutions incorporating erasure codes in the future. To the best of our knowledge, this is the first attempt to evaluate the performance of erasure coding using a real DTN testbed that fully implements the standard DTN architecture.

3 Erasure Coding Experimental Protocol

In this section, we describe the erasure coding (EC) mechanism used in our experiments. EC is incorporated in an end-to-end "transport layer" protocol built on top of the Bundle Protocol and operates only at the endpoints of the communication system. Then, EC is integrated into Interplanetary Overlay Network (ION) DTN implementation [15]. EC mechanism is based on two, large-block, patent-free LDPC codes, namely LDPC Staircase and LDPC Triangle, which are specified in [16]. LDPC Staircase is used in all experiments, since this code presents lower inefficiency ratios [17] for the code rates we are experimenting with. Since only file transfers are considered, complete files are passed by the application to EC for transmission.

Each time a file transmission is requested, the file is handed down by the application to EC for transmission. The file is partitioned into N number of LDPC blocks, where $N = fileSize / maximumBlockLength$. *MaximumBlockLength* is a configuration parameter defined *a priori.* In our experiments, each file typically constitutes a single LDPC block. However, experiments have been conducted also with a 60Mbyte file partitioned into four LDPC blocks of 15MByte length each, in order to specifically investigate the impact of block size itself on EC performance.

Each LDPC block is segmented into k fixed-length source packets. The k source packets along with the value of the code rate are passed on to the LDPC encoder and, consequently, n encoding packets are created, where $n = k / code\_rate$.

Since both LDPC Triangle and LDPC Staircase are systematic codes, the first k encoding packets are the k source packets. The remaining n-k encoding packets are FEC packets. Ideally, code rates could statistically exploit the historical characteristics of

the channel. Assuming that channel PER is known *a priori*, optimal code rate (i.e., the code rate that suffices to protect a file transmission, given some PER) is defined as follows:

$$R = \frac{1 - PER}{1 + \varepsilon \cdot (1 - PER)} \quad (1)$$

where ε is the LDPC inefficiency. In order for decoding to be successful, $(1+\varepsilon)k$ encoding packets are required at the receiver. We note that channel PER accounts for total packet erasures that occur across the end-to-end path.

All encoding packets are passed to BP *in order*, requesting unreliable transmission and each encoding packet is encapsulated into a single bundle. The reception of the last encoding packet (*checkpoint* packet) always triggers the transmission of an *acknowledgement* indicating either the successful (*Ack*) or unsuccessful (*Selective Negative Acknowledgment*, *SNAck*) block reception; hence, *acknowledgment* delivery must be guaranteed. For this reason, a retransmission timer for the last packet of each encoded LDPC block is set. Upon the arrival of an *Ack*, the timer is cancelled; otherwise, upon expiration of this timer, the packet is retransmitted. Upon the reception of a *SNAck*, lost encoding packets are retransmitted. *SNAcks* inform the sender about missing encoding packets, in the same way as in [18]. In order to guarantee reliability, the last packet of each retransmission round also triggers the reliable transmission of *acknowledgments*. When decoding succeeds, LDPC blocks are aggregated into a single application data unit and delivered to the application.

4 Experimental Methodology

4.1 Scenario - Parameters

Our research purpose is to evaluate the potential of erasure coding in deep-space environments. In order to emulate long delays, high error rates, and asymmetric space link channels, ESA DTN testbed [6, 7] established in Space Internetworking Center [19] was used. The testbed allows for emulating current and future DTN-based space communication scenarios and uses Network Emulator (Netem, [20]) to emulate space conditions (i.e., error rates, propagation delays and data rates). Our evaluation scenario is based on a deep-space mission paradigm, where an *in-situ* element that is used for planet exploration relays the observed scientific data for further processing towards Earth base stations. In this initial evaluation, we consider the deep-space link only, where the performance of typical ARQ-based solutions may degrade. In particular, we consider one-hop file transmissions from a Mars orbiter towards Earth. Communication parameters used were taken from Mars missions [21, 22].

Nevertheless, in order to reduce emulation time, conduct more experiment repetitions and increase statistical robustness, we choose to keep the *Bandwidth-Delay Product* (*BDP*) of the deep-space link fixed, by increasing bandwidth and decreasing propagation delay correspondingly. The results are normalized accordingly and thus independent conclusions can be drawn. For example, we show below that for fixed

BDP and PER, file delivery latency normalized based on the RTT depends on the BDP alone (Eq. 5).

Since the BDP remains stable, its capacity in terms of packets is the same. For given PER, the number of retransmitted packets (*rtxData*) and the number of retransmission rounds (*RtxRounds)* cannot pose statistical arguments. Assuming further that queuing and processing delays are negligible, total file delivery latency can be expressed as:

$$FDL_{total} = D_{pr} + D_{tr} \tag{2}$$

where D_{pr} is the total propagation delay, and D_{tr} the total transmission delay. In particular:

$$D_{tr} = \frac{(fileData + rtxData + fecData)}{BW} \tag{3}$$

$$D_{pr} = \frac{RTT}{2} + RTT \cdot RtxRounds = RTT \cdot (\frac{1}{2} + RtxRounds) \tag{4}$$

Thus, based on Equations 2-4 normalized file delivery latency (*NDL*) can be expressed as:

$$NDL = \frac{1}{2} + RtxRounds + \frac{(fileData + rtxData + fecData)}{BDP} \tag{5}$$

The downlink BDP used in our experiments was 120 Megabits (i.e., 15 Mbytes), which is the product of 20 min *RTT* and 100 Kbps downlink data rate. The uplink BDP used is 1.2 Megabits, hence providing the deep-space channel asymmetry. In order to investigate the tradeoffs of erasure coding in both a scenario- and link-independent way, all file sizes used in our experiments are expressed in correspondence to the downlink BDP. In particular, file sizes vary from 0.5xBDP (i.e., 7.5 MBytes) to 4xBDP (i.e., 60 MBytes). Files were truncated into bundles with payload size equal to 1024 Bytes.

We consider two distinct evaluation cases: In *Case 1* we evaluate the performance gains of the EC mechanism and compare it with a simple ARQ-based scheme. This scheme shares the same mechanisms for file partitioning, block segmentation, and packets transmission and retransmission with the EC mechanism. The only difference between these two schemes is that in the former case no encoding is performed and thus only the *k* source packets are transmitted. In case no file partitioning into blocks is performed, its operation is equivalent to the CFDP deferred NAK mode [23]. Typical PERs for deep-space conditions are considered. Additionally, higher PERs are used to emulate extreme space weather conditions (e.g., solar winds). Thus, PER varies between 0% and 30%. Code rate is adjusted according to Eq. 1. In *Case 2*, we evaluate how deviations in channel PER estimation affect the performance gains of the EC mechanism; in parallel we investigate in what extent end-to-end erasure coding can differentiate the service provided to file transfer applications. In this case,

PER remains fixed at 20%, while PER estimations vary between 0% and 35%. Code rate is adjusted based on the estimated PER according to Eq. 1, where inefficiency ratio ε is optimized for each PER and its value varies from $\varepsilon = 0.03$ for PER = 5% to $\varepsilon = 0.09$, for PER = 35%.

A simple independent packet erasure model is considered, in which a packet is lost with a probability equal to PER. A summary of the scenario parameters is given in Table 1 below. Each experiment was repeated adequate times and both average values and 95% confidence intervals are presented.

Table 1. Scenario Parameters and Values

Parameter	Value
EC/ARQ packet size (Kbytes)	1024
File Size (Mbytes)	7.5, 15, 30, 60
Downlink BDP (Mbits)	120
Uplink BDP (Mbits)	1.2
Propagation Delay (sec)	40
Uplink Rate (kbps)	15
Downlink Rate (kbps)	1500
PER (%)	0, 5, 10, 15, 20, 25, 30, 35

4.2 Performance Metrics

In order to examine the tradeoff between the gain in data delivery latency and the waste of bandwidth imposed by redundant transmissions, we evaluate the performance of erasure coding against different performance metrics. Delivery latency is represented by two metrics: *Normalized Delivery Latency* (*NDL*) and *Normalized Delivery Latency Gain (NDLG). NDL* is the file delivery latency normalized based on the RTT. *NDLG* is the gain percentage-wise in file delivery time when the EC mechanism is applied, with respect to the simple ARQ-based mechanism.

Data redundancy imposed either by retransmitted packets or by FEC packets is evaluated using two metrics, *Normalized Redundancy (NR)* and *Normalized Redundancy Loss* (*NRL*). *NR* is the total number of redundant bytes normalized based on the BDP. When the simple ARQ-based mechanism is applied, *NR* accounts only for the retransmitted packets, while in the cases where the EC mechanism is applied, *NR* considers also the *n-k* redundant packets initially transmitted. *NRL* is the increase percentage-wise in *NR* when the EC mechanism is applied, with respect to the simple ARQ-based mechanism.

5 Experimental Results

Figures 1-3 present the experimental results for the first evaluation case, as described in Section 4.1. Since code rates used for the EC mechanism are optimal for each

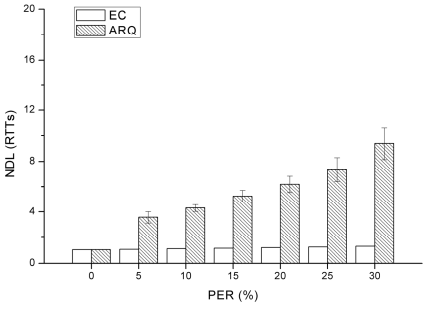
corresponding PER value, EC mechanism achieves in all cases successful file decoding and delivery after the first transmission round, thus avoiding retransmissions. In Fig. 1 we observe that, as PER increases, *NDL* for both schemes increases as well. This increase, however, is remarkably more significant when the ARQ-based mechanism is applied; the higher the PER, the higher the number of retransmission rounds required. On the contrary, when EC mechanism is applied, *NDL* is affected only by the additional delay required for the transmission of the encoding packets; the higher the PER, the higher the number n of the encoding packets (see Eq. 1).

As far as *NDL* is concerned, two major conclusions can be drawn. Firstly, we observe that as PER increases, the benefits of erasure coding, in terms of *NDL,* become more significant. Indeed, as shown in Figure 3, *NDLG* increases considerably as PER increases. The maximum observed gain in *NDL* is approximately 86%, when file size is equal to 0.5xBDP and PER is 30%. We further observe that the gain in *NDL* is significantly affected by the file size. For a given PER, as file size increases, *NDLG* decreases, respectively. For example, when PER is 30% and for file sizes equal to 1xBDP and 4xBDP, *NDLG* decreases from 0.82 to 0.6, respectively. This is explained by the fact that as file size increases, transmission latency comprises a significant portion of the total delivery time.

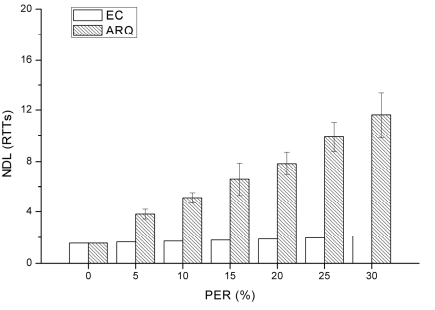
Even though EC reduces file delivery latency significantly, it necessarily imposes a redundancy overhead due to LDPC decoding inefficiency. In other words, although code rate is adjusted properly to protect file transmission for each given PER value, compared to Maximum Distance Separable (MDS) codes, LDPC codes require additional encoding packets for successful decoding. Data redundancy for both schemes and for different PER values is illustrated in Figures 2 and 3. *NR* for both mechanisms presents similar behavior for all file sizes and increases proportionally with file size. Therefore, we omit from Fig. 2 the corresponding graphs for the 1xBDP and 4xBDP file sizes. As expected, *NR* is significantly higher when EC mechanism is applied.

Although *NR* depends on file size, *NRL* does not; it depends only on PER and therefore it is depicted in Fig. 3 with a single line. As shown in Fig. 3, as PER increases, *NRL* decreases. That is, as PER increases, the contribution of LDPC inefficiency in total *NR* becomes, percentage-wise, less significant. For example, *NRL* is almost 0.6 when PER is 5% and decreases to around 0.27 for PER = 30%.

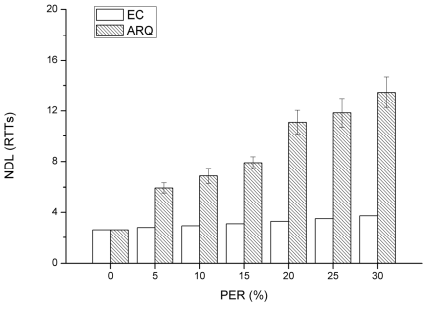
Fig. 3 gives a detailed insight about the tradeoff between the gain in file delivery latency and the waste of bandwidth due to the increased redundancy overhead. We observe that for low PERs (e.g., 5%) the gain in *NDL*, when erasure coding is applied, is comparable to the waste of bandwidth, percentage-wise. However, as PER increases, we observe that *NDLG* increases, while at the same time *NRL* decreases. Fig. 3 can constitute the basis for constructing a cost-based graph, where different cost weights can apply to each performance metric; the intersecting point between the two cost-weighted lines can indicate a performance threshold after which erasure coding, compared to typical ARQ-based solutions, is beneficial.



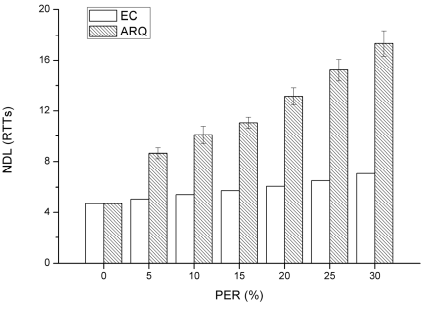
(a) File Size = 0.5 BDP



(b) File Size = 1 BDP

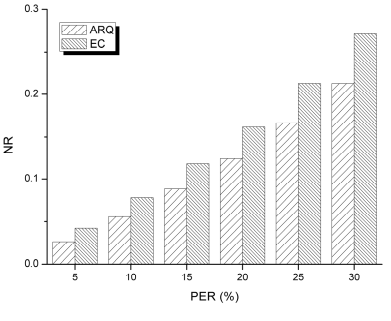


(c) File Size = 2 BDP

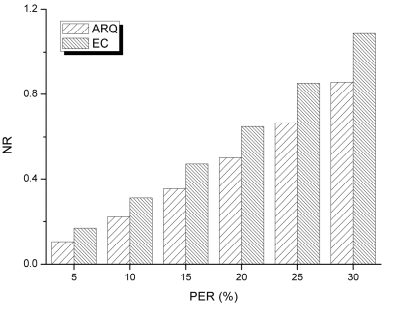


(d) File Size = 4 BDP

Fig. 1. Case 1 – Normalized Delivery Latency



(a) File Size = 0.5 BDP



(b) File size = 2 BDP

Fig. 2. Case 1 – Normalized Redundancy

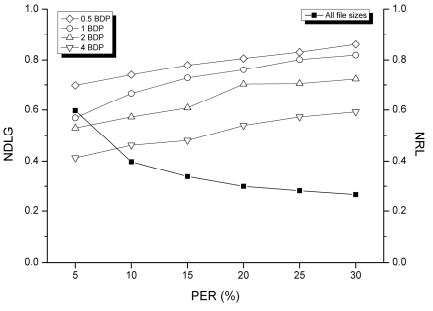
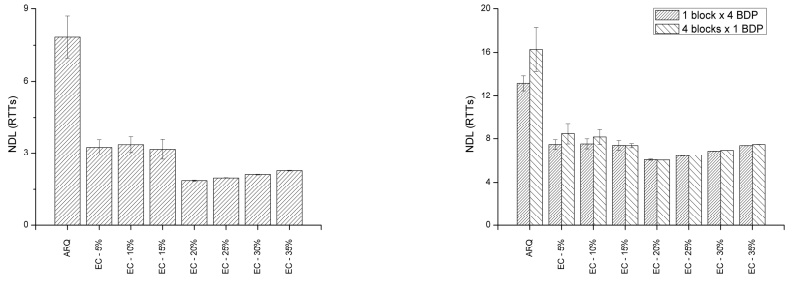


Fig. 3. Case 1 - NDLG vs NRL

In *Case 1* we have assumed that PER is known *a priori* and that code rate was adjusted accordingly (Eq. 1). This is, however, an ideal case and such knowledge is practically nonexistent. Thus, in the second evaluation case (*Case 2*), we investigate performance tradeoffs when predicted PER deviates from actual PER. Figures 4-6 present the evaluation results for *Case 2*.

In Fig. 4, *NDL* for different predicted PER values is shown and compared with the simple ARQ scheme. Two different file sizes are considered: 1xBDP and 4xBDP. In order to investigate in what extent file partitioning into blocks affects performance, we also consider the case where the 4xBDP file is partitioned into four equal-sized blocks.



(a) File size = 1 BDP

b) File size = 4 BDP

Fig. 4. Case 2 – Normalized Delivery Latency for different PER predictions. Actual PER = 20%.

As observed in Fig. 4, even when the predicted PER is lower than the actual, *NDL* decreases significantly when EC mechanism is applied. An interesting remark is that regardless of how low the predicted PER is, the reduction in *NDL* remains almost the same. As expected, the lowest *NDL* is obtained when the predicted PER matches the actual (20%). For higher predicted PERs, there is an increase in *NDL*. This is explained by the fact that when the code rate is lower than the theoretically optimal

(i.e., it targets on protecting higher PERs), decoding inefficiency increases and this affects *NDL*. The impact of such overestimation becomes more significant as file size increases. For example, when file size is 4xBDP and predicted PER is 35%, *NDLG* is comparable to the *NDLG* observed for PERs 5-15% (Fig. 6).

From a different point of view, assuming that actual PER is known *a priori*, we observe that erasure coding, when combined with ARQ-based schemes, can be used as a means to provide coarse-grained service differentiation among file transfers, without requiring any modifications in the underlying network. In particular, considering predicted PER as a configuration parameter, we observe in Fig. 4 that three different classes of service can be provided, in terms of delivery latency, by configuring predicted PER to 0%, 5% and 20%, respectively. Adjusting predicted PER to higher values (e.g., 10-15% and 25%-35%) results in similar *NDL* for the latter two classes, but it wastes bandwidth resources unnecessarily.

Furthermore, we note that the gain in *NDL* is significantly affected by the file size and, as already observed in *Case 1*, *NDLG* decreases when file size increases (Fig. 6). Regarding the multiple-block file transmission, results show that in this case the performance of the simple ARQ-based scheme is considerably affected (Fig. 4b). In particular, when file is segmented into multiple blocks, *NDL* increases by 3 RTTs. This is due to the fact that when multiple blocks are used, the number of *checkpoint* packets increases as well. As a consequence, more *checkpoint* packets are statistically lost, requiring retransmission and thus extending delivery latency. On the contrary, *NDL* achieved by the EC mechanism appears to be less affected by multiple-block transmission and only for lower predicted PERs (5% and 10%). Thus, it becomes clear that when file is partitioned into multiple blocks, the gain in *NDL* with EC mechanism is even higher. Fig. 6 verifies this conclusion.

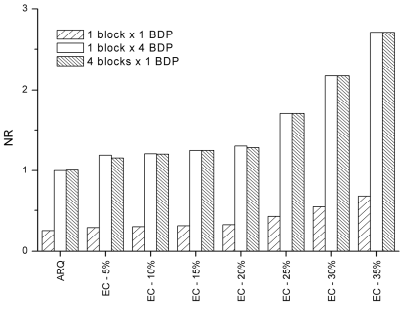


Fig. 5. Case 2 – Normalized Redundancy

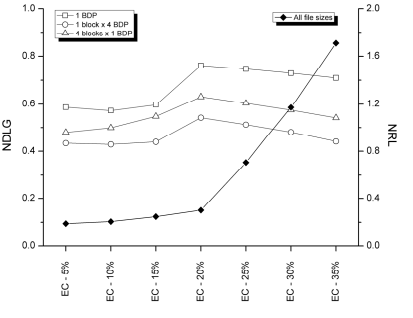


Fig. 6. Case 2 – NDLG vs NRL

Similarly to *Case 1, NR* presents the same behavior for all file sizes and increases proportionally to file size (Fig. 5). Besides, *NRL* is independent to the file size and depends only on the value of the predicted PER (Fig. 6). As shown in Fig. 5, *NR* subtly decreases when predicted PER gets lower than the actual (20%), and remains higher with respect to the simple ARQ-based mechanism. For higher values of predicted PER, however, *NR* is considerably affected. In particular, in Fig. 6 we see that when

PER prediction is 35%, *NRL* increases to 1.7, which can be interpreted as a 170% increase in redundancy, compared to the ARQ-based scheme. Multiple-block configuration imposes almost the same *NR* compared to single-block configuration. Finally, we note again that a cost-weighted graph based on Fig. 6 can be constructed in order to investigate the threshold after which erasure coding is beneficial, according to a cost function.

6 Conclusions – Future Work

In this work, we have examined the tradeoff between file delivery latency and redundancy overhead when erasure coding coupled with ARQ-based schemes is applied. An end-to-end transport protocol that was built on top of BP and which incorporates the EC mechanism introduced in this paper, was implemented and deployed into ESA DTN testbed, established in Space Internetworking Center. Results revealed significant insights on the performance tradeoffs imposed by erasure coding. Scenario-independent conclusions about when and to what extent erasure coding is beneficial in such environments were drawn. Additionally, results gave prominence to the capability of end-to-end erasure coding to administer QoS in terms of file delivery latency.

Although our proposed protocol solution comprises an end-to-end solution, in this initial work we have considered a single deep-space communication link. It is in our future intentions to broaden the analysis and investigate performance tradeoffs in multi-hop scenarios, where alternative paths may exist, communication links are characterized by varying propagation delays and PERs, and packet erasures are also caused by storage congestion. This analysis will further reveal the advantages and disadvantages of end-to-end approaches against hop-by-hop solutions. Finally, we intend to examine the applicability of adaptive erasure coding in space DTNs, where code rate is adjusted based on channel observations.

References

1. Rationale, Scenarios, and Requirements for DTN in Space. CCSDS Draft Green Book (March 2010)
2. Cerf, V., Burleigh, S., Hooke, A., Torgerson, L., Durst, R., Scott, K., Fall, K., Weiss, H.: Delay-Tolerant Networking Architecture. IETF RFC 4838, Informational (April 2007)
3. Rizzo, L.: Effective Erasure Codes for Reliable Computer Communication Protocols. ACM SIGCOMM Computer Communication Review 27, 24–36 (1997)
4. Scott, K., Burleigh, S.: Bundle Protocol Specification. IETF RFC 5050, experimental (November 2007)
5. Gallager, R.: Low Density Parity-Check Codes. MIT Press, Cambridge (1963)
6. Koutsogiannis, E., Diamantopoulos, S., Papastergiou, G., Komnios, I., Aggelis, A., Peccia, N.: Experiences from architecting a DTN Testbed. Journal of Internet Engineering 3(1) (2009)

7. Bezirgiannidis, N., Tsaoussidis, V.: Packet size and DTN transport service: Evaluation on a DTN Testbed. In: International Congress on Ultra Modern Telecommunications and Control Systems and Workshops, ICUMT, Moscow, pp. 1198–1205 (2010)
8. Cola, T.: Use of Erasure Codes in CCSDS Upper Layers: Motivation and Implementation. In: CCSDS Meeting, Portsmouth (May 2010)
9. Paolini, E., Varrella, M., Chiani, M., Calzolari, G.: Recovering from Packet Losses in CCSDS Links. In: IEEE Advanced Satellite Mobile Systems, ASMS, pp. 283–288 (2008)
10. Cola, T., Ernst, H., Marchese, M.: Performance analysis of CCSDS File Delivery Protocol and erasure coding techniques in deep space environments. Elsevier Computer Networks 51(14), 4032–4049 (2007)
11. Cola, T., Ernst, H., Marchese, M.: Application of Long Erasure Codes and ARQ Schemes for Achieving High Data Transfer Performance Over Long Delay Networks. Signals and Communication Technology 5, 643–656 (2008)
12. Cola, T.: A protocol design for incorporating erasure codes within CCSDS: The case of DTN protocol architecture. In: IEEE Advanced Satellite Multimedia Systems Conference, ASMA and The 11th Signal Processing for Space Communications Workshop, SPSC, pp. 68–73 (2010)
13. Fang, J., Akyildiz, I.: RCP-Planet: A Rate Control Protocol for InterPlaNetary Internet. International Journal of Satellite Communications and Networking 25(2), 167–194 (2007)
14. Kutscher, D., Loos, K., Greifenberg, J.: Uni-DTN: A DTN Convergence Layer Protocol for Unidirectional Transport. Work in progress as an internet-draft, draft-kutscher-dtnrg-uni-clayer-00 (2007)
15. Jet Propulsion Laboratory. Interplanetary Overlay Network, https://ion.ocp.ohiou.edu/
16. Roca, V., Neumann, C., Furodet, C.: Low Density Parity Check (LDPC) Staircase and Triangle Forward Error Correction (FEC) Schemes. IETF RMT Working Group, RFC 5170 (June 2008)
17. Roca, V., Neumann, C.: Design, Evaluation and Comparison of Four Large Block FEC Codecs, LDPC, LDGM, LDGM Staircase and LDGM Triangle, plus a Reed-Solomon Small Block FEC Codec. INRIA Research Report RR-5225 (June 2004)
18. Papastergiou, G., Psaras, I., Tsaoussidis, V.: Deep-Space Transport Protocol: A Novel Transport Scheme for Space DTNs. Computer Communications (COMCOM). Special Issue on Delay-/Disruption-Tolerant Networks 32(16), 1757–1767 (2009)
19. Space Internetworking Center, http://spice-center.org
20. Hemminger, S.: Network Emulation with NetEm. In: 6th Australia's National Linux Conference, LCA 2005, Canberra, Australia (April 2005)
21. European Space Agency, Mars Express, `http://www.esa.int/esaMI/Mars_Express/`
22. National Aeronautics Space Administration, Mars Exploration Program, `http://mars.jpl.nasa.gov/`
23. CCSDS File Delivery Protocol (CFDP). CCSDS Blue Book (January 2007)

On Passive Characterization of Aggregated Traffic in Wireless Networks

Anna Chaltseva and Evgeny Osipov

Department of Computer Science Electrical and Space Engineering
Luleå University of Technology,
971 87 Luleå, Sweden
{Anna.Chaltseva,Evgeny.Osipov}@ltu.se

Abstract. We present a practical measurement-based characterization of the aggregated traffic on microseconds time scale in wireless networks. The model allows estimating the channel utilization for the period of time required to transmit data structures of different sizes (short control frames and a data packet of the maximum size). The presented model opens a possibility to mitigate the effect of interferences in the network by optimizing the communication parameters of the MAC layer (e.g. the size of contention window, retransmission strategy, etc.) for the forthcoming transmission. The article discusses issues and challenges associated with the PHY-layer characterization of the network state.

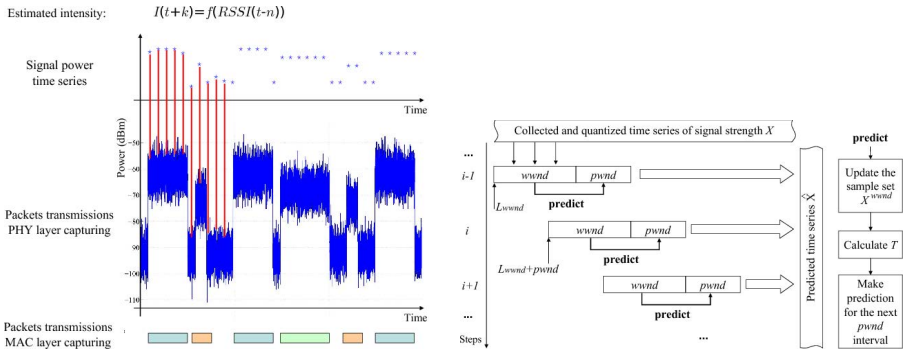
Keywords: Aggregated traffic, RSSI, modeling.

1 Introduction

Interference from external sources (noise) as well as interferences caused by distant communications on the same radio channel are the main reasons for the unstable performance in wireless networks in general and those built upon the IEEE 802.11 standard in particular. The "h" extension of the IEEE 802.11 standard [1] defines a *Dynamic Frequency Selection* (DFS) mechanism. The main idea of DFS is to reduce the interferences between wireless nodes by estimating the current utilization of the available channels based on RSSI (Received Signal Strength Indication) statistics and assuming that the estimated channel state will persist in a short-term future. In this article we present the results of a preliminary investigation of a possibility of using the statistics of the received signal strength not only to conclude about the channel utilization at the time of taking measurements but also predicting the channel utilization in the short-term future, further conceptualized in Figure 1(a) . If the approach is successful the predicted in this waychannel utilization could be used to adjust the parameters of the MAC layer (e.g. size of contention window, retransmission strategy, etc.), so to minimize the packet collision probability. This optimization process falls however outside the scope of this work and will be reported elsewhere.

Our major results are twofold. On the positive side we show that the statistics collected at the physical layer do not behave randomly and it is valid to use this information for characterization of the aggregated traffic in the vicinity of a wireless transmitter. For this purpose we propose a Markov-based model, which allows to predict the channel

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 282–289, 2012.



(a) Overall concept.

(b) The modeling process (see details in Section 3.3).

Fig. 1. Assessment of the quality and accuracy of the model

utilization on micro- and millisecond's time scale. While showing the feasibility of the micro-scale traffic characterization we conclude that more efforts should be spend to increase the accuracy of the prediction as well as developing mechanisms for using this information to improve the performance of next generation cognitive MAC protocols.

The most related to the topic of this article works are [2,3]. The authors in [2] use the autocorrelation function to predict the channel state ("free" or "busy"). In this work we show that the autocorrelation function cannot provide a conclusive picture in the case of mixed traffic under high load. In [3] the authors analytically model the instantaneous spectrum availability for a system with multiple channel using partially observable Markov decision process. This work presents decentralized cognitive MAC which allows the maximization of the overall network throughput. The results of our work could be considered in some extend as a practical compliment to the later approach since we build an empirical estimator of the instantaneous (plus several milliseconds in the future) channel state.

The article is organized as follows. Section 2 presents the research methodology. The passive estimation of the traffic intensity including the description of the experiments, data analysis, modeling, and the assessment of the accuracy is presented in Section 3, which is the main section of this article. Section 4 concludes the article.

2 Methodology

The main hypothesis of our work is that it is possible to derive a PHY-layer characterization of the aggregated traffic on a wireless link by statistical analysis of time series of the received signal strength. Our methodology for verification of the hypothesis consists of three phases: data gathering; randomness and correlation analysis; and modeling and assessment.

Data gathering: All data for further analysis and modeling were obtained in a controllable manner in a radio isolated chamber. We experimented with traffic of different intensities and used a spectrum analyzer to accurately record the signal strength time

series with microsecond's sampling time. The detailed description of the experiments follows in Section 3.1.

Randomness and correlation analysis: In this phase we firstly examine a statistical dependence in the recorded time series. In other words whether we can use the physical layer's statistics for characterization of the channel utilization. The results of the two-sample Kolmogorov-Smirnov test (presented in Section 3.2) allowed us to proceed with the analysis of nature of the statistical dependence by studying the correlation structure of the series described in the same section.

Modeling and assessment: Finally, we build a two-state Markov model of the channel occupancy and use it to estimate channel utilization in time domain during a time interval chosen with reference to the transmission time of data structures of different length (e.g. short control frames and maximum size of a data packet). The rationale for doing this step is simple, if we are able to correctly predict the channel utilization on packet -ransmission time scale we may further use this result to optimize the transmissions of the pending packets.

3 Passive Estimation of Aggregated Traffic Intensity Using PHY-Layer Statistics

In this section we develop our hypothesis of deriving PHY-layer characterization of the aggregated traffic. The subsections below describe the details of data gathering, randomness and correlation analysis as well as present the constructed model and the results of its accuracy assessment.

3.1 Test-Bed Experiments and Data Gathering

The time series of the received signal strength were measured during a set of experiments performed on a wireless test-bed network located inside an isolated 6×3 meters chamber. The walls of the chamber are non-reflecting surfaces preventing multipath propagation. The wireless test-bed consists of four computers equipped with IEEE 802.11abgn interfaces, located in the transmission range of each other. All computers are running Linux operating system (kernel 2.6.32). The transmitted signal power was set to 18 dBm, the testbed operated on channel 4 (2427 MHz). On the MAC layer the Maximum Contention Window is 1023, the short slot time is 9 us, SIFS interval is 10 us, and the short preamble is 72 bits.

The received signal strength time series were recorded using spectrum analyzer Agilent E4440A. The recorded raw signal was sampled with 1MHz frequency. Later during the analysis phase we increased the sampling interval by trimming out the original set. We quantized the recorded signals into two levels. All samples with the signal power less than -87 dBm (the received sensitivity of the used wireless adapter) were assigned a value of 0 (zero). All measurements above this threshold were assigned a value of 1.

Traffic Flows: In total 13 experiments with one, two, three, and four concurrent data sessions were performed. For further discussions we sort all experiments into three groups depending on the aggregated load (low, medium, and high).

The *low traffic* was generated by single UDP or TCP flows, the *medium traffic* was generated by two and three concurrently running UDP and TCP flows in different combinations, the *high traffic* was generated by four concurrently running UDP and TCP flows in different combinations. In all cases nodes were configured with static routing information in order to eliminate the disturbance caused by the routing traffic. In all experiments the payload size was chosen so to fit the maximum transfer unit of 1460 Bytes. In the case of UDP traffic we experimented with two traffic generation rates: 100 Kb/s and 11 Mb/s, to study both the unsaturated and saturated cases. The duration of each experiment was 10 seconds. To remove transient effects, only the last 2.5 seconds of the recorded signal series were used for the analysis.

3.2 Randomness and Correlation Analysis

Denote $X^C = \{x_i^C\}$ the recorded continuous time series. Let $X^R = \{x_i^R\}$ denote a reference random time series obtained by randomly shuffling the original set X^C. In order to verify whether there is a statistical dependency in the original time series we performed the analysis of increments dependence [4]. The increments of time series were obtained for both X^C and X^R as $\Delta x_i^C = x_i^C - x_{i-1}^C$ and $\Delta x_i^R = x_i^R - x_{i-1}^R$ correspondingly.

In order to check the hypothesis that the recorded time series of the signal strength have a statistical dependency, the two-sample Kolmogorov-Smirnov test [5] was performed. This test compares the distributions of the values in the two given sets (the original and the reference time series). The null hypothesis is that the sets are from the same continuous distributions, accordingly, the alternative hypothesis is that the sets are from different distributions. The outcome of Kolmogorov-Smirnov allows the rejection of the null hypothesis with 1% significance level, i.e. there is a statistical dependency in the recorded time series. This conclusion allows us to proceed with the analysis.

The analysis of the autocorrelation function for the time series showed very rapid decay of curves, which indicates that all observed processes have short-range dependence. The graphs are omitted here due to the limited space, the reader is referred to [6].

3.3 Modeling and Assessment

Our approach towards modeling is illustrated in Figure 1(b). A two-state Markov model is constructed using data collected during time interval called *working window* and denoted as *wwnd*. During the course of this work we experimented with different durations of $wwnd$. It appeared that the size of $wwnd$ does not significantly affect the accuracy of the prediction. The results presented in this article are obtained using $wwnd = 0.5$s. Denote $X = \{x_i\}, x_i \in [0; 1]$ the post processed and quantized time series of X^C (See Section 3.1).

The constructed model is then used to predict the presence or absence of the signal during the immediately following time interval called *prediction window* and denoted as *pwnd*. The size of $pwnd$ is chosen with a reference to the time of transmitting a data structure of certain length with a given transmission rate on the physical layer. We choose two values of $pwnd$ in order to illustrate our reasoning: one equals the time it takes to transmit the shortest data structure (RTS frame) with the rate 1Mb/s:

$pwnd = 200$ microseconds. The other value equals the time it takes to transmit the maximum size packet (1460 Bytes) with the highest transmission rate 11Mb/s in our case: $pwnd = 1.5$ milliseconds. The rationale for choosing these values stems from the goal of this work - we want to optimize the performance of the MAC protocol prior of the transmission of a pending packet.

Two-State Markov Model Over $wwnd$: Denote X^{wwnd} a subset of the measured and quantized time series of the received signal strength X of size $wwnd$ expressed in number of samples. Then x_i^{wwnd} denotes the measured and quantized signal strength at sample time i. The Markov model describes the state of the channel at a particular sampling step $i+1$ based on the current state at the step i. The model is defined by a transition probability matrix T as follows:

$$T = \begin{pmatrix} P(x_{i+1}^{wwnd} = 0 | x_i^{wwnd} = 0) & P(x_{i+1}^{wwnd} = 1 | x_i^{wwnd} = 0) \\ P(x_{i+1}^{wwnd} = 0 | x_i^{wwnd} = 1) & P(x_{i+1}^{wwnd} = 1 | x_i^{wwnd} = 1) \end{pmatrix}$$

where P is an empirical conditional probability calculated over $wwnd$ number of samples.

When matrix T is calculated and the prediction of the channel utilization (as described below) is done we shift the working window on the set of original time series X to $pwnd$ samples in the direction of time increase. This moves us to the next iteration of the modeling and prediction process, which is summarized in Figure 1(b).

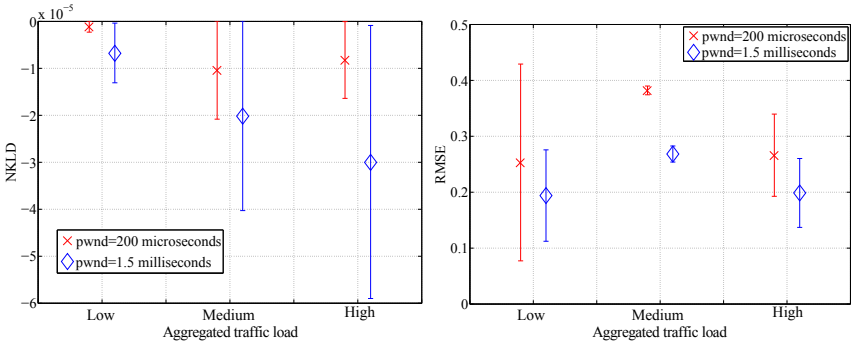
Prediction Procedure Over $pwnd$: The goal of the prediction process is to generate time series X^{pwnd} of the predicted signal presence. Thus $x_i^{pwnd} = 1$ indicates the presence of the signal above the receiver sensitivity threshold while the value of 0 indicates an absence of the signal at sample time i. The probabilities that during ith position of $pwnd$ there will be transmission or not are taken from the matrix T depending on the channel state at time $i-1$. After the probabilities are determined for position i of $pwnd$ we generate an actual value (1 or 0) using conventional technique for generating random numbers from a given distribution. This procedure is then repeated for all positions inside the $pwnd$.

3.4 Assessment of the Quality and Accuracy of the Model

The quality of the model was evaluated by the analysis of the model's performance using normalized Kullback-Leibler divergence. The Kullback-Leibler divergence is a non-symmetric measure of the distance or the relative entropy between two probability distributions $Pr[X]$ and $Pr[\hat{X}]$ [7]. This statistical metric (1) is used to measure how the distribution of the set produced by a stochastic model ($Pr[\hat{X}]$) is different from the distribution of the original stochastic process $Pr[X]$.

$$D_{KL}(Pr[X]||Pr[\hat{X}]) = \sum_i Pr[X]_i * log \frac{Pr[X]_i}{Pr[\hat{X}]_i} \quad (1)$$

The smaller is the value of $D_{KL}(Pr[X]||Pr[\hat{X}])$ the closer the distributions $Pr[X]$ and $Pr[\hat{X}]$ are. In the case when $D_{KL}(Pr[X]||Pr[\hat{X}]) = 0$ the two distributions are identical. To calculate the normalized Kullback-Leibler distance the following formula



(a) NKLD for the proposed model. (b) The proposed model accuracy.

Fig. 2. Assessment of the quality and accuracy of the model

was used: $\bar{D}_{KL}(Pr[X]||Pr[\hat{X}]) = \frac{D_{KL}(Pr[X]||Pr[\hat{X}])}{H(Pr[X])}$ where $H(Pr[X])$ is the entropy of a random variable with the probability mass function $Pr[X]$. $H(Pr[X]) = \sum_i Pr[X]_i * log\frac{1}{Pr[X]_i}$.

Figure 2(a) shows the correspondent graphs of $\bar{D}_{KL}$ for the proposed model. We conclude that the model has satisfactory quality since the distance between the probability distributions of the measured time series and the predicted ones is in the oder 10^{-5}.

Model Accuracy: The accuracy of the model was evaluated with respect to its ability to predict the channel utilization over one $pwnd$ interval, denoted as ξ^{pwnd} (2). The results are presented in Figure 2(b).

$$\xi_j^{pwnd} = \frac{\sum_{i=1}^{pwnd} x_i^{pwnd}}{pwnd} \tag{2}$$

The predicted utilization over one $pwnd$ interval is denoted as $\hat{\xi}_j^{pwnd} = \frac{\sum_{i=1}^{pwnd} \hat{x}_i^{pwnd}}{pwnd}$, where $j \in [1, N]$ and N is the number of $pwnd$ intervals in X and $\hat{X}$.

As the result of calculation of ξ and $\hat{\xi}$ over the original and predicted time series we obtain two sets of utilization $\Xi = \{\xi_j^{pwnd}\}$ and $\hat{\Xi} = \{\hat{\xi}_j^{pwnd}\}$ of measured and predicted utilizations on $pwnd$ chunks of the time series X and $\hat{X}$ correspondingly.

We use the root-mean-square error metric to assess the differences between Ξ and $\hat{\Xi}$ (3).

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(\xi_i - \hat{\xi}_i)^2}{N}} \tag{3}$$

Figure 2(b) illustrates the accuracy of the model for different aggregated traffic loads and different values of $pwnd$. The plot is obtained by assessing the model's accuracy

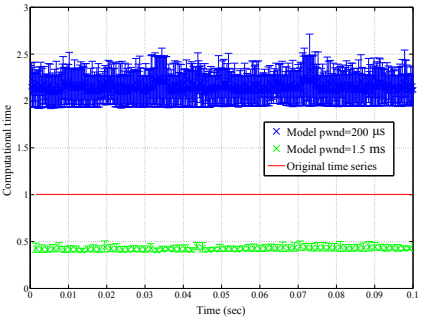


Fig. 3. The computation time of the model

using three different initial positions for $wwnd$ and correspondingly $pwnd$ in the original set of time series X. From Figure 3 we observe that the accuracy of the model is substantially lower for the short $pwnd$ (200 microseconds). Although for some parts of the traces with low traffic intensity the model introduced 10% error, the average error for all traffic loads ranges between 0.25 and 0.4. On the other hand for the larger $pwnd$ the average value never exceeds 0.3 for all traffic loads. In particular in the case of high traffic load our models shows 0.2 prediction error.

Computation Time: In order to assess the computation time of the model we timed the execution of operations for constructing the transition matrix and the prediction procedure for different values of $pwnd$. The time measurements were performed on Lenovo ThinkPad T61 computer with Intel T7300 Core 2 Duo processor, 2GB RAM and running Ubuntu 10.04 LTS operating system. Figure 3 plots the results of the measurements normalized to the duration of corresponding $pwnd$. From the figure one could immediately observe that the choice of $pwnd$ size is essential. The computation time of the model is almost twice higher than the duration of the smallest $pwnd$ (200 microseconds) . On the other hand it is twice less than the duration of the larger $pwnd$ (1.5 milliseconds).

4 Conclusions

In this article we presented a practical measurement-based model of the channel utilization on microsecond's time scale for wireless networks. The model allows estimating the utilization for the period of time required to transmit data structures of different sizes (short control frames and a data packet of the maximum size). The resulting model opens a possibility to mitigate the effect of interferences in the network by optimizing the parameters of the MAC layer for the forthcoming transmission based on the predicted channel utilization based on short-term historical data. The presented model is based on the collected statistic in the wireless test-bed network located inside an isolated chamber and there is clearly a need in additional experimental work in order to validate the model applicability and accuracy in real settings.

Our major conclusion is twofold. Firstly, more efforts should be spend to increase the accuracy of prediction by using more sophisticated models as well as choosing the appropriate dimensions of the working and prediction windows. Here one should make a trade-off between the prediction accuracy and the computation time of the model. Secondly, we foresee that on micro- or millisecond's time scale even the best models would introduce significant error to the predicted channel utilization. It is unrealistic to expect that aggregated traffic could be very accurately characterized solely based on samples of radio signal. One, howeve, still may use this information in more sophisticated cross-layer decision mechanisms. Further development of these issues is a subject for our ongoing and future investigations.

References

1. IEEE standard for information technology - telecommunications and information exchange between systems - local and metropolitan networks - specific requirements - part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications - spectrum and transmit power management extensions in the 5 ghz band in europe (2003)
2. Mangold, S., Zhong, Z.: Spectrum agile radio: Detecting spectrum opportunities. In: International Symposium on Advanced Radio Technologies, ISART, Boulder CO, USA, p. 5 (March 2004), http://www.comnets.rwth-aachen.de
3. Zhao, Q., Tong, L., Swami, A.: Decentralized cognitive mac for dynamic spectrum access. In: 2005 First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, DySPAN 2005, pp. 224–232 (November 2005)
4. Kovalevskii, A.: Dependence of increment in time series via large deviations. In: Proceedings of the 7th Korea-Russia International Symposium on Science and Technology, KORUS 2003, vol. 3, pp. 262–267 (July 2003)
5. Stephens, M.A.: Use of the kolmogorov-smirnov, cramer-von mises and related statistics without extensive tables. Journal of the Royal Statistical Society. Series B (Methodological) 32(1), 115–122 (1970), http://www.jstor.org/stable/2984408
6. Chaltseva, A.: Network state estimation in wireless multihop networks, licentiate thesis (2012), http://pure.ltu.se/portal/files/35972873/Anna_C._Jaquier.Komplett.pdf
7. Cover, T., Thomas, J.: Elements of information theory. Wiley Series in Telecommunications and Signal Processing. Wiley-Interscience (2006), http://books.google.com/books?id=EuhBluW31hsC

TCP Initial Window: A Study

Runa Barik and Dinil Mon Divakaran

School of Computing and Electrical Engineering
Indian Institute of Technology Mandi
Mandi-175001, India
runa@students.iitmandi.ac.in, dinil@iitmandi.ac.in

Abstract. The research community has for sometime argued the need to increase the size of TCP initial-window (IW). While it is probably high time that IW-size is increased, we note that a comprehensive study on this front is missing. In this paper, we attempt to build that gap, analyzing the affects increasing IW on various important parameters correlated to the performance of flows. In particular, given the mice-elephant phenomenon, we focus on how the response times of small TCP flows are affected with increasing IW. Our study reveals that, it is difficult to set one single value for IW that improves various parameters such as number of time-outs, retransmission rate, number of spurious time-outs, completion times of flows, etc. Instead, we propose to use a simple function to set the value of IW of a flow depending on the flow-size. Through simulations, we show that such a function performs better.

Keywords: TCP, IW, RTO, response time.

1 Introduction

Nearly 90% of the Internet traffic volume is carried by TCP, the transmission control protocol [3]. In this context, the performance attained by TCP flows is recognized as important. Among the various TCP parameters that have been studied for performance improvement, one important parameter is the value of initial-window (IW). Researchers have for sometime argued the need to increase the IW-size. In RFC 3390 [1], the upper bound for the IW-size is set to 'min (4*MSS, max (2*MSS, 4380 bytes))' which corresponds to three times the maximum segment size (MSS) in Ethernet LANs.

The relevance of study on TCP's IW is gaining importance, in the light of strong heavy-tail behavior of Internet traffic — a small percentage of flows (large in size) contribute to a large percentage of the Internet's traffic volume. It is also referred to as the *mice-elephant* phenomenon, where 80% of the flows that contribute to 20% of the traffic are called mice (small) flows, and the remaining 20%, the elephant (large) flows. It can be argued that the current TCP/IP architecture is biased against small TCP flows, for the following important reasons. (i) As small flows do not have much data, they almost always complete in the slow-start phase, never reaching the congestion-avoidance phase; and thus typically having a small throughput. (ii) A packet-loss to a small flow most often results in a time-out due to the small congestion window (*cwnd*) size; and time-outs increase the completion time of small flow many folds. (iii) The

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 290–297, 2012.

increase in round-trip-time (RTT) due to large queueing delays hurts the small flows more than the large flows. For the large flows, the large *cwnd* makes up for the increase in RTT; whereas, this is not the case for small flows and is quite perceivable.

The biases against mice flows become more relevant today, with recent studies showing an increase in the mice-elephant phenomenon, with a stronger shift towards a 90-10 rule [3]. Most solutions of this problem can be grouped into either router-centric solutions which include priority-scheduling algorithms and buffer management policies to give priority to small flows both in time and space respectively, or end-host-based solutions which include mostly TCP variants and changing IW. *Size-based scheduling* is a type of priority scheduling that gives priority to packets of flows, based on flow-size. *LAS* (Least Attained Service) and *PS+PS* scheduling algorithms improve the mean response time of small TCP flows [9,2]. Threshold-based-sampling-cum-scheduling policy [5] is a practical approach to size-based scheduling, where large flows are detected probabilistically. Similarly *Spike-detection* method is another way of detecting and *deprioitizing* large flows, for improving the response times of small flows [4].

In T/TCP, the sender starts transmitting data packet along with the first segment (SYN packet) and thus saves response times of transactional services [7]. Quick-Start TCP determines the 'initial throughput' for TCP flows by co-operating with the routers on the network path and does not require per-flow state information in the routers [10]. TCP/SPAND improves the response time of small flows by avoiding slow-start penalty and the optimal IW depends on flow-size and network state information [11].

Google proposed to increase the IW to at least 10 segments, and proposed for standardization by the IETF [6]. They quantified the response times using large IW, that depends on network bandwidth, round-trip time, and nature of applications. Authors of [8] proposed an approach for setting the IW-size in Fast Startup TCP to improve the initial throughput of TCP flows using the bottleneck link and access link bandwidth, and the bottleneck router's buffer size.

One of the main reasons for the biases (against small flows) is the small value of TCP's congestion window (*cwnd*). The *cwnd* for a TCP flow in time t during slow-start phase is given by,

$$cwnd = IW \times \left(2^{\lfloor \frac{t}{RTT} \rfloor}\right),$$

where IW is the IW-size of a TCP flow, RTT is the round-trip-time. As *cwnd* depends on IW-size, IW can be considered as a factor that affects the response times of the small TCP flows. The response times of most of the TCP small flows end within slow-start stage, unless until they face loss in the network. The response time, t, in slow-start without losses (in the ideal case) is [6]:

$$t = \left\lceil \log_2 \left(\frac{S}{IW} + 1 \right) \right\rceil \times RTT_b + \frac{S}{C} + t_q, \tag{1}$$

where S is flow-size, C is the bottleneck-link capacity (all in units of packets). RTT_b is assumed a constant and is equal to twice the propagation delay. t_q is the queueing delay faced by a flow. It is assumed that queueing delay neither causes reordering of packets nor triggers spurious retransmissions. From the Eq. 1, we deduct that for the improvement in response time of flows, IW may depend on flow-sizes.

We focus on how the response times of small TCP flows are affected with increasing IW. Our studies are carried out using simulations, where we consider for performance evaluation, parameters which are adversely affected by IW-size and they are:

1. Mean completion time[1] ($\overline{CT}$) for small, large and all flows conditioned on flow-size. Completion time is the time between sending of first SYN packet to getting the ACK for the last packet for a flow. Conditional mean completion time, which we also use, is the mean completion time conditioned on the flow-size.
2. Number of TCP spurious time-outs (ST) encountered by small and large flows.
3. Number of TCP retransmission time-outs (RT) encountered by small and large flows.
4. Retransmission rate (RR): This represents the percentage of packets lost due to congestion. It is defined as the ratio of number of retransmitted packets to the total number of flows, expressed as packets per flow.
5. Mean completion time for range of flow-sizes.

Empirically, we find that no single value of IW-size gives optimal performance. Hence, instead of having a single IW-size for all flows, we show that a simple flow-size based function to choose an IW-size can give better performance.

The rest of the paper is organized as follows. In Section 2, we conduct simulation-based studies to understand the affects of IW-size on various parameters. In Section 3, we propose a size-based function for IW, and evaluate and compare the performance attained against size-independent IW values. We summarize the paper in Section 4.

2 Analysis Using Simulation

2.1 Settings

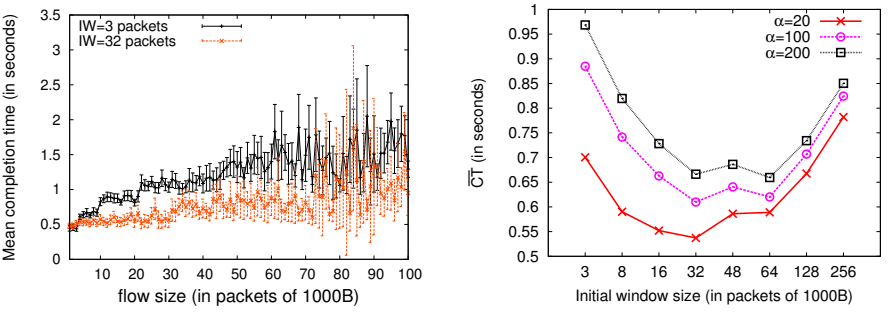
Simulations are carried out in ns-2. A dumbbell topology, representing a single bottleneck-link with capacity of 1 Gbps, connecting source-destination pairs with link capacities of 100 Mbps, was used throughout. The delays on the links are set such that base RTT (consisting of only propagation delay) is equal to 100 ms. The buffer-size of bottleneck-link of data path is set to the bandwidth-delay-product (1 Gbps $\times$ 100 ms). Drop-tail buffers are used at all nodes. The total number of flows is 20,000. 15% of flows are generated using Pareto distribution with shape parameter α as 1.1 and a mean flow-size of 1 MB, and remaining flows are generated using Exponential distribution with a mean flow-size of 20 KB. All flows are carried by TCP using the SACK version with timestamps options set. The packet-size is assumed constant, equal to 1 KB.

As the flows with size less than or equal to 60 KB constitute 80% of the total number of generated-flows, any flow with flow-size less than or equal to 60 KB is considered to be a 'small flow' and with size greater than 60 KB is considered to be a 'large flow'.

2.2 Results

Number of Retransmission Time-outs: Table 1 shows the RT for different IW-sizes, faced by all flows. RT_s and RT_l represent the RT for small and large flows

[1] We often use 'completion time' to refer to 'response time'.



(a) 95% confidence interval for $\overline{CT}$ (b) $\overline{CT_s}$ for different values of α

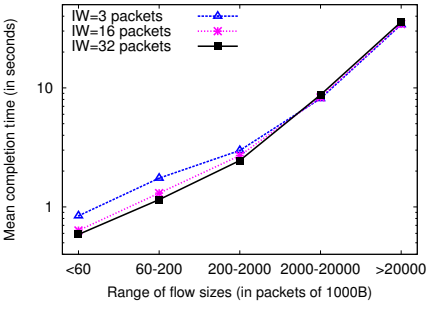
Fig. 1.

respectively. As expected, with increasing IW, *RT* increases. Increasing IW-size for both small and large flows increases the burstiness of TCP traffic. Burstiness can cause router buffer to overflow and that results in large number of packet-drops. Due to large number of packet-drops, either small flows may not able to produce duplicate ACKs or retransmitted packet gets dropped, may cause retransmission timeouts.

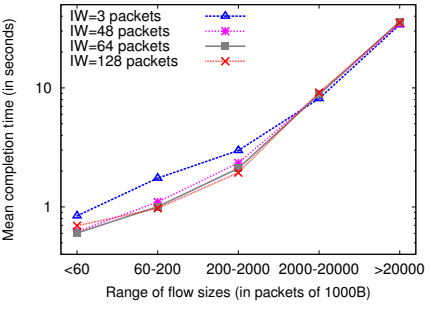
Retransmission Rate: Table 1 shows the retransmission rate for small (RR_s) and large flows (RR_l) with respect to IW-size. It depicts that retransmission rate increases by increasing IW. This may be due to fast retransmissions, retransmission timeouts and spurious retransmissions. Note that, RR_l is considerably high in comparison to RR_s.

Number of Spurious Time-outs: Table 1 also shows the number of spurious time-outs faced by small (ST_s) and large (ST_s) flows for varying IW-size. The number of spurious time-outs faced by all flows decreases with increasing IW-size. This may be due to the fact that large *cwnd* may result in small variation of round-trip-delays.

Mean Completion Time: The last two columns of Table 1 show the mean completion time averaged for small flows ($\overline{CT}_s$), large flows ($\overline{CT}_l$). Large flows show decreasing completion time for increasing IW, as they take small number of rounds. The response times of small flows decrease by almost 300 ms, when IW-size is increased from 3 to 32 packets. Beyond IW-size of 32, there is an increase in response times for small flows, possibly due to increasing packet-losses (with increasing value of IW beyond a point). Fig. 1(a) plots the mean completion time for IW of 3 and 32 packets, for 95% confidence interval. Reduction in response time is clear for flows with sizes less than (around) 70 packets. The gain in response time reduces with increasing size. Figures 2(a) and 2(b) plot the mean completion times of flows within different flow-size ranges. The improvement with increasing IW-size decreases as the flow-size increases. The figure also shows that *medium-size* flows have better improvement in mean completion time with large IW-size.



(a) For IW-sizes of 3, 16, 32 packets



(b) For IW-sizes of 3, 48, 64 and 128 packets

Fig. 2. Mean completion time for ranges of flow-sizes

Table 1. Comparison of different parameters

IW	RT_s	RT_l	RR_s	RR_l	ST_s	ST_l	$\overline{CT_s}$	$\overline{CT_l}$
3	1475	618	0.4495	6.6758	24	60	0.8424	2.3263
8	1346	650	0.5361	8.8865	68	115	0.7048	2.1601
16	1442	649	0.6247	9.1910	48	67	0.6342	1.9521
32	1465	733	0.6720	9.4985	5	29	0.5875	1.7848
48	1713	818	0.7609	11.5080	2	26	0.6191	1.7154
64	1627	724	0.7407	10.6737	2	29	0.6033	1.5851
128	2327	1041	1.0450	15.9948	4	20	0.6953	1.5185
256	3186	1357	1.3740	21.4348	7	17	0.8155	1.5785

Algorithm 1. IW(x, σ, β, γ)

1: **if** $x <= 200$ **then**
2: $IW \leftarrow random\_select(\beta, \gamma)$
3: **else**
4: $IW \leftarrow \sigma$
5: **end if**

Next, we use α as a threshold to define small flows (and thereby differentiate between small and large flows). Fig. 1(b) shows $\overline{CT_s}$, the average of mean completion time for small flows, for different values of α. Similar plot for mean completion time $\overline{CT_l}$ averaged over large flows shows also similar trend for different values of α.

Though the response times of small and large flows are reduced with increasing IW, we see that this is not monotonous. Increasing IW-size for small flows increases the *cwnd* size. If small flows with large *cwnd* size face packet-losses, then the loss can be recovered by fast retransmission instead of timeouts. IW-size larger than flow-size does not increase the burstiness of small flows. But higher burstiness due to large flows cause large number of packet-drops at the bottleneck router which increases the response times of small flows as well as large flows. As large flows spend most of their lifetimes in congestion-avoidance phase, large IW-size has negligible effect on large flows. Empirically, we find that no single value of IW-size gives optimal performance, when compared using various perfomance evaluation parameters.

3 IW as a Function of TCP Flow-Size

From the previous analyzes, we can infer that IW-size for small flows needs to be increased more rather than that for large flows. As it is difficult to get a closed form equation for finding optimal choice for IW-size for a given set of network parameters

Table 2. Comparison of different parameters

IW	RT_s	RT_m	RT_l	RR_s	RR_m	RR_l	$\overline{CT_s}$	$\overline{CT_m}$	$\overline{CT_l}$
3	1475	391	227	0.45	2.75	14.66	0.842	1.744	3.511
32	1465	427	306	0.67	4.36	19.95	0.587	1.150	3.075
IW(x,3,3,64)	1030	237	291	0.43	3.01	17.78	0.557	1.035	3.488
IW(x,10,3,64)	1174	285	234	0.56	3.36	17.73	0.570	1.081	3.234
IW(x,3,32,64)	892	168	249	0.43	2.63	13.55	0.496	0.824	3.434
IW(x,3,48,64)	1091	221	275	0.49	3.41	16.15	0.525	0.855	3.19
IW(x,3,32,128)	1006	225	265	0.507	3.5	15.84	0.505	0.7782	3.008
IW(x,3,32,256)	1140	238	411	0.53	3.40	19.76	0.523	0.719	3.28

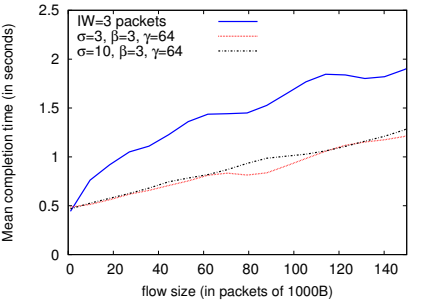


Fig. 3. For varying values of σ

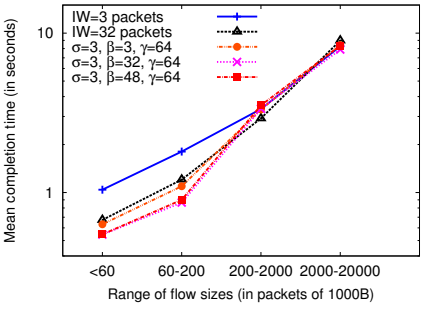
under varying number of TCP sources with different RTTs, we take a simple function for determining IW based on flow-size.

In Algorithm 1, we use x to represent flow-size. The flows with size ≤ 200 packets constitute both small and medium-size flows. σ represents the value of IW-size for large flows. The IW-size for flows with size ≤ 200 packets is uniformly distributed between β and γ. All three — σ, β and γ are in packets.

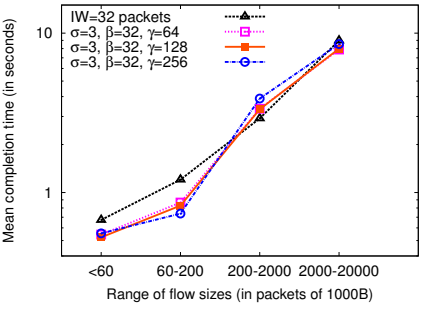
We perform simulations with same the settings as described in Sec. 2.1 for the function IW(x, σ, β, γ), with α set to 60 KB. We define 'medium-size flow' as a flow with size greater than 60 KB and less than or equal to 200 KB. All parameters in Table 2 with a subscript of m are for medium-size flows. In the following, we keep two parameters constant and vary the third one (where σ, β and γ being the parameters).

Varying Values of σ: Here, we set β and γ to (say) 3 and 64 respectively. Fig. 3 plots the mean completion time for flow-sizes less than or equal to 150 packets. As seen, the reduction in mean completion time is high the values for β and γ; while the the performance gain to small flows is insignificant for different values of σ. Between the values of 3 and 10 for σ, the former is better when we observe and compare other parameters RT_s, RR_s, and $\overline{CT_s}$, listed in Table 2. Also observe that both small and medium-size flows show better improvement in response times (reduction of nearly 300 and 700 ms respectively) with σ set to 3 against IW-size of 3. By setting σ to 10 instead of 3, no significant improvement is observed for large flows, whereas, both small and medium flows are affected and get degraded performance. So, we set σ to 3.

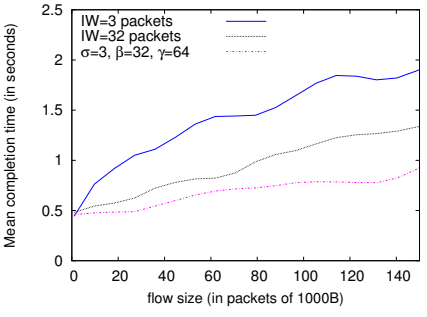
Varying Values of β: Here, we set σ and γ to 3 and (say) 64 respectively. Fig. 4(a) plots the mean completion time for range of flow-sizes. As seen, both small and medium-size flows improve the response times with $\beta = 32$ in comparison to single IW value of 3 or 32. The improvement is insignificant for large flows. Between the values of 3 and 48 for β, the value 32 is better when we observe and compare other parameters RT_s, RR_s, and $\overline{CT_s}$, listed in Table 2. The improvement in response time for both small and large flows with $\beta = 32$ is nearly 60 and 300 ms respectively against $\beta = 3$. The improvement in response time is negligible for β value of 48. Hence, we set β to 32.



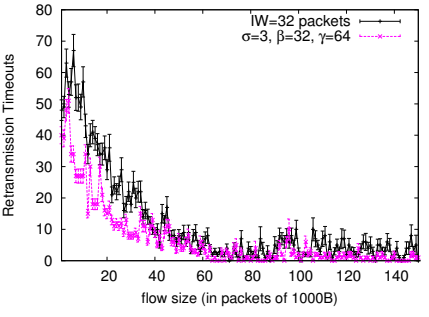
(a) For varying values of β



(b) For varying values of γ



(c) IW(x, 3, 32, 64) v/s IW = 3, 32



(d) Retransmission timeouts

Fig. 4. Performance parameters

Varying Values of γ: Here, we set σ and β to 3 and 32 respectively. Fig. 4(b) plots the mean completion time for range of flow-sizes. As seen, small flows get reduced mean completion time using the IW$(x, 3, 32, 64)$ in comparison to IW-size set to 32 packets. By setting γ to a value more than 64, no significant improvement is observed for small flows, compared to γ set to 64. Also observe in the Table 2, the performance parameters like RT, RR, and $\overline{CT_s}$ increase for values of γ more than 64. So, we set γ to 64.

Fig. 4(c) plots the mean completion time for flow-sizes less than or equal to 150 packets. The figure shows that with the IW$(x, 3, 32, 64)$, flows significantly improve the response times compared to the IW-size set to 3. Medium-size flows show a better improvement in response times using IW$(x, 3, 32, 64)$ compared to IW-size of 32 packets. Table 2 lists other parameters using IW-size function and compares with IW-size of 32 packets. IW$(x, 3, 32, 64)$ shows lesser number of RT compared to IW-size of 32 packets. IW$(x, 3, 32, 64)$ shows a significant improvement in RR compared to IW-size of 32 packets and also improves the response times by nearly 100 and 300 ms for both small and medium-size flows respectively. Fig. 4(d) plots the total number of retransmission time-outs faced by flows for a given flow-size, for 95% confidence interval. It shows that the RT using IW-size function gets reduced for flows with flow-size less than nearly 60 packets and the decrement in RT reduces with increasing flow-sizes.

Both small and medium-size flows showed a significant improvement in response times using flow-size based function for choosing IW-size. We find that the flow-size

based function to choose an IW-size gives better performance rather than having a single value of IW-size for all flows, when compared using various evaluation parameters.

4 Conclusions and Future Work

In this work, we studied the affects of different values of IW on the performance of flows, in particular small flows. Our performance evaluation compared and studied various important parameters. We saw that the performance of flows do not monotonically improve with increasing IW-size. Coming up with a single value of IW for all flows might be difficult, as this depends on a number of network parameters. Instead, we demonstrated that a simple flow-size based function for choosing an IW-size improved the response time of small TCP flows, thereby meeting the intension behind increasing the IW-size. Hence, we recommend IW to be a function of the flow-size instead of being set to a single value for all flows.

We are currently carrying out analysis using an integrated packet-flow model, with initial results reflecting our simulation results. Though we came up with an arbitrary (but simple) function in this paper to show the usefulness of an IW-size function, in future, we plan to work more on the analytical model to find the existence of a size-based function for IW that will improve some importance performance metrics in question.

References

1. Allman, M., Floyd, S., Partridge, C.: Increasing TCP's Initial Window. RFC 3390 (Proposed Standard) (October 2002)
2. Avrachenkovt, K., Ayesta, U., Brown, P., Nyberg, E.: Differentiation between short and long TCP flows: predictability of the response time. In: IEEE INFOCOM 2004, vol. 2, pp. 762–773 (March 2004)
3. Collange, D., Costeux, J.L.: Passive estimation of quality of experience. J. UCS 14(5), 625–641 (2008)
4. Divakaran, D.M., Altman, E., Primet, P.V.-B.: Size-Based Flow-Scheduling Using Spike-Detection. In: Al-Begain, K., Balsamo, S., Fiems, D., Marin, A. (eds.) ASMTA 2011. LNCS, vol. 6751, pp. 331–345. Springer, Heidelberg (2011)
5. Divakaran, D.M., Carofiglio, G., Altman, E., Primet, P.V.-B.: A Flow Scheduler Architecture. In: Crovella, M., Feeney, L.M., Rubenstein, D., Raghavan, S.V. (eds.) NETWORKING 2010. LNCS, vol. 6091, pp. 122–134. Springer, Heidelberg (2010)
6. Dukkipati, N., Refice, T., Cheng, Y., Chu, J., Herbert, T., Agarwal, A., Jain, A., Sutin, N.: An argument for increasing tcp's initial congestion window. SIGCOMM Comput. Commun. Rev. 40, 26–33 (2010)
7. Eggert, L.: Moving the Undeployed TCP Extensions RFC 1072, RFC 1106, RFC 1110, RFC 1145, RFC 1146, RFC 1379, RFC 1644, and RFC 1693 to Historic Status. RFC 6247 (Informational) (May 2011)
8. Kodama, S., Shimamura, M., Iida, K.: Initial CWND determination method for fast startup TCP algorithms. In: IWQoS, pp. 1–3. IEEE (2011)
9. Rai, I., Biersack, E., Urvoy-Keller, G.: Size-based scheduling to improve the performance of short TCP flows. IEEE Network 19(1), 12–17 (2005)
10. Scharf, M.: Performance analysis of the Quick-Start TCP extension. In: BROADNETS 2007, pp. 942–951 (September 2007)
11. Zhang, Y.: Speeding up short data transfers: Theory, architecture support, and simulation results. In: Proc. NOSSDAV 2000 (2000)

Performance Evaluation of Bandwidth and QoS Aware LTE Uplink Scheduler

Safdar Nawaz Khan Marwat[1], Thushara Weerawardane[2], Yasir Zaki[1], Carmelita Goerg[1], and Andreas Timm-Giel[2]

[1] ComNets, University of Bremen, 28359 Bremen, Germany
{safdar,yzaki,cg}@comnets.uni-bremen.de
[2] ComNets, Hamburg University of Technology, 21073 Hamburg, Germany
{tlw,timm-giel}@tuhh.de

Abstract. A Long Term Evolution (LTE) eNodeB Medium Access Control (MAC) uplink scheduler is proposed in this paper for Single Carrier Frequency Division Multiple Access (SC-FDMA) as the uplink transmission scheme. Uplink scheduling algorithms available in literature commonly do not consider all the essential features of the LTE uplink. The proposed scheduler is shown to provide efficient allocation of radio resources to User Equipments (UEs) according to Quality of Service (QoS) of various traffic classes and the instantaneous channel conditions. The scheduler functionality is divided into Time Domain Packet Scheduling (TDPS) and Frequency Domain Packet Scheduling (FDPS). The proposed scheduler also supports multi-bearer UEs. The performance of the proposed scheduler is compared with common TDPS schedulers like Blind Equal Throughput (BET), Maximum Throughput (MT) and Proportional Fair (PF). The results show that the proposed scheduler guarantees provision of QoS to UEs and achieves an acceptable performance in terms of throughput.

Keywords: SC-FDMA, uplink, scheduling, bandwidth, QoS.

1 Introduction

Single Carrier Frequency Division Multiple Access (SC-FDMA) in Long Term Evolution (LTE) uplink divides the transmission bandwidth into subcarriers to provide resource allocation flexibility and to achieve high spectral efficiency. The User Equipment (UE) consumes low battery power due to low Peak-to-Average Power Ratio (PAPR) of the SC-FDMA signals. A major SC-FDMA constraint is that the subcarriers allocated to a single UE should to be adjacent to each other. Designing a packet scheduler requires tackling of various conflicting requirements such as channel conditions, fairness, throughput, delay etc. The scheduler should be aware of the number of maximum Physical Resource Blocks (PRBs) allocable to a UE determined by Power Control (PC). The scheduler should support multi-class UEs. The resources should be allocated to UEs according to UE buffer size.

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 298–306, 2012.

Literature survey shows that most of the research work in this field is directed towards downlink scheduling [1,2] etc. Subcarrier allocation algorithm proposed in [3] is search-tree based with fixed size and contiguous bandwidth allocation. This algorithm does not utilize the bandwidth flexibility feature of SC-FDMA. Adaptive transmission bandwidth algorithms in [4,5] provides better throughput performance but other uplink scheduling aspects (e.g. QoS provision) are not addressed. Throughput based QoS metric is proposed in [6] for optimizing resource utilization and fairness but multi-bearer UE support is not addressed. [7] suggests the number of UEs to be scheduled in a Transmission Time Interval (TTI) be adjusted according to Transmission Control Protocol (TCP) congestion window size but QoS provision and other scheduling aspects are not discussed. The weighted metrics based on path loss [8] and intercell interference [9,10] can improve the cell throughput, but QoS fulfillment cannot be ensured. Throughput based QoS weight is introduced in [11] but it cannot serve delay sensitive traffic efficiently. Most of the work on the topic does not consider the PC functionality.

2 Bandwidth and QoS Aware Uplink Scheduler

This paper presents Bandwidth and QoS Aware (BQA) LTE uplink scheduler, which encapsulates most of the features of SC-FDMA based LTE uplink in its functionalities by combining the bits-and-pieces of the research work previously done on the topic. The BQA scheduler is optimized to guarantee QoS provision to the UEs within the maximum number of allowed PRBs (tuned by Fractional Power Control (FPC) scheme based on Closed Loop Power Control (CLPC) [12]). It maximizes the cell throughput by giving priority to UEs with better channel conditions and maintains some level of fairness by providing resources to UEs with adverse channel conditions. Multiple QoS traffic type UEs are also supported by the scheduler, which is the missing aspect of most of the schedulers proposed in literature. The scheduling is performed in two main phases i.e. the Time Domain Packet Scheduling (TDPS) and the Frequency Domain Packet Scheduling (FDPS).

2.1 Time Domain Packet Scheduling

The TDPS phase is designed to prioritize scheduling candidate UEs for a particular TTI in a given cell. A candidate gets high metric value if it has stringent QoS requirements, better channel conditions and/or has been unable to obtain any significant resources in the recent past TTIs. The presence of data packets in the buffer of a UE is reported to the eNodeB using Buffer Status Reports (BSRs). The channel condition of the active UEs (having pending uplink data) is acquired using Channel State Information (CSI) carried by Sounding Reference Signals (SRSs). In this work, it is assumed that the SRSs are received at the eNodeB in each TTI for all the PRBs of active UEs; and the eNodeB is aware of the Power Spectral Density (PSD) of each active UE using power headroom reports. The TDPS metric values are generated for each active UE by using algorithm named as 'weighted Proportional Fair' (wPF) algorithm. The metric value is based on QoS weight and channel conditions of UEs along with fairness. The QoS weight of a UE depends on throughput and delay requirements of its radio bearers. The TDPS metric value for UE i is formulated as:

$$\Lambda_i(t) = \frac{R_{inst,i}(t, n_i)}{R_{avg,i}(t)} \sum_k W_{i,k}(t) \tag{3}$$

Where $\Lambda_i(t)$ is the TDPS metric value for UE i, $R_{inst,i}(t, n_i)$ is the instantaneously achievable throughput of UE i having maximum allowed number of PRBs n_i (set by FPC), $R_{avg,i}(t)$ is the average throughput of UE i at time t expressed as in (4), $W_{i,k}(t)$ is the QoS weight of the bearer k of UE i at time t and expressed as in (5):

$$R_{avg,i}(t) = \left(1 - \frac{1}{\mathrm{T}}\right) R_{avg,i}(t-1) + \frac{1}{T} R_{ach,i}(t) \tag{4}$$

Where T is the Exponential Moving Average (EMA) time window and $R_{ach,i}(t)$ is the actual bit rate achieved by UE i in previous TTI.

$$W_{i,k}(t) = \frac{R_{min,k}}{R_{avg,i,k}(t)} . \frac{\tau_{i,k}(t)}{\tau_{max,k}} . \varrho_k(t) \tag{5}$$

Here, $R_{min,k}$ is the bit rate budget (minimum throughput) and $\tau_{max,k}$ is the end-to-end delay budget of QoS class k, $R_{avg,i,k}(t)$ is the average throughput and $\tau_{i,k}(t)$ is the average delay of bearer k of UE i, $\varrho_k(t)$ is a variable with value set to 10 if $\tau_{i,k}(t)$ is above the threshold value of bearer k at time t, otherwise equal to 1. A list of bit rate budget, packet delay budget and delay threshold values for various QoS classes (defined according to their traffic models) is given in TABLE 1.

Table 1. Bearer Bit Rate Budget; Delay Budget and Delay Threshold

Traffic Type	Bit rate budget (Kbps)	Packet end-to-end delay budget (ms)	Packet delay threshold (ms)
VoIP	55	0.1	0.02
Video	132	0.3	0.1
HTTP	120	0.3	--
FTP	10	0.3	--

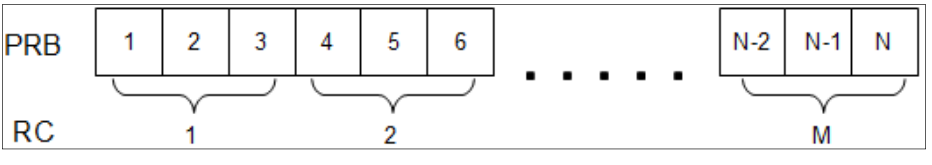


Fig. 1. M RCs with chunk size 3 and bandwidth of *N* PRBs

2.2 Frequency Domain Packet Scheduling

In FDPS, a certain number of high priority UEs are selected for allocation of frequency resources within the TTI. The bandwidth is divided into portions (Fig. 1) called Resource Chunks (RCs). The RCs are allocated to the chosen UEs based on the FDPS metric values for each RC of each UE and the maximum RCs allowed to each UE (set

by FPC). The FDPS metric values also consider the criteria of QoS assurance, throughput maximization and fairness. The allocation of resources is achieved by using a search-tree based algorithm. The bearers of UE with multiple QoS traffic types are scheduled according to their QoS requirements.

In FDPS, a certain number of UEs with highest TDPS metric values are selected for scheduling. The 'Proportional Fair Scheduled QoS-aware' (PFSchedQ) FDPS metric is introduced. The FDPS metric value for a PRB c is expressed as follows:

$$\lambda_{i,c}(t) = \frac{R_{inst,i,c}(t)}{R_{sch,avg,i}(t)} \cdot \sum_{k} W_{i,k}(t) \tag{6}$$

Where $\lambda_{i,c}(t)$ is the FDPS metric value for PRB c of UE i, $R_{inst,i,c}(t)$ is the instantaneously achievable throughput for PRB c of UE i, $R_{sch,avg,i}(t)$ is the instantaneously achievable throughput of UE i over only those TTIs where i successfully enters the FDPS.

PRB Allocation Algorithm

The 'Fixed Size Chunk and Flexible Bandwidth' (FSCFB) algorithm is proposed for PRB allocation to UEs. This algorithm divides the spectrum into several RCs. Variable number of RCs can be allocated to UEs. All possible RC allocation combinations are checked in order to find the best one. The algorithm is computationally intense and therefore, the resolution of this algorithm in frequency domain has been reduced to RC level (and not PRB level). The combinations not following the restrictions of contiguity, buffer size, and maximum allowed PRBs are discarded, resulting in reduced complexity. The steps involved in this algorithm are summarized as follows:

1. Make a UE-RC table (Figure 2) with each element being the RC metric value of the UE, i.e. the sum of PRB metric values within that RC.
2. Make all possible combinations of UE-RC allocation using search-tree algorithm (explained later) while respecting the contiguity, buffer size and maximum allowed PRBs constraint; and determine the resulting global metric value for each combination.
3. Choose the combination with the best global metric value.
4. Obtain the resource allocation from best combination.

	RC_0	RC_1	RC_2
UE_0	10	6	3
UE_1	11	10	5

Fig. 2. A sample UE-RC table for two UEs and three RCs

Step 2 utilizes a search-tree based resource allocation algorithm named as 'unique Depth-First Search' (uDFS) algorithm with contiguity, buffer size and maximum allowed PRBs constraints. The uDFS checks all possible combinations of RC

allocation. The combinations which do not follow the constraints are discarded and further depth of that node is not explored. In Figure 3, it is assumed that at most, 2 RCs can be allocated to a UE. The light grey nodes breach contiguity constraint and the dark grey nodes breach maximum PRBs constraint.

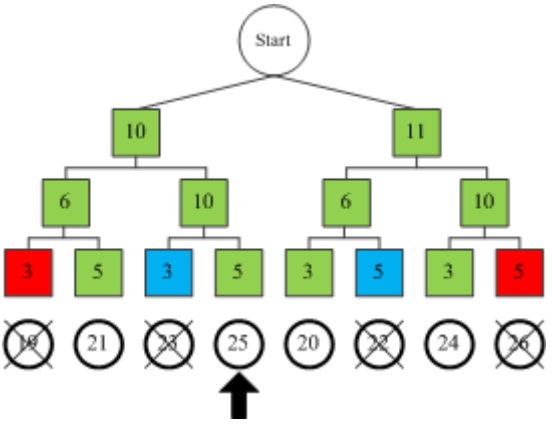


Fig. 3. uDFS tree for two UEs and three RCs

UE Bearer Service

In multi-bearer UE scenario, the allocated bandwidth is further subdivided among the UE bearers. Each UE bearer has its own QoS requirements related to delay budget, rate budget and delay threshold. The UE has to feed its bearers in efficient manner to ensure QoS provision and to avoid bearer starvation. The method proposed for bearer service is named as "weighted service". In this method, a bearer is served according to its QoS weights, $W_{i,k}(t)$ without any priority to GBR. However, the bearers having reached their packet delay threshold are given strict priority and the available resources are allocated to them before serving other bearers.

3 Simulation Results and Analysis

The QoS and the throughput performance of BQA scheduler is compared with commonly used TDPS schedulers; the Blind Equal Throughput (BET), the Maximum Throughput (MT) and the Proportional Fair (PF). In FDPS, these schedulers are combined with the Proportional Fair Scheduled (PFSched). All the schedulers in the simulations avail the FSCFB algorithm and therefore, show performance mostly comparable to the proposed scheduler. The reference schedulers serve the UE bearers by giving strict priority to GBR bearers. TABLE II gives the simulation parameters used.

VoIP, FTP and Video Single-Bearer UEs Scenario

In this scenario, the BQA scheduler and the reference schedulers (BET, MT and PF) are compared with 8 FTP UEs initially. The traffic is modified for successive simulations by adding 2 VoIP UEs and 2 video UEs step-wise. So the simulations are performed for 8 FTP UEs with 0, 2, 4, 6, 8 and 10 VoIP/video UEs respectively. The

average cell throughput and FTP response time results are depicted in Figure 4 and 5 respectively (legend only in Figure 5). The PF and MT schedulers have higher average throughput and lower FTP response time compared to BQA. However, the VoIP end-to-end delay results in Figure 6 show that BQA out-performs PF and MT. Similarly, the video end-to-end delay results (Figure 7) also show that BQA performs better than MT and PF (MT out of range in Figure 5 and 6 due to huge packet delays).

Table 2. Main Simulation Parameters

Parameter	**Setting**
Cell Layout	3 Cells, 1 eNodeB
System Bandwidth	5 MHz (~25 PRBs)
Frequency reuse factor	1
Cell radius	375m
UE velocity	120kmph
Max UE power	23dBm
Path loss	$128.1+37.6\log_{10}(R)$, R in km
Slow fading	Log-normal shadowing, 8dB standard deviation, correlation 1
Fast fading	Jakes-like method [13]
Mobility Model	Random Way Point (RWP)
Power Control	FPC, $\alpha = 0.6$, $P_0 = -58$dBm
Traffic environment	Loaded
Max FDPS UEs	5
RC size	5
VoIP traffic model	
Silence/ talk spurt length	Exponential(3) sec
Encoder scheme	GSM EFR
Video traffic model	
Frame size	1200 bytes
Frame inter-arrival time	75ms
HTTP traffic model	
Page size	100Kbytes
Page inter-arrival time	12 sec
FTP traffic model	
File size	20Mbytes
File inter-request time	Uniform distribution, min 80s, max 100s

VoIP, FTP and HTTP Multi-Bearer UEs Scenario

In this scenario, multi-bearer UEs with VoIP, FTP and HTTP bearers are simulated. The performance of schedulers is compared in terms of average cell throughput, FTP average upload response time, HTTP page response time and VoIP packet average end-to-end delay. The average cell throughput and FTP graphs are similar to the previous scenario. However, BQA turns out to be the best scheduler for providing VoIP service. It has the least average end-to-end delay for VoIP packets in all the

simulations under this scenario. This is illustrated in Figure 20. The HTTP average page response time graphs are depicted in Figure 22 and show acceptable results for BQA, considering the fact that high priority VoIP traffic is present.

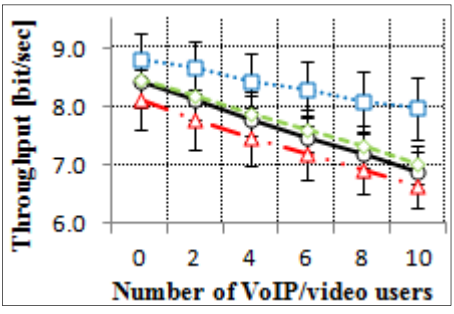


Fig. 4. Average cell throughput

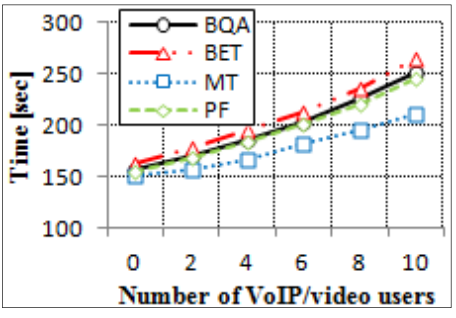


Fig. 5. FTP average upload response time

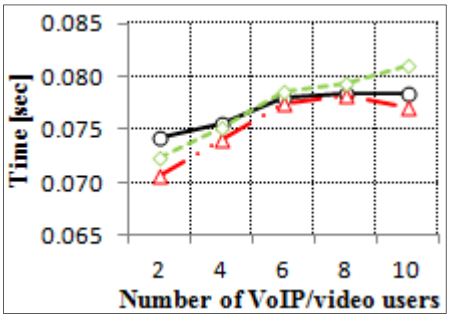


Fig. 6. VoIP average end-to-end delay

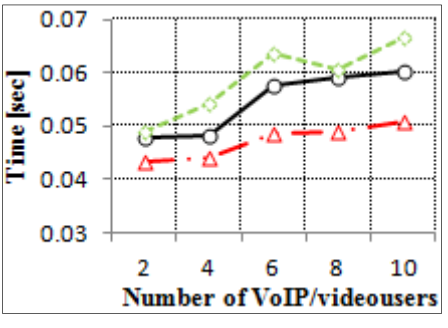


Fig. 7. Video average end-to-end delay

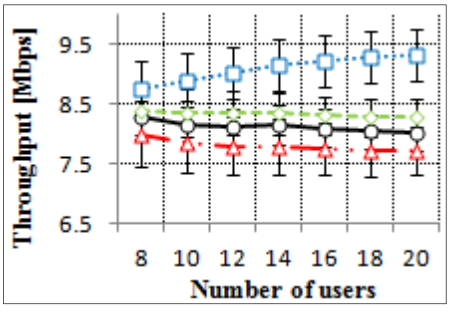


Fig. 8. Average cell throughput

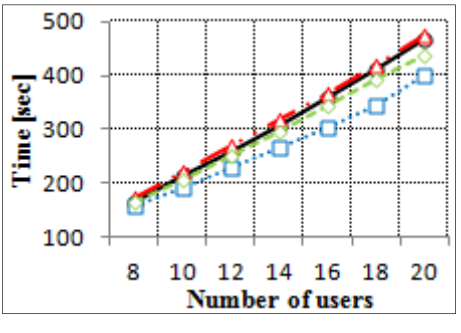


Fig. 9. FTP upload response time

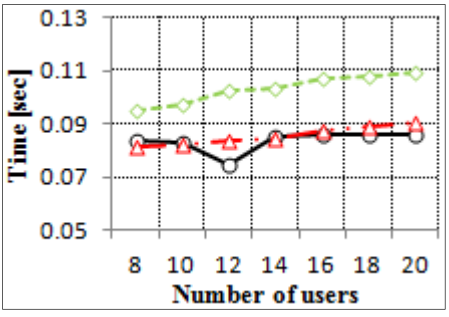


Fig. 10. VoIP average end-to-end delay

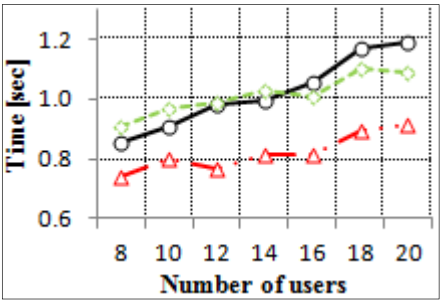


Fig. 11. HTTP average page response time

4 Conclusion and Outlook

This paper proposes the BQA scheduler, designed to guarantee QoS provision to the UEs. It maximizes the cell throughput by giving priority to UEs with better channel conditions. Multi-bearer UEs are supported by the scheduler. The scheduler decisions are in accordance with FPC. The scheduler is time and frequency domain decoupled. The resource allocation is performed with bandwidth flexibility, contiguity constraint of subcarriers and UE buffer size consideration. Simulation results confirm QoS provision of the scheduler to UEs. In future, reducing the tree algorithm complexity and implementing Admission Control (AC) functionality would be of interest.

References

[1] Gavrilovska, L., Talevski, D.: Novel Scheduling Algorithms for LTE Downlink Transmission. In: 19th Telecommunications Forum, November 22-24, pp. 398–401 (2011)

[2] Wang, D., Soni, R., Chen, P., Rao, A.: Video Telephony over Downlink LTE Systems with/without QoS Provisioning. In: 34th IEEE Sarnoff Symposium, May 3-4, pp. 1–5 (2011)

[3] Calabrese, F.D., et al.: Search-Tree Based Uplink Channel Aware Packet Scheduling for UTRAN LTE. In: IEEE 67th Vehicular Technology Conference, May 11-14, pp. 1949–1953 (2008)

[4] Calabrese, F.D., et al.: Adaptive Transmission Bandwidth Based Packet Scheduling for LTE Uplink. In: IEEE 68th Vehicular Technology Conference, September 21-24, pp. 1–5 (2008)

[5] Ruiz de Temiño, L., Berardinelli, G., Frattasi, S., Mogensen, P.E.: Channel-Aware Scheduling Algorithms for SC-FDMA in LTE Uplink. In: IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications, September 15-18, pp. 1–6 (2008)

[6] Ghandour, F., Frikha, M., Tabbane, S.: A Fair and Power Saving Uplink Scheduling Scheme for 3GPP LTE Systems. In: International Conference on the Network of the Future, November 28-30 (2011)

[7] Wang, X., Konishi, S.: A Novel TCP-Oriented Multi-Layer Packet Scheduling Algorithm. In: IEEE 73rd Vehicular Technology Conference, May 15-18, pp. 1–5 (2011)
[8] Yang, C., Wang, W., Qian, Y., Zhang, X.: A Weighted Proportional Fair Scheduling to Maximize Best-Effort Service Utility in Multicell Network. In: IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications, September 15-18, pp. 1–5 (2008)
[9] Wang, Y., Pedersen, K.I., Navarro, M., Mogensen, P.E., Sørensen, T.B.: Uplink Overhead Analysis and Outage Protection for Multi-Carrier LTE-Advanced Systems. In: IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications, September 13-16 (2009)
[10] Yan, Y., et al.: A New Autonomous Component Carrier Selection Scheme for Home eNB in LTE-A System. In: IEEE 73rd Vehicular Technology Conference, May 15-18, pp. 1–5 (2011)
[11] Monghal, G., et al.: QoS Oriented Time and Frequency Domain Packet Scheduler for the UTRAN Long Term Evolution. In: IEEE Vehicular Technology Conference, May 11-14 (Spring 2008)
[12] 3GPP Technical Specification 36.213 V 9.2.0, Physical layer procedures (June 2010)
[13] Cavers, J.K.: Mobile Channel Characteristics. Kluwer Academic Publishers (2002)

Voice Quality Improvement with Error Concealment in Audio Sensor Networks

Okan Turkes and Sebnem Baydere

Department of Computer Engineering
Yeditepe University, Istanbul 34755, TR
{oturkes,sbaydere}@cse.yeditepe.edu.tr
http://cse.yeditepe.edu.tr

Abstract. Multi-dimensional properties of audio data and resource-poor nodes make voice processing and transmission a challenging task for Wireless Sensor Networks (WSN). This study analyzes voice quality distortions caused by packet losses occurring over a multi-hop WSN testbed: A comprehensive analysis of transmitted voice quality is given in a real setup. In the experiments, recorded signals are partitioned into data segments and delivered efficiently at the source. Throughout the network, two reconstruction scenarios are considered for the lost segments: In the first one, a raw projection is applied on voice with no error concealment (V-NC) whereas the latter encodes a simple error concealment (V-EC). It is shown that with an affordable reconstruction, a comprehensible voice can be gathered even when packet error rate is as high as 30%.

Keywords: audio sensor networks, voice quality assessment, wireless multimedia sensor networks, voice coding, error concealment.

1 Introduction

Over the recent years, audio utilization in resource-constrained wireless networks has been a progressive subject. However, nodes composing these networks have inherently confined capabilities to handle streams generated. Hence, an affordable consistency has to be provided between stream delivery and limited resources. Another significant challenge is to receive an intelligible content at the network end-point. Accordingly, several applications related with audio sensor networks are developed [1,2,3,5] and several performance criteria are analyzed [13,12]. However, audio is a multi-dimensional function that no measure itself can accurately evaluate all of its aspects. So is the voice, which is a specific audio data type targeted in this study. Since related applications need a voice quality assessment (VQA), a great deal of data properties need to be analyzed as promptly as practicable. Despite the diversity of VQA methods [4,8,9], there is no standard measure which can evaluate several properties in company. Besides, many studies are not validated by real testbed experiments in spite of notable theoretical solutions. We elaborate on an objective VQA metric adapted from transmission rating factor defined in E-Model of ITU-T [6]. The proposed metric is modified according to the properties of the real environment targeted.

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 307–314, 2012.

This study mainly focuses on data quality distortions caused by packet losses during voice transmission over a real wireless sensor network (WSN). Over 9,000 generated streams are transmitted over a multi-hop line. VQA of each delivery is analyzed with an experimental setup. During the transmission of segmented network packets, nodes accommodated with a simple buffering mechanism try to increase data integrity. As a follow-up work of [11], two scenarios are considered for lost segment reconstruction: First one set silence onto the lost samples in projected voices which maintains a raw form with no error concealment (V-NC). In the second one, lost segments are encoded with an averaging method (V-EC) based on a reconstruction between its neighbor packets. The results are evaluated for both scenarios in terms of several data and network parameters.

The rest of the paper is organized as follows. Voice coding and transmission model is presented in Section 2. VQA is issued in Section 3. Section 4 renders the transmission and evaluation environment. Section 5 discusses the system performance. Conclusion is given in Section 6 with comments on forward plans.

2 Voice Coding and Transmission

Transmission model consists of two types of nodes; Type 1 A_i, i=1, 2 . . . , n_2, source node equipped with acoustic sensor on it and Type 2 S_{ij}, i=1, 2 . . . n_1, j=1, 2 . . . , n_2, node that is simple routing sensor. In this model, different network properties are analyzed with several data segmentation and transmission characteristics with regard to the presented error concealment (EC) schemes.

Partitioning of streams at the source node is necessary since the overall data cannot be fit within the limited network packet size. Besides, segmentation has to include low-cost steps in order to decrease processing delay. Size of a segment (s_w) should also be maximized as much as possible to decrease total number of generated packets, $n(p)$. In a particular transmission, $n(p)$ is determined with:

$$n(p) = \frac{f_s \times t}{s_w} \tag{1}$$

In this study, nodes are built up to hold $s_w = \{20, 40, 80\}$ amplitude values in a data segment. For a specific s_w, $n(p)$ varies according to the sampling frequency (f_s) and the duration (t) of a voice selected in the data set. We aim to examine the effect of s_w on voice quality when data and network characteristics differ.

Heavy number of data packets passed over to each inter-hop of the network struggles with transmission delay and bandwidth. To minimize pre-transmission processing delay, A_i utilize a simplistic mechanism which buffers the segments with corresponding packet indices into the data memory. Same buffering structure is also accommodated on S_{ij} in order to minimize the relay time.

In each successful packet transfer, its corresponding index is also gathered. Thus, unattained packets are determined at the end of a voice transfer. Hence, loss pattern for a transfer is generated and projection is applied for the lost segments by considering two construction schemes: The first scheme inserts s_w zero amplitude values into the location of each lost segment, thus maintaining

the raw form with no concealment (V-NC) over the data set. For the second scheme (V-EC), a lost segment is corrected by siting the arithmetic mean of the sample units gathered from the previous and the next successfully gathered packets onto the lost sample units. The details of the algorithm is given below:

Algorithm 2.1. Algorithm for V-EC

```
1: Read the received voice data, D^r.
2: Determine the indices of lost segments in ascending order.
3: for all lost segment do
4:    Find the starting location of the sample units going to be reconstructed in D^r.
5:    Find the starting locations for the preceding and the next segment of the lost segment
      being dealt. If the next or previous segment is also missing, refer to next or previous
      neighbor until a healthy one is found. If the lost segment being dealt is the initial
      segment of the data, assign zero amplitude values into the segment. If the lost segment
      being dealt is the last one of the data, assign zero amplitude values into the segment.
6:    Create a temporary array having a size of s_w.
7:    for s_w times do
8:       Sum up each value sample unit value of the neighbor segments.
9:       Store their arithmetic mean in the array.
10:   end for
11:   Locate the array to the location of the lost segment.
12: end for
13: Generate the overall constructed data at the sink.
```

3 Voice Quality Assessment

In this study, VQA is valuated by a simplified version of transmission rating factor (R-factor) of ITU-T, which is an objective metric that can be easily accommodated on sensor nodes. The parameters of the equation is given below:

$$R = R_0 - I_s - I_d - I_{e,eff} + A \tag{2}$$

This study treats the packet loss probability (P_{pl}) defined in $I_{e,eff}$ as the main impairment factor. P_{pl} is inversely associated to transmission success rate (TSR), thus a relationship between voice quality and TSR is wanted to be revealed. Besides, f_s of a data is associated to simultaneous impairment factor I_s and investigated with quantizing distortion unit (qdu) defined in I_s. The permitted interval for qdu starts from value 1, meaning that a complete data quantization is supplied. When the quantization is at the lowest, qdu ends at value 14. To specify a scale between f_s and qdu, we assume the maximum f_s utilized in the tests—16KHz has the complete quantization. Conversely, the minimum audible f_s that a human ear can sense—3KHz is set for the maximum distortion. For all f_s used in our data set, qdu grades are determined and corresponding I_s values are generated, as shown in Table 1. By setting other parameters to their default values, R-factor is simplified to the following function of f_s and P_{pl}:

$$R(f_s, P_{pl}) = 58.9843 - 95\frac{P_{pl}}{P_{pl}+1} + 2.0714 \times f_s \tag{3}$$

A value obtained with R-factor can be mapped to a Mean Opinion Score (MOS) which is a widely used subjective VQA method. It is simply determined

Table 1. qdu and I_s values according to several f_s

f_s (Hz)	qdu	I_s
16000	1	1.4136
11025	6.025	10.0949
8000	9	17.1918
6000	11	22.0516
4000	13	26.4005

by the perceptual grades of an experimental group of audience. Ranging from bad to excellent, MOS is identified among a numerical quality scale from 1 to 5, respectively. In this way, R-factor gives an advantage of a VQA in both objective and subjective manners. For example, 90, 70 and 50 as R-factor values are mapped to MOS values of 4.3 (excellent), 3.6 (fair) and 2.6 (bad), respectively.

The correlation among each inter-hop link quality is traced with the following signal-to-noise ratio (SNR) metric

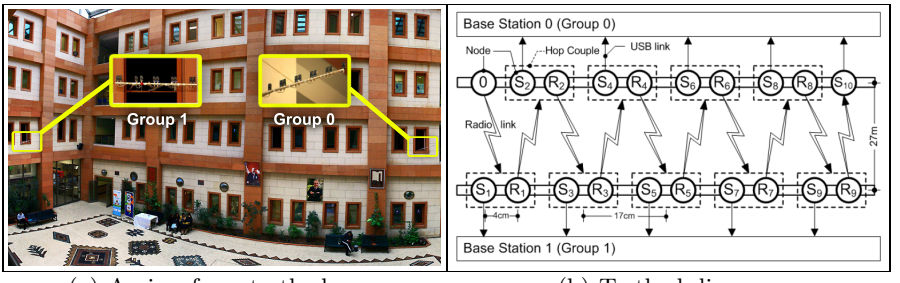
$$SNR(dB) = 10 \log_{10} \frac{|A_{signal}|}{|A_{noise}|} \tag{4}$$

where $|A_{signal}|$ is the total absolute values of the original amplitudes and $|A_{noise}|$ is of the difference between the original and reconstructed voices. SNR is utilized to examine the performance between the reconstruction techniques. Quality of both network and voice is assessed by the comparison of SNR and R-factor.

4 Experimental Setup

Actual transmissions are conducted inside a building with a large *atrium* as shown in Figure 1(a). A homogeneous testbed environment comprising a 10-hops network is constructed by 20 TMote Sky [10] sensor nodes. The nodes are associated with two groups, Group 0 and Group 1, which are lined up with a 28m distance in a parallel manner. The groups consist of five "hop couples", as depicted in Figure 1(b). TinyOS v2.1 with nesC v1.3 [7] is utilized to realize the data transfer. We have generated a voice data set comprising of simple invocatory commands each having a same fixed t. Each voice is generated with varying f_s listed in Table 2 and bit depths bd={8, 16}. Data $D^s{}_i$, i=1, 2 . . . , 8 are prerecorded with different f_s and bd via an acoustic sensor. Then, the samples with 8KHz/8bit are segmented and transmitted to the source serially, and then over the wireless transmission route with a fixed s_w in each unique test.

In our voice transfer scheme, each hop consists of two nodes called "hop couple". One of the nodes in each couple, called relay node (R_i, i=0, 1 . . . 9), is used to send the incoming data to the consecutive hop couple via radio link. Meanwhile, the other node, called snooping node (S_i, i=1, 2 . . . 10), is used to send the incoming data to the base station computer via USB link. To make hop based VQA with a wide variety of TSR, intermediate results in each hop



(a) A view from testbed area (b) Testbed diagram

Fig. 1. Testbed Environment

are recorded by snooping nodes. Nodes having IDs 0 and 10 are the source and the sink, respectively. In each test, the received data $D^r{}_i$, i=1, 2 . . . , 8 are saved at every hop with their corresponding packet indices. When a transmission of a voice data is over, unperceived data segments are determined. Corresponding mask files are generated for every s_w and f_s. Since f_s=8KHz is utilized in real tests, masks for lower and higher f_s are derived with down-sampling and up-sampling, respectively. With a conducive simulation, V-NC and V-EC are applied on the lost segments during projection over different voices in the data set.

Table 2. $n(p)$ according to s_w and f_s

		f_s (Hz)				
		4000	6000	8000	11025	16000
s_w	20	800	1200	1600	2205	3200
	40	400	600	800	1103	1600
	80	200	300	400	552	800

5 Performance Analysis

We have conducted 864 real voice transfer tests spreading to 10 days and gathered 22,800 voice loss patterns at 10 hops. The reflection of applying V-EC on the data gathered in comparison to V-NC can be clearly seen in Figure 2, which consists of nearly 18,000 SNR values calculated for all s_w. The distinction between each hop is noticed easily with color gradients for both V-NC and V-EC. For all s_w, values indicate that V-EC notably increases the quality. For s_w=40 and s_w=80, V-EC gets fair values in comparison to V-NC, but not so much higher as for s_w=20.

The graphs which show R-factor, SNR and TSR relationships in Figure 3 and Figure 4 include all the results projected to 8 different voice data with all f_s and *bd* in the simulation environment. It is seen that SNR values for the same voices

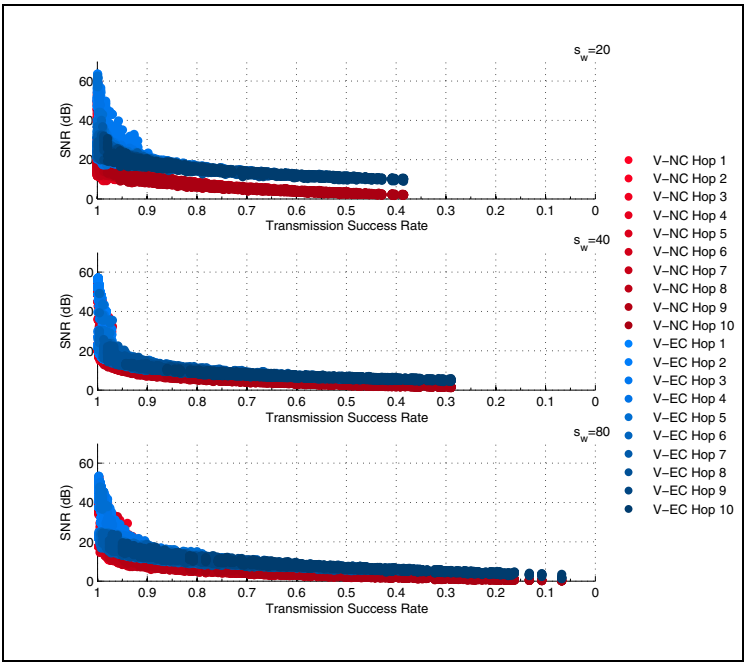


Fig. 2. SNR values of V-NC and V-EC algorithms wrt segment size

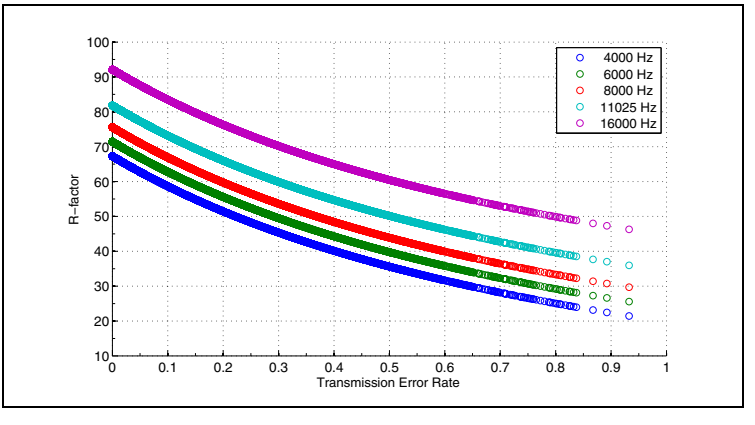


Fig. 3. Scatter plot of R-factor vs transmission error rate

with different *bd* versions are very similar to each other. Regardless of s_w or *bd* of a data transmitted, R-factor only depends on the overall TSR and f_s.

Figure 3 depicts the relation between R-factor and overall link quality. The graph shows that a comprehensible voice can be gathered when packet error rate is insured to be less than 30%. For a voice sampled at f_s=16KHz, R-factor value

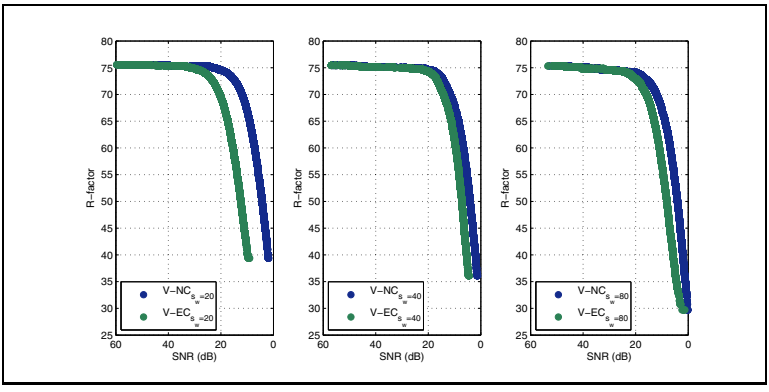


Fig. 4. Relationship between R-factor vs SNR

is nearly 70 even when TSR is 60%, which means that voice has a fair quality in terms of MOS. However, the decrease in f_s also decreases the metric values.

In Figure 4, the correlation between R-factor and SNR is depicted when f_s=8KHz. For both of the data resolutions—8 bit and 16 bit, SNR values for concealment algorithms on the overall data show resemblance. The effect of s_w can smoothly be seen on the data set. When s_w=20, increase in V-EC values are at maximum. Quality metrics for network and data intelligibility—R-factor and SNR visibly relate with each other.

6 Conclusion

In this study, a real wireless voice transmission testbed is established in order to disclose quality gradients of the continuous data being dispatched in a lossy multi-hop sensor network environment. The characteristics of the network against environmental factors are kept track of with several number of time-varying tests. The basic characteristics of voice data are essayed with different network properties. The results obtained after 9,000 real testbed transmissions reveal strong correlation between values obtained with the VQA metrics and TSR. The empirical results also showed that an affordable correction algorithm over the lost segments can provide a reasonable achievement in voice quality. We aim to concentrate on several EC algorithms and investigate their performances. Aside from error correction strategies, we aim to examine the effects of several factors defined in R-factor. Thus, data can be evaluated more concisely.

It is quite apparent that the network bandwidth must be efficiently used during voice transmission. Therefore, data characteristics for capturing and transmission must be kept as light as possible. However, the quality evaluation results clearly demonstrate that a transmitted stream becomes incomprehensible when the conventional characteristics of a voice—f_s and bd are lowered. In order to satisfy the affordance between network properties and data qualifications, the significance level of the information in voice data can be utilized. With these

considerations, a priority-based transmission scheme can be an exact solution for data integrity and validity. Several revisions and enhancements in both implementation and evaluation will pave the way for generating a complete voice transmission framework in Wireless Multimedia Sensor Networks.

References

1. Alesii, R., Graziosi, F., Pomante, L., Rinaldi, C.: Exploiting wsn for audio surveillance applications: The vowsn approach. In: 11th EUROMICRO Conference on Digital System Design Architectures, Methods and Tools, DSD 2008, pp. 520–524 (September 2008)
2. Azimi-Sadjadi, M.R., Kiss, G., Feher, B., Srinivasan, S., Ledeczi, A.: Acoustic source localization with high performance sensor nodes. In: Proceedings of SPIE, vol. 6562, 65620Y–65620Y–10 (2007)
3. Berisha, V., Spanias, A.: Real-Time Implementation of a Distributed Voice Activity Detector. In: Fourth IEEE Workshop on Sensor Array and Multichannel Processing, pp. 659–662 (2006)
4. Carvalho, L., Mota, E., Aguiar, R., Lima, A.F., de Souza, J., Barreto, A.: An E-Model Implementation for Speech Quality Evaluation in VoIP Systems. In: 10th IEEE Symposium on Computers and Communications, ISCC 2005, pp. 933–938 (2005)
5. Facchinetti, T., Ghibaudi, M., Anna, S.S.S., Pi, S.G.T., Goldoni, E., Savioli, A.: Real-Time Voice Streaming over IEEE 802.15.4, pp. 985–990. Packaging, Boston (2010)
6. International Telecommunications Union: Itu-t recommendation g.107 (2011), http://www.itu.int/itudocr/itu-t/aap/sg12aap/history/g107/g107ww9.doc
7. Levis, P.: Tinyos: An operating system for sensor networks (2006), http://www.tinyos.net/tinyos-2.x/doc/pdf/tinyos-programming.pdf
8. Li, L., Xin, G., Sun, L., Liu, Y.: QVS: Quality-Aware Voice Streaming for Wireless Sensor Networks. In: 2009 29th IEEE International Conference on Distributed Computing Systems, pp. 450–457 (June 2009)
9. Palafox, L.E., Garcia-Macias, J.A.: Wireless Sensor Networks for Voice Capture in Ubiquitous Home Environments. In: 2009 4th International Symposium on Wireless Pervasive Computing, pp. 1–5 (February 2009)
10. Telosb Crossbow: Telosb data sheet (2010), http://www.willow.co.uk/TelosB_Datasheet.pdf
11. Turkes, O., Baydere, S.: Voice Quality Analysis in Wireless Multimedia Sensor Networks: An Experimental Study. In: The International Conference on Intelligent Sensors, Sensor Networks and Information Processing, ISSNIP, pp. 317–322. IEEE (December 2011)
12. Wang, C., Sohraby, K., Jana, R., Ji, L., Daneshmand, M.: Voice communications over zigbee networks. IEEE Communications Magazine 46(1), 121–127 (2008)
13. Xu, J., Li, K., Shen, Y., Min, G., Qu, W.: Adaptive Energy-Efficient Packet Transmission for Voice Delivering in Wireless Sensor Networks. In: 2009 Sixth IFIP International Conference on Network and Parallel Computing, pp. 86–92 (October 2009)

Analysis of the Cost of Handover in a Mobile Wireless Sensor Network

Qian Dong and Waltenegus Dargie

Chair of Computer Networks, Faculty of Computer Science,
Technical University of Dresden, Germany, 01062
{qian.dong,waltenegus.dargie}@tu-dresden.de

Abstract. This paper investigates the latency of packet transmission during mobility with and without the support of a handover mechanism. The Receiver-Initiated MAC protocol (RI-MAC) is used for the analysis. When the handover mechanism is applied, it enables a node to establish a new connection before the existing link breaks. A mathematical model is set up to examine the performance of RI-MAC when it uses the handover mechanism and when it does not. The analytical result shows that the handover latency is much less than the latency introduced when a node waits until an existing link breaks and then establishes a new link.

Keywords: MAC protocol, wireless sensor networks, mobility, handover.

1 Introduction

This paper investigates the problem of mobility in wireless sensor networks and the need to maintain an unbroken link during data transmission. It examines the latency introduced when a mobile transmitter carries out a seamless handover and when it establishes a new link after an existing link is broken. The type of applications that require mobile nodes have been discussed elsewhere [1].

Mobility can cause frequent topology changes and the deterioration of communication links [2] [3]. Since most of the existing or proposed MAC protocols do not accommodate mobility, a node has two options to deal with a deteriorating link: (a) to continue data transmission until the link breaks and then establish a new link with a new relay node; or (b) to seamlessly transfer the communication to a better link parallel to the data transmission over the existing link. Both approaches will inevitably introduce latency. This paper aims to quantify and compare these latencies.

This paper makes two contributions. First, we assert that the use of a handover mechanism reduces the transmission latency. Second, we provide a mathematical model to quantitatively confirm our assertion.

The remaining part of this paper is organized as follows: in Sections 2 and 3, an optimization to RI-MAC is presented and a handover mechanism is developed. In Section 4, the performance of the optimized RI-MAC with and without the handover support is evaluated. In Section 5, the analytical results are visualized and discussed. Finally, in Section 6, concluding remarks are given.

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 315–322, 2012.

2 RI-MAC And Its Optimization

The Receiver-Initiated MAC protocol (RI-MAC) [4] is chosen to support a seamless handover. Even though the collision detection and packet recovery scheme enables RI-MAC to achieve high packet delivery ratio, it has some demerits. Firstly, in a round of transmission, the BW size in beacons either remains unchanged or keeps on increasing whenever a collision is detected. The increase will be fast in case of small transmission intervals, which can lead to a large back-off window. This makes RI-MAC to work inefficiently in terms of energy and latency. Secondly, the phenomenon that a receiver does not receive any data packet during a dwell time can be because of an unsuccessfully transmitted beacon instead of an idle channel. Consequently, a receiver may go to sleep even though there are transmitters willing to communicate with it.

To address these problems, a burst transmission pattern of data packets should be adopted. Therefore, instead of sending one packet, a node transmits a predefined number of packets in burst each time. The beacon between every two packets is thus only the ACK beacon, aimed at preventing other neighbors of the receiver from competing for the channel in the midst of data communication. In this way, collision can be avoided and the BW size can be considerably reduced. The working principle of the optimized RI-MAC is summarized in Figure 1a.

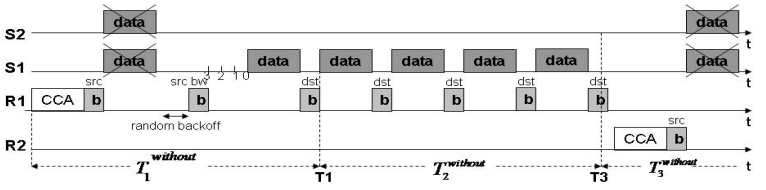
3 Handover

In order to justify a handover, the probability that a link breaks should be reasonably high. This requires a comparatively long data communication duration. The optimized RI-MAC meets this precondition. We predefine a distance threshold d to begin a handover. Figure 1b summarizes a handover process. After the initial back-off and a data transmission, the transmitter, $S1$, will receive an ACK beacon from the intended receiver, $R1$. Based on the RSSI value obtained from the ACK beacon, $S1$ can deduce the relative distance between $R1$ and itself.[1] If the distance is estimated to be larger than d, $S1$ will realize that its data packets cannot be completely transmitted before the link breaks. Therefore, it will immediately broadcast a data packet in which a handover request is embedded.

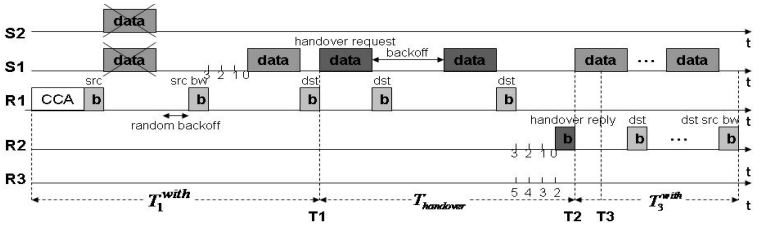
After $R1$, as the original receiver, receives the request, it will learn that $S1$ is searching for a new receiver and thus reply with a normal ACK beacon only. This ACK beacon is necessary, since it ensures that the packet transmission continues whether the handover attempt is successful or not. However, the remaining neighbors of $S1$ will send back handover replies, but not all of them are eligible to participate. Only those active nodes whose distance with respect to $S1$ does not exceed d are qualified. This requirement ensures that the handover mechanism will be triggered at most once during a node's data transmission.

If the neighbors of $S1$ are active, they will definitely receive the handover request. To avoid collision, a node has to do a back-off before transmitting a

[1] This paper assumes that a relationship between the RSSI and d can be established, but it does not examine how this relationship can be accurately established.



(a) without handover support



(b) with handover support

Fig. 1. Communication pattern with and without the handover support

handover reply. Unlike the back-off field used in beacons whose value is variable, the BW size is the same in all the handover requests. The node that wins the channel contention will be regarded as the new receiver for $S1$. However, if $S1$ does not receive any reply until BW expires, it will broadcast another handover request. As soon as a new link is established, $S1$ will resume its data transmission with the newly found receiver, $R2$, in a unicast way. By overhearing the data packet between $S1$ and $R2$, the original receiver, $R1$, will enter into a sleep state.

By taking advantage of the Boolean field in a packet, the transmission mode can be controlled to either broadcast or unicast. If the Boolean field with the value *true* is inserted in a data packet, it signals that it is a handover request. The Boolean field is also employed in every handover reply to avoid collision.

4 Performance Analysis

4.1 With Handover Support

Threshold Determination. Suppose n data packets are transmitted in burst. Among them, the first one is used to obtain the first ACK beacon to estimate the distance. Then $(n-1)$ data packets are left for the transmission. Assuming the node is moving at a uniform speed v along the radius of the radio transmission range, R, of its partner, the relative moving distance of the transmitter during this time can be evaluated. By denoting the data packet size, the ACK beacon size, the transmission rate and the switch time of the radio mode as N_{data}, N_b, R_t and T_{SIFS}, respectively, the distance threshold d can be defined as:

$$d = R - (n-1)\left(\frac{N_{data}}{R_t} + \frac{N_b}{R_t} + 2T_{SIFS}\right)v \tag{1}$$

Handover Latency. Suppose the number of neighbors[2] of the transmitter except the original receiver is N. Among the N neighbors, only those which are already active and whose distance with respect to the transmitter does not exceed d can attempt to send back a handover reply. If the number of these nodes is greater than one, a collision may occur.

To simplify the expression, we assume for each handover attempt, k out of N neighbors of the transmitter are selected to be in an active state, and l out of k nodes are further selected supposing their distance with respect to the transmitter does not exceed the threshold. By considering the duty cycle, D, the average handover latency, $\overline{T}$, can be expressed as:

$$\overline{T} = \frac{\sum_{k=1}^{N}\sum_{l=1}^{k}\left(C_N^k D^k (1-D)^{N-k}\right)\left(C_k^l (\frac{d^2}{R^2})^l (1-\frac{d^2}{R^2})^{k-l}\right)\overline{t_l}}{\sum_{k=1}^{N}\sum_{l=1}^{k}\left(C_N^k D^k (1-D)^{N-k}\right)\left(C_k^l (\frac{d^2}{R^2})^l (1-\frac{d^2}{R^2})^{k-l}\right)} \tag{2}$$

The polynomials $(C_N^k D^k (1-D)^{N-k})$ and $(C_k^l (\frac{d^2}{R^2})^l (1-\frac{d^2}{R^2})^{k-l})$ represent the probability that k nodes are awake and the probability that l nodes have a relative distance not larger than d. Since the term $\overline{t_l}$ refers to the average handover latency caused by a collision on the l nodes, its value is mainly relevant to the number of competitors, l. If l equals to one, no collision will occur and therefore, $\overline{t_{l=1}}$ can be expressed as:

$$\overline{t_{l=1}} = \frac{N_{request}}{R_t} + t_{SIFS} + \frac{N_b}{R_t} + \frac{N_{reply}}{R_t} + \frac{\sum_{m=1}^{BW}(m\sigma)}{BW} \tag{3}$$

The parameters $N_{request}$, N_{reply} and σ denote the size of a handover request and reply and a slot duration, respectively. However, if l is greater than one, a collision and thus a handover failure may occur. Since a handover is assumed to be successful, in the worst case, at i_{th} attempt, $\overline{t_{l\geq 2}}$ should be expressed as:

$$\overline{t_{l\geq 2}} = \frac{p_l^s\left(t^s + \sum_{j=1}^{i-1}\left((1-p_l^s)^j (jt^f + t^s)\right)\right)}{1-(1-p_l^s)^{i-1}} \tag{4}$$

Here t^f represents the time consumed in a handover attempt that fails and it can be expressed as:

$$t^f = \frac{N_{request}}{R_t} + t_{SIFS} + \frac{N_b}{R_t} + BW\sigma \tag{5}$$

The term p_l^s in equation (4) denotes the probability that no collision occurs on the handover reply packets transmitted by the l nodes. If one of the l nodes replies in a particular time slot, to make a handover successful, all the remaining $(l-1)$ nodes should select a time slot having a larger value. But whether these time slots are the same or not is unimportant. p_l^s can thus be expressed as:

$$p_l^s = \frac{C_l^1\left(\sum_{j=1}^{BW-1}(BW-j)^{l-1}\right)}{BW^l} \tag{6}$$

[2] In the subsequent analysis, we do not consider the original receiver as one of the contending neighbors. The use of "neighbors" should be clear from the context.

The back-off time which determines the value of t^s in equation (4) depends on the time slot selected on average by a node for the transmission of the handover reply during which no collision occurs. Therefore, t^s can be written as:

$$t^s = \frac{N_{request}}{R_t} + t_{SIFS} + \frac{N_b}{R_t} + \frac{N_{reply}}{R_t} + \frac{\sigma\Big(\sum_{j=1}^{BW-1}\big((BW-j)^{l-1}j\big)\Big)}{\sum_{j=1}^{BW-1}(BW-j)^{l-1}} \tag{7}$$

Time For The Remaining Data Transmission. As soon as the transmitter finds the new relay node, it will transfer the communication to it immediately. As only one data packet is sent before $T1$ whether the handover mechanism is used or not, $n_1^{with} = n_1^{without} = 1$. Since the total number of data packets transmitted in burst is n and the average number of data packets transmitted during a handover process is $n_2^{with} = \frac{\overline{T}-t^s}{t^f} + 1$, the time required for the remaining data transmission over the new link, t_3^{with}, can be expressed as:

$$t_3^{with} = (n - n_1^{with} - n_2^{with})(\frac{N_{data}}{R_t} + \frac{N_b}{R_t} + 2T_{SIFS}) - T_{SIFS} \tag{8}$$

Finally, by adding $\overline{T}$ and t_3^{with} together, the time required to transmit $(n-1)$ data packets in the optimized RI-MAC in which the handover mechanism is used can be evaluated.

4.2 Without Handover Support

Time Consumed In The Original Link. If a transmitter whose distance with respect to the receiver exceeds d wins the channel, its data transmission will be interrupted because the link will break in the middle of the communication when the handover mechanism is not used. Since the transmitter may be located at any place between d and R within the radio transmission range of the receiver, the distance it travels between $T1$ and $T3$ on average can be evaluated as: $s = \frac{R-d}{2}$. Hence, the time needed to cover this distance, $t_2^{without}$, can be written as:

$$t_2^{without} = \frac{R-d}{2v} \tag{9}$$

At the time the link breaks, either the transmitter fails to receive the ACK beacon or the receiver fails to receive the data packet. But the packet has to be retransmitted in any case. Therefore, the number of data packets left for transmission after the original link breaks, $n_3^{without}$, can be quantified as:

$$n_3^{without} = n - n_1^{without} - \frac{t_2^{without} - \left(\frac{3N_{data}}{4R_t} + \frac{T_{SIFS}}{2} + \frac{N_b}{4R_t}\right)}{\frac{N_{data}}{Rt} + \frac{N_b}{R_t} + 2T_{SIFS}} \tag{10}$$

Time for Setting Up A New Link. If at least one extra receiver is awake before or at the time of a link disruption, a base beacon will be broadcasted at $T3$. This enables the transmitter to establish a new link with a minimum delay. However, the time required for setting up a new link can be very different depending on when the transmitter is able to win the channel. Since at least two nodes are assumed to be present around a particular node in order to carry out the handover, a collision on the data packets must occur in the first round of medium contention. This necessitates a BW field to be additionally included in the next beacon. As a result, in the subsequent contentions, it may happen that (a) the original transmitter wins the channel, (b) one of the other transmitters wins the channel, or (c) no one wins the channel due to a collision. For case (a), the transmitter can directly resume its remaining data transmission. For case (b), the transmitter cannot start sending until the node which wins the channel completes its data transmission. And for case (c), the transmitter can only contend for the medium after the collision.

As different number of active nodes results in different back-off durations, which will further influence the setting up time of a new link, $t_3^{without}$ should be quantified as:

$$t_3^{without} = \frac{\sum_{k=1}^{N} \left(C_N^k D^k (1-D)^{N-k}\right) t_3^{without\_k}}{\sum_{k=1}^{N} C_N^k D^k (1-D)^{N-k}} \tag{11}$$

According to the three possible transmission patterns from the perspective of the original transmitter, $t_3^{without\_k}$ can be expressed as:

$$\begin{aligned} t_3^{without\_k} = & \frac{\sum_{m=1}^{BW1-1} (BW1-m)^k}{BW1^{k+1}} t_a + \frac{\sum_{m=1}^{BW1-1} C_k^1 (BW1-m)^k}{BW1^{k+1}} t_b \\ & + \frac{\sum_{u=1}^{BW1-1} \left(\sum_{m=2}^{k+1} C_{k+1}^m (BW1-u)^{k+1-m}\right) + 1}{BW1^{k+1}} t_c \end{aligned} \tag{12}$$

Here the three coefficients that determine the values of t_a, t_b and t_c denote, in respective order, the probability that $S1$ wins the medium, the probability that one of the other nodes wins the medium, and the probability that a collision occurs in the second transmission attempt. t_a, t_b and t_c can be expressed as:

$$t_a = T_{CCA} + \frac{2N_b}{R_t} + \frac{N_{data}}{R_t} + T_r + T_{backoff1} + n_3^{without} T_{unit} \tag{13}$$

$$t_b = T_{CCA} + \frac{2N_b}{R_t} + \frac{N_{data}}{R_t} + T_r + 2T_{backoff1} + (n + n_3^{without}) T_{unit} - T_{SIFS} \tag{14}$$

$$t_c = T_{CCA} + \frac{3N_b}{R_t} + \frac{2N_{data}}{R_t} + 2T_r + T_{backoff1}^{col} + T_{backoff2} + n_3^{without} T_{unit} \tag{15}$$

The above terms T_r and T_{unit} represent the random back-off duration ($T_r = \frac{\sum_{m=1}^{BW} (m\sigma)}{BW}$), and the time needed to transmit a data packet ($T_{unit} = \frac{N_{data}}{R_t} + \frac{N_b}{R_t} +$

$2T_{SIFS}$). The parameters $T_{backoffm}$ and $T^{col}_{backoff1}$ denote the back-off interval determined by k nodes out of which one wins the channel in the $(m+1)_{th}$ contention $(m = 1, 2)$, and the back-off interval determined by a collision in the 2_{nd} contention, respectively. These two parameters can be further expressed as:

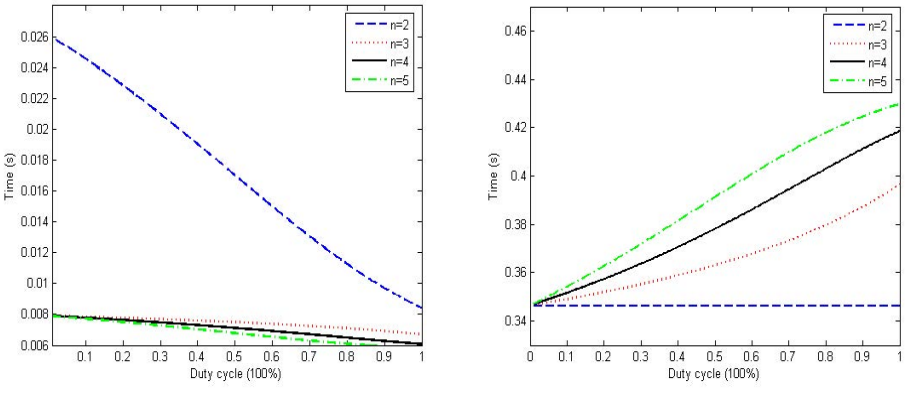
$$T_{backoffm} = \frac{\sum_{u=1}^{BWm-1}(BWm-u)^k u\sigma}{\sum_{u=1}^{BWm-1}(BWm-u)^k} \qquad (m = 1, 2) \tag{16}$$

$$T^{col}_{backoff1} = \frac{\sum_{u=1}^{BW1-1}\left(\left(\sum_{m=2}^{k+1} C^m_{k+1}(BW1-u)^{k+1-m}\right)u\sigma\right) + BW1\sigma}{\sum_{u=1}^{BW1-1}\left(\sum_{m=2}^{k+1} C^m_{k+1}(BW1-u)^{k+1-m}\right) + 1} \tag{17}$$

By combining equations (11)-(17), the time needed to set up a new link can be obtained. Then by combining equations (9) and (11), the time required for transmitting $(n-1)$ data packets without the handover support can be evaluated.

5 Analytical Result and Discussion

We employ Matlab version 7.0.1 to visualize and compare the time required for transmitting a burst of data packets in the optimized RI-MAC when the handover mechanism is applied and when it is not applied. The average moving speed of human beings is normally $1.5m/s$. The size of the beacon and data packets are set to 18 and 45 bytes, respectively. The distance threshold is defined as $24.5m$, which is $0.5m$ smaller than the radio transmission range. 140 packets are transmitted in burst, and the maximum handover attempt is limited to 4.



(a) (b)

Fig. 2. Performance comparison with respect to the handover mechanism

The handover latency is inversely proportional to the duty cycle as well as the network density, as shown in Figure 2a. When the network density grows, the number of nodes whose relative distance is less than d increases. Meanwhile, the probability that a collision occurs among the simultaneously transmitted handover replies increases. But the first (which decreases the handover latency) increases faster, since it is only one of the parameters that determines the second (which increases the handover latency). Even though when the duty cycle approaches 0, the handover latency has a limited value and it remains the same for the network densities that are larger than two. This is because the handover is assumed to be successful at the latest at the i_{th} attempt. In addition, when the duty cycle is extremely small, the probability that k nodes are awake at the same time is very low. Since the l nodes, which determine the handover duration are selected from the k nodes, l has even a smaller value. Figure 2b shows the time needed for the remaining data transmission after the original link breaks. As the duty cycle and the network density decrease, the number of nodes that are able to contend for the medium becomes small, leading to a shorter time for setting up a new link. The duty cycle equal to 0 is the extreme case, in which only the original transmitter is awake. This makes the time consumed in the remaining data transmission minimal regardless of the network density. It shows that the time introduced with the handover support is much less than that without the handover support. The latency can be saved by at least $0.16s$ when the handover mechanism is applied. This figure can be even larger as the duty cycle and the network density increase.

6 Conclusion

In this paper, we proposed a seamless handover mechanism and evaluated the time required for transmitting a burst of packets with and without the handover support. It is asserted that the time can be saved by at least $0.16s$ when the handover mechanism is applied in the optimized RI-MAC protocol.

References

1. Dargie, W.: A medium access control protocol that supports a seamless handover in wireless sensor networks. Journal of Network and Computer Applications (2012)
2. Dong, Q., Dargie, W.: A survey on mobility and mobility-aware MAC protocols in wireless sensor networks. IEEE Communications Surveys and Tutorials (2012)
3. Li, W., Han, J.: Dynamic wireless sensor network parameters optimization adapting different node mobility, pp. 1–7 (March 2010)
4. Sun, Y., Gurewitz, O., Johnson, D.B.: Ri-mac: a receiver-initiated asynchronous duty cycle mac protocol for dynamic traffic loads in wireless sensor networks. In: Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems, pp. 1–14 (2008)

Optimized Service Aware LTE MAC Scheduler with Comparison against Other Well Known Schedulers

Nikola Zahariev[1], Yasir Zaki[1], Xi Li[1], Carmelita Goerg[1], Thushara Weerawardane[2], and Andreas Timm-Giel[2]

[1] ComNets, University of Bremen, 28359 Bremen, Germany
{nkz,yzaki,xili,cg}@comnets.uni-bremen.de
[2] ComNets, Hamburg University of Technology, 21073 Hamburg, Germany
{tlw,timm-giel}@tuhh.de

Abstract. Long Term Evolution (LTE) is the latest deployed broadband wireless technology from the 3rd Generation Partner Ship Project (3GPP) family. Nowadays there is an increased demand on the QoS requirements and inherently on further optimizing the usage of not only the scarce radio resources but also the overall network performance over the flat IP network more efficiently. Traditional scheduling algorithms are not able to fulfill such complex requirements. In this paper an Optimized Service Aware (OSA) scheduling algorithm is proposed, aiming at addressing the aforementioned challenges. It consists of three main functional units. The paper presents through simulations that the proposed OSA algorithm is able to achieve optimum radio and end user performance. The performance of the OSA is compared against other well-known scheduling algorithms: Weighted Blind-Equal Throughput (w-BET), Weighted Maximum Throughput (w-MaxT), and Weighted Proportional Fair (w-PF).

Keywords: OSA, LTE, Radio Resource Management, Scheduling.

1 Introduction

LTE is the latest system from the 3GPP family. The main requirements of LTE such as high data rate up to 300 Mbps and low latency along with short Transmission Time Interval (TTI) of 1 ms are summarized in [1]. LTE supports scalable system bandwidth ranging from 1.4 MHz up to 20 MHz. Another characteristic of LTE is that it is a completely packet-switched network for both real time and Best Effort (BE) services. As a result the system architecture has evolved to a flatter structure and is now called Evolved Packet System (EPS). LTE supports cost efficient end-to-end Quality of Service (QoS) with the "bearer" being the smallest granularity in the QoS classification [2], [3].

In this paper a novel scheduling algorithm – Optimized Service Aware (OSA) – is proposed and its performance compared against well-known schedulers such Weighted Blind-Equal-Throughput (w-BET), Weighted Maximum-Throughout (w-MaxT) and Weighted Proportional Fair (PF). The OSA scheduler consists of three parts: QoS Class Identifier (QCI) classification, Time Domain (TD) Scheduler, Frequency Domain (FD) Scheduler.

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 323–331, 2012.

2 LTE Scheduling State of the Art

The issue of defining an effective LTE scheduling algorithm is the subject of many papers in the literature nowadays. A joint combination of a time and frequency domain scheduler is proven to be the beneficial approach as given in [5], [6], [7], [8], where many different combinations of standard algorithms for the time and the frequency domain are proposed. In [9] the authors investigate the performance of several Time-Domain (TD) and Frequency-Domain (FD) scheduling combinations under Fractional Load. They show that a proportional fair scheduler provides a gain of about 4 dB against a Round Robin (RR) scheduler at 25% system load and that a combination of TD-PF/ FD-PF outperforms a TD-PF/ FD-Equal Resources (FD-ER) solution.

3 Optimized Service Aware Scheduler

The main target of the Optimized Service Aware (OSA) scheduler is to satisfy the QoS requirements for different traffic types while guaranteeing reasonable level of fairness and throughput. The OSA general framework is shown in Figure 1. By using the Differentiated Services (DiffServ) architecture, the scheduler is able to differentiate between the different traffic priorities and assign the radio resources in such a manner, that their individual QoS requirements are met. For instance, these QoS requirements at the application level include application end-to-end delay and application throughput. During the TD Scheduling the OSA scheduler creates a candidate list of bearers to be scheduled. It is able to differentiate between two main types of bearers - Guaranteed Bit Rate (GBR) and Non-Guaranteed Bit Rate (Non-GBR) and distribute all bearers into five different MAC QoS classes according to a scheduling metric. Two classes - MAC QoS Class 1 and Class 2 - are dedicated for the GBR bearers, while MAC QoS Class 3, 4 and 5 are for the Non-GBR bearers.

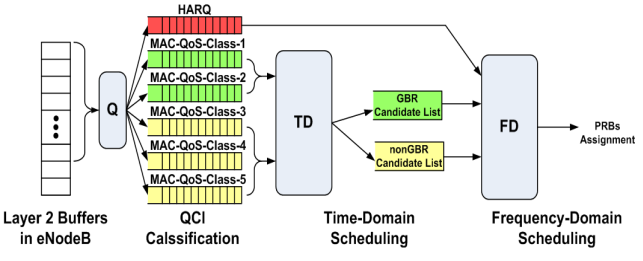


Fig. 1. Framework of the OSA Scheduler

3.1 QCI Classification

The first step in the OSA scheduler is the QoS Class Identifier (QCI) classification. Each TTI the scheduler checks the eNodeB buffer and the HARQ buffer of each user. If one of these buffers has data, the user is considered for scheduling within this TTI. The users with pending HARQ retransmissions are given the highest priority. Each

traffic type has different QoS requirements, thus each bearer is assigned to a single MAC-QoS class by mapping its QCI. In the proposed framework four different QCI classes have been considered, nevertheless this number can easily be extended. TABLE I shows the mapping of the different bearer types.

Table 1. Qci to MAC-QOS Class and to Weighting Factor Mapping

Bearer Type	Traffic Type	QCI Class	MAC QoS Class	W_{MQC}
GBR	VoIP	QCI-1	MAC-QOS-1	-
Non-GBR	Video Conf.	QCI-7	MAC-QOS-3	5
	HTTP	QCI-8	MAC-QOS-4	2
	FTP	QCI-9	MAC-QOS-5	1

3.2 Time Domain Scheduling

The Time Domain Scheduler sorts all active bearers in a prioritized candidate list and its task is to guarantee fairness between them. Bearers are picked from different MAC-QoS classes and are prioritized based on a Time Domain priority metric.

GBR Bearers: The Voice over IP (VoIP) service is transferred via a GBR bearer and is assigned to the MAC-QoS-1 class. VoIP traffic is characterized by small data rates around 10-15 kbps with very high requirements on the end-to-end delay. In order to meet these requirements, the VoIP (i.e. GBR) bearers are served with strict priority before the Non-GBR bearers. To ensure minimum end-to-end delay, VoIP bearer k is sorted according the following metric:

$$P_{k,GBR}^{TD}(t) = \arg\max_{k} t_p, \tag{1}$$

with t_p (p is for packet) being the HOL delay of the 1st packet at the bearer's buffer.

Non-GBR Bearers: The Non-GBR bearers can be used to transfer video conferencing and Best Effort (BE) traffic types. Only video conferencing traffic has a stricter QoS requirement in terms of packet end-to-end delay. In the TD the Non-GBR bearer k is sorted in the Non-GBR candidate list according to the following metric:

$$P_{k,NGBR}^{TD}(t) = \arg\max_{k} \frac{\hat{\gamma}(t)}{\hat{\Theta}(t)} \cdot W_{MQC}. \tag{2}$$

Here $\hat{\gamma}(t)$ represents an estimate of the channel conditions of bearer k at time instant t, $\hat{\Theta}(t)$ a throughput estimate of bearer k at time instant t and W_{MQC} a weighting factor. The channel estimate $\hat{\gamma}(t)$ is defined as a dimensionless value:

$$\hat{\gamma}_k(t) = \frac{\bar{\gamma}_k}{\gamma_{\max k}}, \tag{3}$$

where $\bar{\gamma}_k$ is a measure of the average SINR channel conditions of bearer k at time instant t. It is calculated using a moving average window as follows:

$$\bar{\gamma}_k(t) = \left(1 - \frac{1}{\tau}\right) \cdot \bar{\gamma}_k(t-1) + \frac{1}{\tau} \cdot SINR_{x_k}(t). \tag{4}$$

Therein τ denotes the size of the moving average window and can be tuned in accordance to the speed of the user. $SINR_{Xk}$ is a sample probe of the instantaneous channel conditions of one randomly chosen PRB x of bearer k. $\gamma_{\max_k}$ is a normalization factor used to obtain a normalized channel estimate $\hat{\gamma}_k$.

The throughput estimate $\hat{\Theta}_k(t)$ is also defined as a dimensionless value:

$$\hat{\Theta}_k(t) = \frac{\bar{\Theta}_k(t)}{\Theta_{\max_k}}. \tag{5}$$

The numerator $\bar{\Theta}(t)$ is the average throughput achieved by bearer k and is calculated again by applying a moving average window:

$$\bar{\Theta}_k(t) = \begin{cases} (1 - \frac{1}{\tau}) \cdot \bar{\Theta}_k(t-1) + \frac{1}{\tau} R^{BW}(t), & \text{if k is served in timeslot t} \\ (1 - \frac{1}{\tau}) \cdot \bar{\Theta}_k(t-1), & \text{otherwise.} \end{cases} \tag{6}$$

τ represents the length of the moving average window and $R^{BW}(t)$ is the instantaneous achievable data rate using the whole bandwidth. The normalization factor $\Theta_{\max k}$ is defined as the maximum throughput achievable by bearer k using all PRBs and assuming perfect channel conditions. The weighting factor W_{MQC} is used to prioritize the bearers according to their MAC-QOS class (MQC) and offers better bearer differentiation. TABLE I shows the weighting factor in accordance to the traffic type.

3.3 Frequency Domain Scheduling

The Frequency Domain Scheduler is the last step in the OSA framework. It has Round Robin-link nature and its main task is to efficiently assign the available PRBs among the different bearers that are sorted by the TD scheduler. The idea is to serve first all GBR bearers, since they have the highest QoS requirements. The scheduling algorithm for the Non-GBR bearers is shown in Figure 2. At the beginning of each

TTI a fixed number of N (configurable parameter) bearers are picked up for scheduling from the Non-GBR Candidate List. In the first turn all M bearers (all GBR + N Non-GBR), starting from the highest priority, reserve their best PRB. Having the best SINR value of that PRB one can get the maximum achievable Modulation and Coding Scheme (MCS) by looking at the 10% Block Error Rate (BLER) of the Additive White Gaussian Noise (AWGN) curves for the MCS. With the number of PRBs and the MCS the achievable Transport Block Size (TBS) can be found by using the Transport Block Size Table [10]. In the next step this value is compared to the current size of the PDCP buffer of that particular bearer. If the TBS is greater (i.e. all data in the buffer can be transmitted), then the bearer is served by assigning those PRBs.

If not, the bearer is put back into the queue by storing the PRB reservation and the achievable TBS for the next turn. The whole procedure continues until there are no more PRBs left or until all M bearers are served. If the latter occurs, the $N+1$ bearer from the Non-GBR Candidate List is taken. In the case when multiple PRBs are assigned to a bearer, a Link-to-System Mapping function is used to map the SINR values of the different PRBs to a single scalar, the so-called Effective SINR. For that purpose the Effective Exponential SINR Mapping (EESM) is used as given by [11]:

$$SINR_{eff} = -\beta \ln\left[\frac{1}{p}\sum_{q=1}^{Q}\exp\left(-\frac{SINR_q}{\beta}\right)\right] \qquad (7)$$

Therein Q represents the total number of PRBs, $SINR_q$ the Signal-to-Noise-and-Interference ratio of the q-th PRB and β [14] is a scaling factor (MCS dependent).

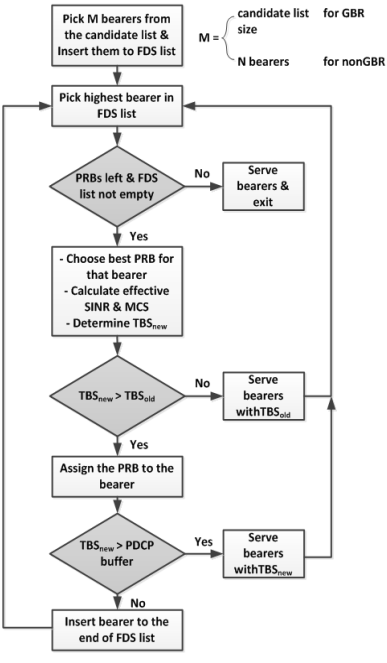


Fig. 2. FD scheduler flow chart

4 LTE Simulation Model, Configurations and Results

The LTE model has been simplified by focusing on the user-plane end-to-end performance analysis. The LTE core modeling is represented by the PDN-GW and aGW, and the E-UTRAN part is represented by the transport routers, eNodeBs and the UEs. All user-plane protocols in the simulator have been implemented according to the 3GPP release 8 specifications. In addition, a channel model, considering slow and fast fading, has been implemented [4].

To verify the performance of the OSA scheduler, simulations are performed with four traffic types – VoIP, video conferencing and Best Effort: HTTP and FTP. These services have different requirements and are carried by different bearer types. It is

important to mention at this point that video conferencing, in contrast to video streaming, cannot tolerate long delays due to the nature of the service. The simulations are performed with a mixture of different bearer types. Table II summarizes the main configuration parameters of the LTE simulation model. The simulation scenario compares the performance of the OSA scheduler against modified versions of the classical schedulers BET, MaxT and PF. The modification includes adding the QoS weighting factor W_{MQC} (TABLE I). It provides QoS differentiation, in order to fairly compare the schedulers. Their priority scheduling metrics are given as:

$$P_{k,BET}^{TD}[t] = \arg\max_{k} \frac{1}{\hat{\Theta}[t]} \cdot W_{MQC}, \tag{8}$$

$$P_{k,MaxT}^{TD}[t] = \arg\max_{k} R^{BW}[t] \cdot W_{MQC}, \tag{9}$$

$$P_{k,w-PF}^{TD}(t) = \arg\max_{k} \frac{R^{BW}[t]}{\hat{\Theta}[t]} \cdot W_{MQC}. \tag{10}$$

Table 2. Main Simulation Parameters

Parameter	**Assumption**
Cell Layout	Single Cell, 1 eNodeB with 5 MHz (~25 PRBs)
Channel Model	Macroscopic Pathloss model [12], Correlated Slow Fading [13] and Jakes-like Fast Fading model with user profile ITU-Veh. A.
Mobility Model	Random Way Point (RWP)
OSA Parameters	N=5 Non-GBR chosen for scheduling each TTI; τ = 1000; weighting factors–see TABLE III
HARQ	8 Processes with 10% BLER
VoIP Traffic Model	**10 users**
Silence/ Talk Spurt Length	exponential (3) sec with GSM FER encoder
Video Traffic Model	**20 users**
Frame inter. arr. time and size	0.1sec, frame size: 3200 bytes
HTTP Traffic Model	**10 users**
Number of pages per session	1 (with 1 object of size 100 KBytes in each page)
Reading Time	12 sec
FTP Traffic Model	**20 users**
Inter-request time/File Size	Uniform(30,60)sec with 3MByte file size

Figure 3 shows the simulation results of the different bearer types for all four schedulers in a spider web graph. The graph has four different axes, each representing a specific application delay performance: VoIP application end-to-end delay, Video application end-to-end delay, HTTP page response time and FTP file download time. Since all of the axes represent delays, then the smaller the spider web shape, the better

the performance is. One can observe that the end-to-end delay of the GBR VoIP bearers is the same for all four schedulers. This is to be explained by the fact that all GBR traffic is served with strict priority before the Non-GBR traffic and there are enough resources to serve all bearers.

It can be seen that for the video users OSA outperforms the other schedulers. The w-MaxT scheduler prefers users with good channel conditions and thus if a user experiences low SINR over a longer period, its performance will be poor while the w-BET algorithm will try to give similar priority to all video bearers, without considering whether the user has good or bad channel conditions. OSA has around 10 ms better performance than w-PF and the reason is that w-PF considers the instantaneous achievable throughput of the users in the current TTI, whereas OSA takes the average channel conditions of the user over a time period when calculating the priority metric.

When looking at the HTTP bearers it is clear to see that OSA performs better than w-BET and w-Max. It is interesting to observe that w-MaxT performs slightly better than w-BET. The reason for this is the bursty nature of the HTTP traffic. New data comes in segments and since the HTTP bearers are preferred before the FTP bearers, w-MaxT allows transmitting HTTP data faster by giving priority to the HTTP bearers with better channel conditions. w-PF performs better than OSA, because w-PF considers the instantaneous achievable throughput and is more aggressive to transmit data in one TTI and achieves higher spectral efficiency than OSA.

Looking at the FTP bearers it is not surprising to see that w-MaxT performs much better than the other schedulers, since bearers with higher instantaneous achievable rate are preferred. In that way FTP users are able to send much more data per TTI, thus, lowest file response time. When comparing OSA with w-PF, it can be observed that in case of OSA the FTP bearers have longer file download time.

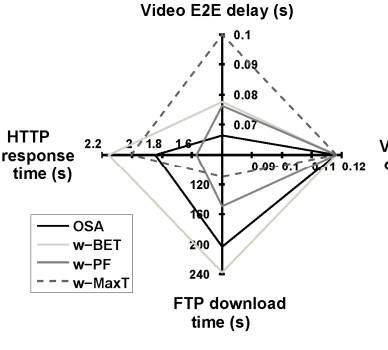


Fig. 3. Application delay performance

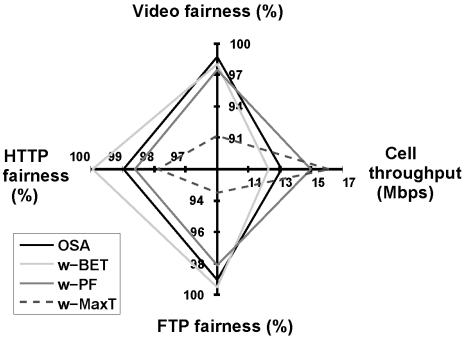


Fig. 4. Fairness and cell throughput

From the results we see that OSA offers better performance for users with constant offered load like video users, while w-PF is better for BE users like HTTP and FTP. Figure 4 shows the fairness and cell-throughput results. The fairness is measured in terms of how similar the user performance of each class is and is calculated as:

$$Fairness = 1 - {}^{1}\!/_{N} \cdot \sum_{i=1}^{N} \left| T_i - T_{avg} \right|. \quad (11)$$

Therein T_i is the individual per-user UU throughput, T_{avg} is the average of all per-user UU throughputs and N is the number of users.

Looking at the figure it can be seen that w-MaxT has worst fairness for all Non-GBR traffic but highest cell-throughput. The OSA scheduler shows very good fairness properties. Since there is always a trade-off between user fairness and cell throughput, OSA is able to successfully guarantee fairness at the price of cell throughput. It can be seen from the figure that w-PF has better cell throughput when compared to OSA while showing worse fairness performance. This is because w-PF prefers user with better instantaneous channel conditions, while OSA prefers users with on average good channel conditions.

5 Conclusion and Outlook

This paper proposes an Optimized Service Aware (OSA) scheduling algorithm that aims at satisfying the QoS requirements for different bearer types while guaranteeing reasonable level of fairness and achieving optimum end user performance. The performance of the OSA scheduler is investigated through extensive simulations using own developed LTE model. In this paper we compare the OSA scheduler against several other well-known schedulers: w-BET, w-MaxT and w-PF. From the performed simulations it is difficult to conclude which algorithm performs better. It depends on the situation, user requirements and the desired trade-off between real time services and spectral efficiency. When comparing OSA against w-BET and w-MaxT, we can prove that OSA is a compromise between the other two schedulers, offering a good level of fairness for the BE users. It gives priority to bearers with good channel conditions but at the same time tries to maintain equal amount of data. By comparing OSA with w-PF it can be seen that OSA gives better performance for users with constant offered load like video conferencing users. w-PF is better for BE users like HTTP and FTP, as OSA gives weight to users with good average channel conditions, whereas w-PF prefers users with higher instantaneous achievable throughput.

References

[1] 3GPP TR 25.913. Requirements for Evolved UTRA and UTRAN. V 2.1.0
[2] TS 36.211 Evolved Universal Terrestrial Radio Access (E-UTRA). Physical Channels and Modulation. Release 9 (March 2010)
[3] TS 36.201 Evolved Universal Terrestrial Radio Access (E-UTRA). LTE physical layer; General description. Release 9 (March 2010)
[4] Zaki, Y., Weerawardane, T., Görg, C., Timm-Giel, A.: Long Term Evolution (LTE) model development within OPNET simulation environment. In: OPNET Workshop 2011, Washington, D.C., USA, August 29-September 01 (2011)

[5] Pokhariyal, A., Kolding, T.E., Mogensen, P.E.: Performance of Down-link Frequency Domain Packet Scheduling for the UTRAN Long Term Evolution. In: IEEE 17th PIMRC (2006)
[6] Chung Beh, K., Armour, S., Doufexi, A.: Joint Time-Frequency Domain Proportional Fair Scheduler with HARQ for 3GPP LTE Systems. In: IEEE 68th Vehicular Technology Conference, VTC 2008-Fall, pp. 1–5 (2008)
[7] Monghal, G., Pedersen, K.I., Kovacs, I.Z., Mogensen, P.E.: QoS Ori-ented Time and Frequency Domain Packet Schedulers for the UTRAN Long Term Evolution. In: IEEE VTC, pp. 2532–2536 (2008)
[8] Pokhariyal, A., Pedersen, K.I., Monghal, G., Kovacs, I.Z., Rosa, C., Kol-ding, T.E., Mogensen, P.E.: HARQ Aware Frequency Domain Packet Scheduler with Different Degrees of Fairness for the UTRAN Long Term Evolution. In: IEEE 65th VTC, pp. 2761–2765 (2007)
[9] Pokhariyal, A., Monghal, G., Pedersen, K.I., Mogensen, P.E., Kovacs, I.Z., Rosa, C., Kolding, T.E.: Frequency Domain Packet Scheduling Under Fractional Load for the UTRAN LTE Downlink. In: IEEE 65th Vehicular Technology Conference, VTC, pp. 699–703 (2007)
[10] 3GPP TS 36.213. Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures, V10.2.0 (June 2011)
[11] Zhuang, J., Jalloul, L., Novak, R., Park, J.: IEEE 802.16m Evaluation Methodology Document (EMD) (July 2008)
[12] 3GPP TS 25.814 Physical layer aspects for E-UTRA. V7.1.0 (2006)
[13] Zaki, Y., Weerawardane, T., Timm-Giel, A., Görg, C.: Multi-QoS-aware Fair Scheduling for LTE. In: IEEE 73rd VTC 2011, Budapest, Hungary (2011)
[14] Valentin, S.: Chism – a wireless channel simulator for OMNET++. In: TKN TU Berlin Simulation Workshop (September 2006)

Achieving Energy-Efficiency with DTN: A Proof-of-Concept and Roadmap Study*

Dimitris Vardalis and Vassilis Tsaoussidis

Space Internetworking Center, Dept Of Electrical and Computer Engineering,
Demokritos University of Thrace, Xanthi 67100, Greece

Abstract. Mobile networking devices autonomy can be prolonged by condensing sporadic traffic at the last hop, allowing the receiver to *sleep* during *idle* intervals. We claim that Delay Tolerant Networking principles are a natural fit for this application and, along with a novel rendezvous mechanism, employ DTN to achieve energy efficiency. The effectiveness of the proposed scheme is supported by select evidence from our previous experimental work. The presented experimental evidence is followed by a detailed discussion on ongoing development work and future research directions.

1 Introduction

Mobile devices capable of connecting to the Internet through a wireless 802.11 LAN have become commonplace, even with non technologically-savvy users. The sophistication of these devices, along with the need to operate longer hours, creates an ever-increasing energy demand, not matched by the advances in battery technology. The networking subsystem of a mobile device has been identified as a major culprit in draining battery power, accounting for up to 60% of the total energy consumption in network intensive applications [1].

The need to save energy had been identified early in the development of 802.11 and, as such, energy saving provisions have become part of the standard [2]. The standard defines a *sleep* state for a Wireless Network Interface Card (WNIC), in which the device maintains its status as a member of the LAN, while energy expenditure remains very low. Switching the WNIC to sleep state in an intelligent manner has been a research focus in the past; it is, however, a novel contribution on our part to employ Delay Tolerant Networking (DTN) ([3], [4]) concepts in order to achieve energy efficiency. The proposed internetworking overlay exploits two major DTN properties: (i) storing packets for as long as it is necessary, regardless of communications disruptions; and (ii) enhancing the edge nodes with the functionality of collecting a sufficient amount of data, prior to transmitting to the end node.

* The research leading to these results has received funding from the European Community's Seventh Framework Programme ([FP7/2007-2013_FP7-REGPOT-2010-1, SP4 Capacities, Coordination and Support Actions) under grant agreement **n° 264226** (project title: Space Internetworking Center-SPICE).

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 332–341, 2012.

In this work we present select concepts and results from our past work on the subject ([5], [6]), as well as focus on ongoing work, future goals and promising ideas. Evaluation of the proposed schemes and mechanisms is made possible with the ns-2 network simulator. Energy efficiency is facilitated by a novel rendezvous mechanism, which takes advantage of the traffic shaping capabilities of DTN [6]. At the time of writing, we are in the process of developing a DTN agent for ns-2 that will allow for more sophisticated experimentation, and assist in addressing design and implementation issues beyond a proof-of-concept study. We have also implemented in the simulator a passive bandwidth estimation (BE) mechanism hoping to give greater visibility to the BS and thus allow for more efficient scheduling decisions.

The rest of the paper is organized as follows: In section 2 we present related work focusing on energy conservation, DTN, and bandwidth estimation approaches. In section 3 we present our proposal, as well as the simulation model used in the experiments, while in section 4 we include select experimental results. In section 5 we discuss ongoing work and future research directions and, finally, in section 6 we summarize our conclusions.

2 Related Work

In network intensive applications, a significant portion of the overall energy required for the device operation is consumed by the networking subsystem [1]. In [7] Jones et al. provide a comprehensive survey of specific mechanisms that can be employed in each of the layers in the network protocol stack. The 802.11 protocol [2] provides a mechanism that buffers incoming data at the BS, allowing the mobile devices to switch their WNICs to sleep state in the meantime. The energy conservation potential of this mechanism is limited by the relatively small buffer space at the BS and lack of visibility at higher network layers, leading many researchers into examining alternative methods based on the same core principle. In [8], Adams et el. propose a technique that buffers data at higher network layers, hiding it from the BS. Authors in [9] further develop the proxy idea, introducing a scheduler service at the Base Station and a proxy at the mobile terminal.

Delay/Disruption Tolerant Networking allows for wide-spread store-and-forward strategies that could extend data buffering and traffic shaping beyond the BS. The architecture for DTN was designed initially to facilitate packet-switched data transmission in space communications [3]. The deployment of the DTN Bundle Protocol [4] in space aims at seamless communication between network components on earth and space devices. However, researchers investigate applying DTN on networks with similar characteristics to space networks such as Mobile Ad-hoc Networks [10], Ad-hoc Sensor Networks [11], and highway networks [12]. However, to the best of our knowledge, DTN has not been evaluated as an overlay network for providing energy efficiency. Transmission scheduling at the BS can be greatly improved if bandwidth availability is known to DTN.

Estimating the available bandwidth has been a research focus in the networking community for over two decades; one of the first such efforts was packet pair probing

in the early nineties [13]. Bandwidth estimation techniques usually belong to one of two broad categories: active and passive. Active bandwidth estimation techniques inject probing traffic into the network, take certain measurements and use them to calculate available bandwidth. Passive bandwidth estimation in wireless networks take advantage of the broadcast nature of the communication in order to snoop on in-range transmissions and calculate idle periods of the medium. For single-hop, wireless networks such as the 802.11, many researchers suggest that passive methods are more pertinent than active ones ([14], [15], [16]). Generally, passive bandwidth estimation techniques are non-intrusive (i.e. do not burden the network with extra traffic), more responsive (i.e. no probing is required; channel utilization information is readily available), and more accurate than active techniques (i.e. active techniques rely on assumptions that do not hold for wireless 802.11 LANs [16]).

3 Energy-Efficient Internetworking Overlay

Our DTN-based, energy-efficient internetworking overlay takes advantage of the 802.11 feature that allows switching the WNIC to the *sleep* state [2]. While in the sleep state, the energy consumption of a WNIC is at least an order of magnitude less than the consumption in one of the active states (*transmit*, *receive*, *idle*) [17]. The BS buffers data at the last hop of an incoming data transfer, allowing the WNIC of the wireless receiver to be safely switched to the sleep state in the meantime. In this work we assume infinite buffers; postponing study of storage limitation issues for the future.

In order to take advantage of the idle connection intervals created by the buffering at the BS, we proposed a *rendezvous* mechanism between the BS and the wireless receiver [6]. When the buffered data is flushed, the receiver is notified of the next rendezvous time, switches its WNIC to the sleep state and wakes it up again in time to receive the next bunch of data. At every rendezvous the BS calculates the time interval for the next rendezvous, taking into account the incoming data rate of the previous interval and the target buffer occupancy set by the user (detailed explanation and numerical examples can be found in [6]).

3.1 DTN Overlay Simulation Model

The behavior of the bundle protocol was emulated by introducing a proxy application at the BS. The application connects to an input TCP agent, receiving data from the source of the data transfer, and an output TCP agent, transmitting data to the wireless receiver. Energy expenditure calculations were facilitated by modifying the physical layer of the wireless node, so that the state transitions are logged. Post-simulation processing computed energy consumption based on the state transitions and the effect of the rendezvous mechanism.

At post-simulation, the energy expenditure is calculated based on the following parameters: transmit power (txPower), receive power (rxPower), idle power (idlePower), sleep power (sleepPower), transition power (transPower) and transition time (transTime).

Idle intervals are converted to sleep intervals whenever possible. The power figures for the various WNIC states are set as follows [17]: txPower = 1.400 Watts, rxPower = 0.950 Watts, idlePower = 0.805 Watts and sleepPower = 0.060 Watts.

The diagram in Fig. 1 depicts the network topology used in the simulation. Network nodes are named as N1 – N6, with N4 being the BS node and N6 the wireless receiver. Links L13, L23, L34 and L45 are wired, while WL is the wireless link between the BS and the receiver. The data transfer follows the N1→N3→N4→N6 route, while the competing flow, when present, follows the N2→N3→N4→N5 route. The bandwidth and delay values for all the links are as follows: L13 - 2Mb, 100ms, L23 - 3Mb, 100ms, L34 – 300ms delay and varying bandwidth, L45 - 3Mb, 100ms, WL - 802.11 with a data rate of 11Mb and a basic rate of 1Mb. More on the experimental setup can be found in [6].

Fig. 1. Network Topology

4 Experimental Results

This section includes select experimental results from our previous work. The results in greater detail can be found in [6]. We present two setups, both of which experiment on an FTP connection when a competing, CBR flow is present. In the first setup the variable is the wired bandwidth, while in the second the variable is the Target Buffer Occupancy (TBO). In both setups there is an *End-to-End* (E2E) scenario, where an end-to-end connection between N1 and N6 is tested, and a *Split* scenario, where the connection uses the splitting application at the BS.

4.1 Varying Bandwidth FTP Transfer

In this set of experiments the backbone link is assigned a constant delay of 300ms, while the bandwidth varies from 0.5Mb to 3.5Mb in steps of 0.5Mb. The transfer duration of the E2E and the Split cases are virtually identical, spanning from around 360 seconds in the 0.5Mb setting to 90 seconds in the 3.5Mb setting.

Fig. 2 shows the energy consumption required at the mobile receiver for each transfer. It can be observed that small bandwidth values of the bottleneck link (leading to high congestion) produce greater energy gains when the Split application is employed. In the highest congestion setting (i.e. 0.5Mb bottleneck bandwidth), use of the Split application achieves approximately 64% energy conservation compared to the E2E case. As the bottleneck link capacity is decreased (network congestion eases) the energy conservation decreases as well, remaining however at significant levels.

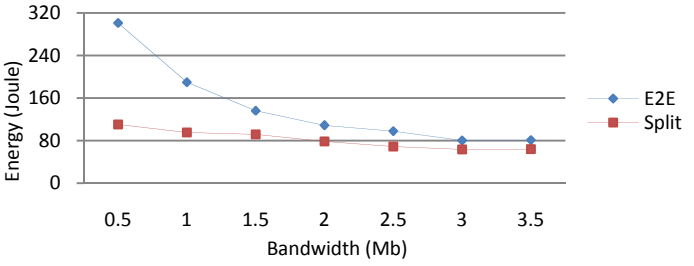


Fig. 2. Varying Bandwidth Energy Expenditure

Fig. 3 presents the state transitions for 10 indicative seconds during the file transfer in both cases, at a backbone bandwidth of 2Mb and with the presence of a competing flow. The Active state at the top includes sending, receiving and idle intervals (not adequately long for switching to sleep state), while the potential sleep state at the bottom includes idle intervals long enough to allow a switch to the sleep state. The chart clearly shows that in the E2E case, the WNIC needs to switch to active more frequently and it usually needs to remain active longer than in the Split case.

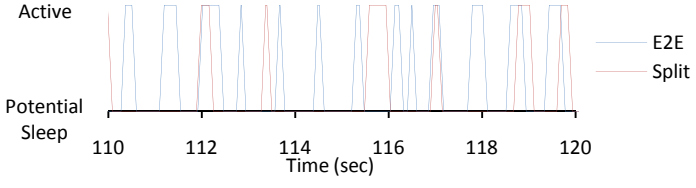


Fig. 3. FTP, 2Mb Bandwidth, Competing Flow State Transitions

4.2 Varying Target Buffer Occupancy FTP Transfer

The experimental setup of this section uses a 300ms delay and a 2Mb bandwidth for the backbone link; a competing flow is present and the TBO is varied from 0 to 180KB. The transfer duration for all TBO values is identical and, therefore, not reported here.

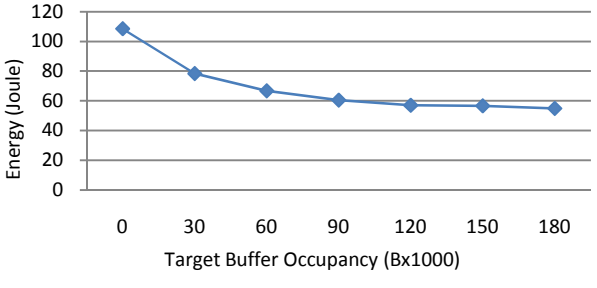


Fig. 4. FTP, Varying Target Buffer, FTP, With and Without a Competing Flow, Energy Consumption

Fig. 4 depicts the energy expenditure for all tested buffering values. As the TBO increases, the energy expenditure drops, reaching a value of around 55 Joule for the 180 Bx1000 case. The wired part of the connection is significantly slower than the last hop, so the wireless LAN is consistently underutilized, allowing for more efficient buffering. Fig. 5 depicts the actual buffer occupancy fluctuations throughout the 60 Bx1000 TBO data transfer. It can be seen that the buffer occupancy goes through two distinct phases during the transfer, bordering at around second 60. At the start of each phase, TCP is in slow start, and the incoming data flow increases sharply so the rendezvous mechanism takes a few seconds to respond. For the rest of the duration of each phase, the buffer occupancy falls within a 40 – 80 Bx1000 range, approximately 33% of the TBO. In a real-world situation where multiple devices receive data in parallel, the aggregate buffer occupancy could be predicted, allowing for efficient buffer planning.

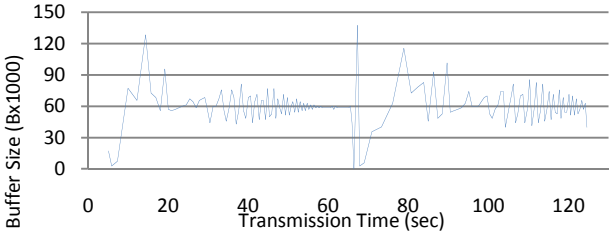


Fig. 5. Actual Buffer Occupancy at 60 Bx1000 Target Buffer Occupancy

5 Model Development and Future Work

5.1 Bandwidth Estimation Model

The motivation for employing bandwidth estimation techniques on the last-hop wireless link was to enable higher network layers exploit bandwidth availability information in order to make data transmission scheduling more efficient. For example, the DTN layer at the BS could dynamically adjust the amount of buffered data based on the available bandwidth. Another possible application of bandwidth estimation is enabling the transport layer to distinguish between congestion related packet drops and wireless error of random nature. The DTN layer at the BS could employ a specialized TCP version that would adjust its transmission strategy according to bandwidth availability information.

Latest trends in estimating bandwidth availability, as described in the related work section, clearly suggest that employing a passive bandwidth estimation method, in favor of an active, probing-based one, would be the most suitable approach in our case. A simple, passive bandwidth estimation mechanism has been implemented in ns-2. The mechanism is fitted into the wireless MAC layer of the BS and records busy

periods for the medium, as well as idle periods that are part of the collision avoidance strategy of the 802.11 protocol. Due to the nature of the wireless network and the inability of WNICs to transmit and sense for other transmissions simultaneously, 802.11 employs both a physical and a virtual Carrier Sense mechanism in order to minimize the probability of collisions. Our bandwidth estimation mechanism considers periods of inactivity that are part of the protocol collision avoidance strategy as busy periods. Thus, our algorithm takes into account: reception of segments, transmission of segments, back-off periods, Inter-Frame Spaces (i.e. Short IFS, Distributed IFS, etc.), and transmission deferral due to virtual Carrier Sense (i.e. Network Allocation Vector channel reservations). Details about the 802.11 protocol can be found in [2].

The busy and total duration amounts are stored in the first elements of two arrays and they get updated every time there is a switch in the state of the algorithm (i.e. busy-to-idle, idle-to-busy). A timer is set off at regular intervals and shifts the measurements in both arrays by one position to the right (it resets the first element and discards the last element of both arrays). The number of the time slots in the two arrays, as well as the time interval for the shifting of the values can be set by the user. When higher network layers query for the available bandwidth estimate, recent measurements have a higher contribution to the calculated utilization than measurements that are farther away into the past. Higher layers are passed a pointer to the MAC layer so that cross-layer communication of that sort can be realized. The calculated channel utilization figure is combined with the maximum channel capacity giving the available residual bandwidth.

5.2 DTN Agent Model

The experimental results presented in section 4 show that, in principle, application of DTN in conjunction with the rendezvous mechanism can lead to significant energy conservation in mobile wireless receivers. In order to expand our experimental work we are in the process of developing an ns-2 DTN agent that incorporates both a set of desired DTN-related characteristics, as well as the rendezvous mechanism. At the time of writing, a basic version of the agent has already been implemented, while new functionality is being continuously added.

The DTN agent will be deployed on multiple nodes along the network path (additionally to the BS) and enable studying issues such as: data storage distribution, route selection based on available buffer size, bundle sizing based on delay requirements. Incorporating the rendezvous mechanism into the DTN agent will allow for using the inherent energy model of ns-2, as opposed to the post-simulation calculations currently applied, solidifying the energy expenditure reporting. The DTN agent will also do scheduling in case multiple mobile end-nodes are receiving data at the same time and/or multiple flows are being directed to the same mobile end-node.

Achieving the desired DTN functionality implies that each DTN entity must have multiple incoming and multiple outgoing transport agents. As part of the current design, no routing functionality will be available at the DTN layer, so all routes must ultimately lead to the final destination (this requirement should be met during

topology setup). Outgoing agents will be sorted in priority order so that dynamic route selection will be possible. In this original design, route selection will depend only on available storage. Therefore, the DTN entity will select for each incoming bundle the outgoing agent with the highest priority that has available buffer space to store the bundle and, thus, accept custody for it. A DTN agent with a receiving application and no outgoing agents will be considered as the sink node and will notify the application of a received bundle instead of forwarding the bundle downstream. The agent receives data either from the attached application (if any), or from the upstream DTN agent to which it is connected. At the event of receiving a bundle segment, the DTN agent will either immediately forward it to the downstream node (cut-through) or wait for the entire bundle to be received before proceeding.

Storage-based routing assumes that each DTN entity must have knowledge of the buffer space availability of the downstream DTN entities in order for route selection to be possible. Exchange of storage space availability information calls for backward communication between DTN entities. Backward communication is also necessary for acknowledging bundle reception (i.e. accepting custody). Development of the backward communication is underway and can be realized both as a TCP acknowledgment piggy-back as well as a stand-alone DTN control bundle.

The new agent also needs to convey information on the forward path, in order to accommodate both the standard DTN functionality as well as the rendezvous mechanism. The rendezvous mechanism requires that the mobile host is notified about when each chunk of data has finished transmitting, as well as the time until the next rendezvous. Standard DTN as well as the rendezvous-related information is included in a new DTN header, created in ns-2. Among others, the DTN header contains information such as: bundle sequence number, bundle size, segment number, next rendezvous time, available buffer space, and custody acceptance. The DTN header is included in bundles travelling in both directions of a connection.

6 Conclusions

The goal of this paper was two-fold: i) presenting our DTN-based scheme for energy conservation, along with select results of the simulation experiments conducted thus far, ii) reporting on the ongoing work and future directions of our research. The experimental results provide adequate evidence that our proof-of-concept design is capable of enabling energy-efficient communication, without the need to sacrifice data transfer performance. Furthermore, the proposed rendezvous mechanism appears to be an effective, in-band means of communicating idle interval information, allowing the mobile receiver to safely switch its WNIC to sleep mode during periods of inactivity. The experimental results also advocate that the rendezvous mechanism promptly responds to incoming data rate fluctuations, facilitating efficient buffer planning. In the future, the response to fluctuations of incoming traffic can be further improved by utilizing historical data and employing more sophisticated prediction algorithms.

The prototype simulator design, used in the presented experiments, must be extended and encompass functionality necessary for deployment in real-world

environments. One such extension is the scheduling capability at the BS. The BS must be able to schedule data transmission from multiple flows destined to the same mobile receiver, as well as from multiple flows destined to multiple receivers in such a way that exploitable idle intervals can still be produced. Scheduling data transmissions at the last hop can also be benefitted by a passive, bandwidth estimation mechanism, a simulation model of which has already been created. The scheduling algorithm must also be able to accommodate real-time traffic, with certain delay limitations. Finally, we need to exploit the inherent DTN capability of distributing buffering storage over the network and, thus, relieve the BS of the pressure for excessive storage requirements. The ns-2 DTN agent that is under development will assist in studying all the above issues.

References

[1] Acquaviva, A., Simunic, T., Deolalikar, V., Roy, S.: Remote Power Control of Wireless Network Interfaces. In: Chico, J.J., Macii, E. (eds.) PATMOS 2003. LNCS, vol. 2799, pp. 369–378. Springer, Heidelberg (2003)

[2] IEEE Standard for Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications (2007)

[3] Cerf, V., Burleigh, S., Hooke, A., Torgerson, L., Durst, R., Scott, K., Fall, K., Weiss, H.: Delay-Tolerant Networking Architecture, RFC 4838, IETF (2007)

[4] Scott, K., Burleigh, S.: Bundle Protocol Specification, RFC 5050, IETF (2007)

[5] Vardalis, D., Tsaoussidis, V.: Energy-Efficient Internetworking with DTN. In: Masip-Bruin, X., Verchere, D., Tsaoussidis, V., Yannuzzi, M. (eds.) WWIC 2011. LNCS, vol. 6649, pp. 220–233. Springer, Heidelberg (2011)

[6] Vardalis, D., Tsaoussidis, V.: Energy-efficient internetworking with DTN. Journal of Internet Engineering 5(1) (2011)

[7] Jones, C., Sivalingam, K., Agrawal, P., Chen, J.: A Survey of Energy Efficient Network Protocols for Wireless Networks. ACM Journal on Wireless Networks 7(4) (2001)

[8] Adams, J., Muntean, G.M.: Adaptive-Buffer Power Save Mechanism for Mobile Multimedia Streaming. In: Proceedings of IEEE International Conference on Communications 2007 (2007)

[9] Zhu, H., Cao, G.: A Power-Aware and QoS-Aware Service Model on Wireless Networks. In: Proceedings of IEEE INFOCOM 2004 (2004)

[10] Ott, J., Kutscher, D., Dwertmann, C.: Integrating DTN and MANET Routing. In: Proceedings of the SIGCOMM CHANTS Workshop (2006)

[11] Li, X., Shu, W., Li, M., Huang, H., Wu, M.-Y.: DTN Routing in Vehicular Sensor Networks. In: Proceedings of IEEE Global Telecommunications Conference (2008)

[12] Ott, J., Kutscher, D.: From Drive-thru Internet to Delay-tolerant Ad-hoc Networking. In: Conti, M., Crowcroft, J., Passarella, A. (eds.) Mobile Ad-hoc Networks: From Theory to Reality. Nova Science Publishers, Inc. (2007) ISBN 978-60021-605-3

[13] Keshav, S.: A control-theoretic approach to flow control. In: Proceedings of ACM SIGCOMM (September 1991)

[14] Sarr, C., Chaudet, C., Chelius, G., Lassous, I.G.: Bandwidth Estimation for IEEE 802.11-based Ad Hoc networks. IEEE Transactions on Mobile Computing 7(10) (2008)
[15] Lee, H.K., Hall, V., Yum, K.H., Kim, K.I., Kim, E.J.: Bandwidth Estimation in Wireless Lans for Multimedia Streaming Services. Hindawi Publishing Corporation Advances in Multimedia 2007 (2007)
[16] Gupta, D., Wu, D., Mohapatra, P., Chuah, C.: Experimental Comparison of Bandwidth Estimation Tools for Wireless Mesh Networks. In: Proceedings of the 27th Conference on Computer Communications, IEEE INFOCOM (2008)
[17] Shih, E., Bahl, P., Sinclair, M.J.: Wake on Wireless: An Event DrivenEnergy Saving Strategy for Battery Operated Devices. In: Proceedings of the ACM International Conference on Mobile Computing and Networking (2002)

A Novel Security Architecture for a Space-Data DTN

Nathan L. Clarke[1,2], Vasilis Katos[3], Sofia-Anna Menesidou[3],
Bogdan Ghita[1], and Steven Furnell[1,2]

[1] School of Computing and Mathematics, Plymouth University, Plymouth, United Kingdom
info@cscan.org
[2] School of Computing and Security, Edith Cowan University, Western Australia
[3] Department of Electrical and Computer Engineering,
Democritus University of Thrace, Greece

Abstract. In this paper we reflect upon the challenges and constraints of a DTN infrastructure handling space data and propose a suitable security architecture for offering security services. The security requirements are expressed in terms of architecture components and supporting security processes. The architecture is provided as a point of reference for validating and evaluating future security controls and processes suitable for space data DTN environments.

Keywords. DTN, security DTN, secure communications, security architecture.

1 Introduction

Delay or Disruption Tolerant Networks (DTNs) are becoming popular both in terrestrial and deep space environments as they maintain certain advantages over traditional networking protocols such as TCP/IP. The benefits of adopting DTN technologies are clear in environments where connectivity in terms of end-to-end path availability cannot be guaranteed for the lifetime of a communications session.

Although DTNs by their nature support high availability, they are not short of security issues. This is primarily due to the constraints of the unwelcoming and hostile environment within which the communications take place. The three main limitations composing a typical space-internetworking environment are: the limited bandwidth, the relatively high bit error rates, and the periods lacking connectivity where in some cases open loop communication is the only option.

Security issues therefore arise in how to achieve end-to-end security of communications, with many standardised approaches being ineffective due to the high number of handshaking messages required to setup the secure channel. These would simply be not possible in a delay/disruptive network environment. Furthermore, issues regarding successful polices for enabling authentication, authorisation and accountability services within a Space-Data DTN exist. Indeed, little research has been published demonstrating how this can be achieved in reality.

The purpose of this paper is to propose a novel architecture to support the secure management and delivery of data across a Space-Data DTN. Section 2 describes the current state of the art, highlighting the unique threats present within a DTN and the

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 342–349, 2012.

advances made on developing a protocol for securing the channel. Section 3 and 4 present the novel architecture and secure data channel models, with a detailed explanation of both being provided. A discussion of these processes follows this, before Section 6 presents the conclusions and future work.

2 Background Literature

The area of security in Delay Tolerant Networks is relatively new and many research challenges remain to date. The DTN Research Group has published Internet-drafts on DTN Security Overview, Bundle Security Protocol Specification and Bundle Security Protocol Specification [1-3]. The Bundle Protocol (BP) exists within the DTN architecture and provides the capability of dealing with particular DTN characteristics, such as intermittent connectivity, custody retransmission and differing types of service delivery (e.g. scheduled, predicted and opportunistic connectivity)[2]. The DTN architecture [4] defines the "bundle layer" that may exist anywhere between the transport and the application layers of the OSI model.

The DTN Security Overview provides a useful insight into the possible threats faced within an DTN-based architecture [1]. The authors have identified the following potential threats: non DTN node threats; resource consumption; amplifying threats via forwarding bundles that were not sent by authorized DTN nodes; denial of service threats; attacks against the confidentiality and integrity of data; traffic storms (i.e. particular bundle protocol configurations allow for the generation of extra bundles) and partial protection (i.e. not all DTN nodes will have the ability to enact all security functionality).

The recently published Bundle Security Protocol Specification (BSP) states that addressing security issues is important for the Bundle Protocol (BP) [3]. The specification defines security features for the BP for use in DTNs. It specifically describes four security blocks to provide different security services. The four blocks, the Bundle Authentication Block (BAB), the Payload Integrity Block (PIB), the Payload Confidentiality Block (PCB) and the Extension Security Block (ESB), are defined in the Abstract Security Block (ASB). However, in the specification key management is not covered and the authors explicitly state that such exclusion is a result of an informed decision.

The author in [5] states some requirements for key management in delay tolerant networks but no solution is yet proposed. The internet draft [1] also provides an overview of the security requirements and mechanisms considered for DTNs security. More recently, two new internet drafts [6-7] extend the specification [3] and specify eight new Ciphersuites for use with the BSP's security blocks. However, until now few solutions have been proposed to address the security in DTNs. The work in [8] provides a security analysis of the RFCs and internet drafts with a focus on space-based communication networks. The author also identifies the problem that the management of security of the mission systems and the communication infrastructure is currently separate.

The authors in [9] introduce a solution based on Identity-Based Cryptography (IBC), a cryptographic method that enables message encryption and signature

verification using a public identifier. In [10], the authors use the non-interactive Sakai-Ohgishi-Kasahara (SOK) key agreement scheme, which is based on Boneh-Franklin IBC scheme. However, such IBC solutions appeared to superficially solve the problem. The work in [11] examines and identifies a number of problems and issues of the BP. They point out the lack of integrity checksums for reliability checks in the BP and the need for network time synchronization in order to increase the performance and reliability of the BP. Finally, a more recent study [12] addresses the key management problem in DTN by using one-pass authenticated key exchange protocol. The authors try to minimize the communication cost by using an adoption of Horsters-Mitchels-Peterson protocol.

To summarize, the Bundle Security Specification Protocol provides a baseline of cryptographic services for the bundle layer. The BSP supports flexibility and extensibility for the cryptographic mechanisms, allowing the relevant header fields to match the constraints and requirements imposed by the underlying environment or application domain. However, key management remains an open issue and would benefit from further research. Furthermore, little research exists on proposing how to achieve various other security requirements required within an operational DTN infrastructure. For instance, with regards to providing authentication, authorization and accountability (AAA) services.

3 Security Architecture

Based upon a set of analyses, which included deploying a stakeholder questionnaire to capture end-user requirements and expert analysis, the following architecture was proposed. The Space-Data DTN also includes an additional requirement beyond normal DTN systems in that it must support the long-term storage of large volumes space data within the DTN itself. The architecture is comprised of the following key components:

- Management Application (MA) – a web application that facilitates end users obtaining space-data. The application provides authentication, authorisation and accountability services
- Data originator (DO) – the original source of space data that is placed within the DTN. These components are assumed to be trusted.
- End-Users (EU) – the final destination of space data
- Trusted DTN nodes (TDTN)– a subset of the DTN nodes that are able to deliver space-data datasets

The term Trusted DTN is used in order to differentiate between the already defined Security DTN. The latter is defined by the Bundle Protocol and provides the communication security between security DTN nodes at the bundle layer. Trusted DTN nodes are security DTN nodes but also include additional functionality:

- Operating above the Bundle layer, they provide the functionality to store (and subsequently forward) complete datasets, rather than simply bundles (as defined by the Bundle protocol)

- Trusted DTN nodes have standard Internet-based communication capabilities with the Management Application – i.e. all management signalling information between the MA and TDTN conforms to standard internet based traffic conditions and is not subject to delay, disruption that a DTN network connection could be.

Fig. 1 illustrates the principal interactions of the key components within the Space-Data DTN. Contrary to typical DTN implementations, this architecture replies upon access to normal network communications in addition to the DTN. This capability permits the use of standard security mechanisms to protect key services – mechanisms whose operation could not be relied upon in a DTN where delay and disruption are present.

For simplicity and ease of understanding some DTN network connectivity between nodes is missing; however node "W" provides an indication of the interconnectivity of nodes within the DTN. The figure presents three different types of network connectivity for illustration. This is not a definitive set of connectivity but merely an example of the interactions between the principal components. Data Originators A, B and C are all storing their datasets within the DTN network – at the Bundle layer within both the Security and Trusted DTN nodes. Complete datasets are stored at the Trusted DTN nodes. Users A, B and C are also downloading datasets from the DTN network from the Trusted DTN nodes. In all three examples, data is sent within the DTN to untrusted DTN nodes with security being maintained between security and trusted DTN nodes (as specified by the Bundle Protocol security). The Management Application provides the mechanism for Trusted Nodes and Users to communicate and request datasets.

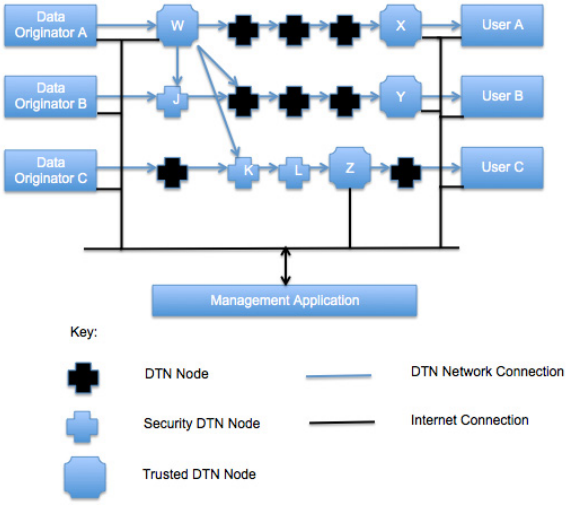


Fig. 1. Security Architecture Overview

Fig. 2 illustrates the network interactions that are sent when downloading space data from the DTN. A user requests a dataset by logging into the Management Application and clicking upon the available datasets. A one-time URL (with sufficiently long

freshness) is generated by the Management Application and sent to the most appropriate (frequently this would be geographically nearest) Trusted DTN nodes that is currently storing the dataset alongside additional information identifying the user. The same URL is then set to the User so that they can directly request the data across the DTN. All communication sent across the Internet-based network is secured. This process does rely upon a number of assumptions (which hold true):

- A process exists for datasets to be distributed from Data Originators onto the DTN.
- A process exists for the Management Application to be knowledgeable of where the datasets are distributed throughout the DTN.
- The Management Application, Users and Trusted DTN nodes can communicate via a normal Internet-type connection

In reality, the communication path indicated by the label (4) could be any combination of un-trusted DTN nodes, security DTN nodes and Trusted DTN nodes. Indeed, on some data requests, the user might find themselves a single hop from a Trusted DTN node with the necessary datasets. On other occasions, the datasets might need to transverse large segments of the network.

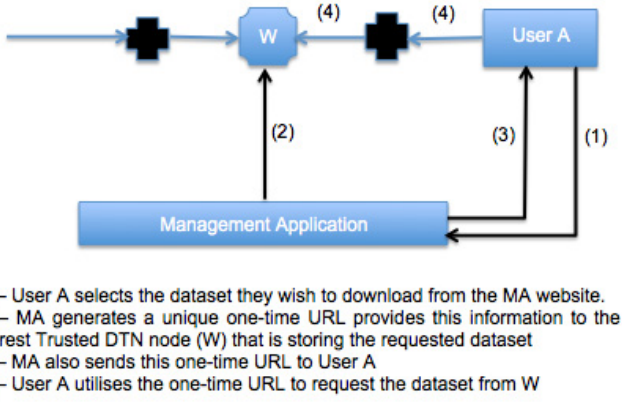


Fig. 2. Data Request Process

Based upon this architecture, there is a clear division between signals that communicate actual space-data and those that are concerned with management/control-based information:

- Space-Data transverses the DTN, is subject to delay and disruption but is capable of transferring large volumes of data reliably and securely.
- Management Data transverses the Internet, is not subject to delay and disruption and consists of relatively short volumes of data that enable efficient and secure operation of the Space-Data DTN.

The reasons for such a division reside with the capability of utilising existing security infrastructures within the Internet-based communications. Through being able to es-

tablish trust within key components of the network, the resulting threats are reduced and subsequent security mechanisms required can be taken from well-accepted standardised protocols (e.g. Transport Layer Security (TLS)).

4 DTN Data Security

The two main constraints influencing the design and deployment of the security mechanisms of a DTN infrastructure operating in a space environment are the limited bandwidth and the limited - yet in many cases predicted - connectivity between nodes. These constraints combined with the opportunistic data transfer approaches of DTN lead to the need for developing hybrid policies to effectively manage the trade off between security (i.e. the underlying computational and communication costs) and communication efficiency. As such, a data router must be equipped with the functionality to make routing decisions influenced by the security policy and needs.

In order to support efficient security mechanisms, key distribution consists of two key phases. The first phase involves computationally and communications intensive establishment of the long term key infrastructure. This can involve PKI components such as digital certificates. In addition, due to the limitations many devices may have in space (including power), low energy and memory consumption algorithms need to be considered, such as elliptic curve based PKIs.

The second phase refers to the secure session establishment. In this context, the term session depends on the security assertions and underlying scenario and is used to describe the situation where a node needs to create a confidential channel to some destination (not necessarily the final destination of the data, as end to end security cannot always be guaranteed or offered). Preference is given to one-pass security protocols.

The cryptographic keys and the cryptographic protocol metadata information will be transported using the Bundle Security Protocol specification. The BSP provides adequate flexibility to incorporate a wide range of key management protocols through the Extension Security Block (ESB) specified in BSP.

The BSP will also be used to support integrity services. Integrity in the space data layer is primarily offered by the Bundle Security Protocol when possible. In a DTN path that contains a mix of security aware and unaware nodes, integrity on the bundle layer will be verified whenever the custodian is a node capable of supporting BSP.

However there may be cases where the either the whole path is non BSP aware, or the integrity requirements are higher and the bundle layer integrity policies are not sufficient. Consider for example the case of remote firmware or operating system upgrades, where the upgrade instructions and firmware payload transferred to a deep space location will need to be both authenticated its integrity verified. In such a scenario, integrity will need to be offered by the application. Clearly in this case it is the MA which maintains the scope definitions of the data requiring integrity.

Finally, routing decisions are influenced by the security requirements of the underlying bundle and the data it holds. A router will need to implement a set of simple security and routing policies. A policy example is shown in Fig. 3.

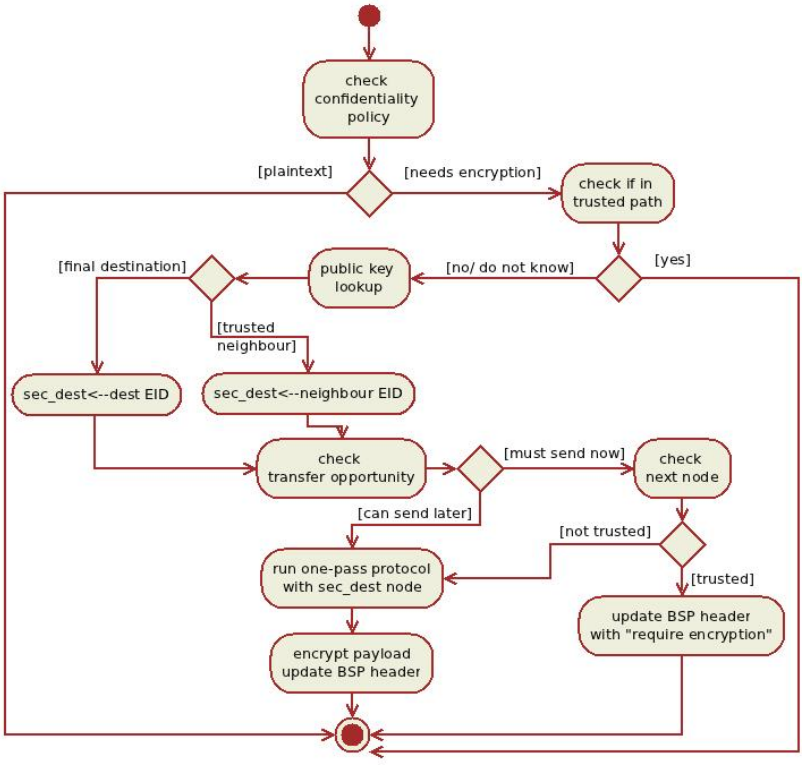


Fig. 3. Hybrid routing/security policy example (source: [12])

5 Conclusions and Future Work

The current state of the art clearly shows a significant lack in due consideration to both data-level security and to the operational security requirements. Given the set of security requirements, the security architecture has been proposed that has addressed the key requirements identified and protects against a wide range of DTN and non-DTN based threats that the system is vulnerable against. Key to this architecture is the use of both DTN and non-DTN networks that permit the use of a combination of both DTN specific security protocols and well-established (and thus accepted) security protocols typically found within secure internet-based services.

Future work will seek to validate the proposed architecture within an operational Space-Data DTN where it will be possible to evaluate the performance characteristics and usability of the proposed security mechanisms.

Acknowledgements. The research leading to these results has received funding fro the European Community's Seventh Framework Programme (FP7/2007-2013_FP7-SPACE-2010-1, SP1 Cooperation, Collaborative Project) under grant agreement no. 263330 (project title: SPACE-DATA ROUTERS for Exploiting Space DATA). This paper reflects only the authors views and the Union is not liable for any use that may be made of the information contained therein.

References

[1] Farrell, A., Symington, S.F., Weiss, H., Lovell, P.: Delay-Tolerant Networking Security Overview, internet-draft (2009), http://tools.ietf.org/html/draft-irtf-dtnrg-sec-overview-06

[2] Scott, K., Burleigh, S.: Bundle ProtcolSpecification, Request for Comments, RFC 5050

[3] Symington, S., Farrell, S., Weiss, H., Lovell, P.: Bundle Security Protocol Specification. Request for Comments, RFC 6257

[4] Cerf, V., Burleigh, S., Durst, R., Scott, K., Fall, K., Weiss, H.: Delay-Tolerant Networking Architecture, RFC 4838 (2007), http://www.ietf.org/rfc/rfc4838.txt

[5] Farrell, S.: DTN Key Management Requirements, work in progress as an inter-net-draft (2007), http://tools.ietf.org/html/draft-farrell-dtnrg-km-00

[6] Burgin, K., Hennessy, A.: Suite B Ciphersuites for the Bundle Security Proto-col, inter-net-draft (2012), http://www.ietf.org/id/draft-hennessy-bsp-suiteb-ciphersuites-00.txt

[7] Burgin, K., Hennessy, A.: Suite B Profile for the Bundle Security Protocol, internet-draft (2012), http://www.ietf.org/id/draft-hennessy-bsp-suiteb-profile-00.txt

[8] Ivancic, W.D.: Security Analysis of DTN Architecture and Bundle Protocol Specification for Space-Based Networks. In: Aerospace Conference, pp. 1–12 (2010)

[9] Asokan, N., Kostiainen, K., Ginzboorg, P., Ott, J., Luo, C.: Towards securing disruption-tolerant networking. Technical Report NRC-TR-2007-007 (2007)

[10] Kate, A., Zaverucha, G., Hengartner, U.: Anonymity and Security in Delay Tolerant Networks. In: 3rd International Conference on Security and Privacy in Communications Networks and the Workshops, Secure Communication, pp. 504–513 (2007)

[11] Wood, L., Eddy, W.M., Holiday, P.: A bundle of problems. In: Aerospace Conference, pp. 1–14 (2009)

[12] Menesidou, S.A., Katos, V.: Authenticated Key Exchange (AKE) in Delay Tolerant Networks. In: Gritzalis, D., Furnell, S., Theoharidou, M. (eds.) SEC 2012. IFIP AICT, vol. 376, pp. 49–60. Springer, Heidelberg (2012)

Cirrus: A Disruption-Tolerant Cloud

Eleftheria Katsiri[1,2,*]

[1] Department of Electrical Engineering and Computer Engineering
Democritus University of Thrace,
Xanthi, 67100, Greece
[2] Institute for the Management of Information Systems,
Research and Innovation Centre in Information, Communication and Knowledge Technologies, "Athena",
Artemidos 6 and Epidavrou,
Maroussi, 15125, Greece
eli@imis.athena-innovation.gr

Abstract. *Cloud computing* is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, virtual machines, applications, and services) that can be rapidly and *elastically* provisioned, to quickly scale out, and rapidly released to quickly scale in. However, commercially available cloud services such as public grids target the needs for the broader customer base and do not meet the specialized requirements of *real-time, data-centric* applications, such as sensor data aggregation, messaging, media streaming and commodity exchange, that need to process very large volumes of diverse, streaming data in near real time. To make matters worse, end-to-end communication paths between real-time data providers and consumers are no longer guaranteed, due to either node unavailability or service unavailability. The DTN paradigm has shown to promote interoperable and reliable communications in the presence of disruptions, however, is not directly applicable to cloud computing. A new cloud computing model is therefore needed for the above scenarios.

This paper proposes a novel concept, that of a generalized cloud, *Cirrus*, defined as a computing cloud with the following characteristics: (i) abiding by the NIST Cloud Definition, (ii) providing specialized, core Cloud services targeted to real-time, data centric applications, (iii) allowing for the elastic use of Cirrus cloud resources by ad-hoc networks and (iv) allowing for the elastic incorporation of nomadic and/or severely resource constrained devices, in Cirrus. Cirrus is built on top of DTN application-layer extensions, such as the Bundle Protocol (BP). As a result, Cirrus behaves as an "overlay Cloud", elastically forming, expanding and shrinking over networks of dynamic topology that may contain both fixed and ad-hoc infrastructure, thus providing a more fair and de-centralized Cloud Computing solution that is not exclusive to "big players" in the field.

* The author is Assistant Professor-elect at the Department of Electrical and Computer Engineering at the Democritus University of Thrace.

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 350–357, 2012.

1 Introduction

Real-time, data-centric applications such as environmental monitoring, traffic and transport monitoring and disaster management, but also social networking, file sharing and commodity exchange require specialized computational models that process very large amounts of (streaming) data in near-real time while at the same time trying to extract useful knowledge from the data, in order to best serve the user without compromising their privacy. In the data-centric paradigm, it is *timely and useful information* that is most highly valued while for energy-constrained devices, *prolonged device lifetime* also ranks high. This is a shift from the earlier computation-centric paradigm, such as scientific computing, where the focus is placed mainly on computational resources such as CPU and memory. The computational tasks involved in data-centric applications span from mathematical *aggregation* and *logical condition evaluation* to *fusion, data analytics, data mining* and *pattern classification*. Hybrid applications exist also, such as Smart-Grid [8] applications and certain scientific computing applications [11].

Playing a critical role in the world's computing and data storage requirements is *Cloud Computing* that has evolved into a model for enabling convenient, on-demand network access to a shared pool of configurable computing capabilities (e.g. networks, servers, storage, virtual machines, services, and applications) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This is known as *Infrastructure as a Service (IaaS)*. Users involved in the study of real-time applications need to re-think their strategies to process, share and store large datasets with the advent of this technological advancement. However, commercially available cloud services such as public grids target the needs for the broader customer base and do not meet the specific requirements of real-time applications such near real-time response, large-scale stream processing, semantic interoperability and often science-class performance and capacity requirements. Furthermore, the available applications that are provided as cloud services, known as *Software as a Service (SaaS)* are general-purpose and are not directly applicable to the computational needs of this domain. Concerns for data security, data governance and reliability have also been expressed in the literature. *Mobile clouds* [2,18] are also emerging, where the user can access cloud services from their smart phone or laptop, while being on the go.

To make matters worse, the proliferation and diversity of wireless communication technology in combination with processor and sensor miniaturization has led to inexpensive sensor networks and mobile devices that can be easily deployed at large scale, from deep space to deep ocean. On the one hand, this has increased both the amount and the diversity of available data, allowing for richer, more useful knowledge and better reliability, on the other hand it has lifted the assumption that end-to-end communication paths between real-time data providers and consumers are guaranteed, thus making the deployment of data-centric applications over these devices, unreliable, ad-hoc and uncoordinated, thereby infeasible in a systematic way.

Delay-Tolerant-Networking (DTN) [9] is a set of protocols that act together to enable a standardized method of performing store and forward communications, in scenarios where end-to-end connectivity cannot be assumed. DTN operates in two basic environments: low-propagation delay and high-propagation delay. In a low-propagation environment such as may occur in near-planetary or planetary surface environments, DTN bundle agents can utilize underlying Internet protocols that negotiate connectivity in real-time. In high-propagation delay environments such as deep space, DTN bundle agents must use other methods, such as some form of scheduling, to enable connectivity between the two agents. The convergence layer protocols provide the standard methods for transferring the bundles over various communications paths. Examples of environments where DTN has made significant contributions include spacecraft [12], military/tactical, some forms of disaster response, underwater, and some forms of ad-hoc sensor/actuator networks. It may also include Internet connectivity in places where performance may suffer such as developing parts of the world.

It is therefore clear that a different model is needed. It should be noted that our approach is inline with NASA's Mobile cloud, as defined in [18] however, our goals are different: Cirrus is focused specifically on the requirements of (personalized) streaming applications, while providing in addition support for environmental applications, social applications and commodity exchange, while NASA's mobile cloud targets mainly mobile phone users and has a much broader scope in terms of integration with existing clouds and applications.

1.1 Aims and Objectives

We define *Cirrus*, a generalized, Cloud Computing model, with the following characteristics:

1. **Abiding by the NIST Cloud Computing model definition** [3] that advocates five essential characteristics (*on-demand self-service, broad network access, resource pooling, rapid elasticity and measured service*) and three service models (*Cloud Software as a Service - SaaS, Cloud Platform as a Service - PaaS, Cloud Infrastructure as a Service - IaaS*).
2. **Providing specialized IaaS, SaaS and PaaS services for real-time, data centric applications.** Such services lay the foundations, i.e. the "plumbing" for data-centric applications that run on the cloud, as SaaS instances. They provide among others, a distributed Event Service, a distributed, federated Data Management Service, a Streaming Service with filtering and aggregation operators, a Personalization Service and a Semantic Interoperability Service. These IaaS services can be used by PaaS instances, in order to develop SaaS instances, i.e. applications, on Cirrus, such as a Personalized Twitter-like [17] Messaging Service, a location-aware Media Streaming service, a Commodity Exchange Service.
3. **Allowing for the elastic use of cloud services by ad-hoc networks.** This is the simpler interaction with the cloud. Here, ad hoc devices can use the above mentioned cloud services in order not to have to develop their

own. For example, a user equipped with a smart phone that can pick up data from ambient sensor networks, can store the data on the cloud. If the user is mobile, then the data should be stored in the nearest fixed-location on the cloud, in order to save battery resources.

4. **Allowing for the elastic incorporation of ad-hoc resources in the cloud.** This level of interaction allows ad-hoc and nomadic nodes to participate in the cloud resources by hosting a part of a Cloud service, resulting in cloud services that are distributed services over dynamic topologies of both fixed and mobile infrastructure. For example, by sharing a VM that is hosted on one mobile device, a second device's computational resources are pooled to those of the first one. The same applies to storage resources.
5. **Is built on top of DTN application-layer extensions, such as the Bundle Protocol.** However, several extensions are required at the Session and Application layers, to realize the above scenarios.

2 Research Challenges

The analysis of the application domain as well as the Cirrus characteristics of Section 1.1 raise a number of research challenges that are discussed in more detail in this section.

2.1 Abiding by the NIST Cloud Computing model definition

On-demand self-service *means that a consumer can unilaterally provision computing capabilities as needed, automatically without requiring human interaction with each service's provider.* We extend the NIST concept of cloud capabilities to include not only physical resources but also logical resources, i.e., Iaas instances, the "plumbing" that was mentioned earlier. We define a generalized virtual machine, the *Cirrus virtual Node - CN* that provides an abstraction over the above capabilities as well as the physical location where they are deployed and can be configured to provide any number of such capabilities on demand.

Broad network access *advocates that capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g., mobile phones, laptops, and PDAs).* A promising approach towards providing broad network access in Cirrus is to adopt the *Service-Oriented-Architecture -SOA* [6] paradigm for both the supported IaaS, SaaS and PaaS services. SOA provides an interoperable interface for all devices via the state-of-the art XML standard. Other approaches include the development of *Thin Client* software or *Smart Client* software.

Resource pooling *has to do with the fact that the providers computing resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand.* The Cirrus virtual machine serves this purpose well by being able to offer remote, often distributed capabilities, without disclosing their physical location. Another approach here would be to create a CN that supports parallel processing. This is an extended aim of this work.

Rapid elasticity *refers to the ability of rapidly and elastically provisioning capabilities, in some cases automatically, to quickly scale out, and rapidly releasing them to quickly scale in.* On one hand, elasticity is inherent in Cirrus that has the ability to scale on resources that were not previously available, such as mobile devices and sensor networks. On the other hand, in order to realize this feature, more research is required in order to investigate how to best implement elasticity at the VM (CN) level. One successful approach here is that of Amazon's Elastic Cloud (EC2) [5] where VMs can be composed in a flexible manner to form larger VMs and vice-versa. This in turn implies that the Cirrus VM (CN) also needs to be SOA-enabled, in order to be composable with other VMs.

Measured Service *implies an algorithm and its implementation for automatically controlling and optimizing resources use by leveraging a metering capability at some level of abstraction appropriate to the type of service.* In Cirrus, such a measuring service needs to be developed possibly as an embedded service inside the CN in order to monitor both physical and logical resources. One promising approach would be to develop an ontology of measurable resources that caters as much as possible for these differences. Furthermore, the Cirrus Measuring Service may have to aggregate individual results to produce an overall estimate.

2.2 Providing Specialized IaaS, SaaS and PaaS Services for Real-Time, Data Centric Applications.

Infrastructure as a Service - IaaS This is where the core, "plumbing" services belong to. In addition to the Cirrus virtual Node (CN) , IaaS services include but may not be restricted to a distributed Event Service, a distributed File Service, a distributed, federated Data Management service, a Streaming Service with filtering and aggregation operators, a Personalization Service [1] and a Semantic Interoperability service. ***Software as a Service - SaaS*** This category contains the cloud applications that aim to satisfy the specific computational requirements of data-centric applications, ranging from aggregation (of both numeric data, e.g. temperature values, and text data, e.g. *tweets*) to peer-to-peer commodity exchange. Other examples include a Twitter-like messaging application, a Map-creation visualization service, a Media Player service, a Commodity Exchange service. ***Platform as a Service - PaaS*** This category includes tools such as work flow editors that allow for the composition of both IaaS and SaaS services to form other applications, also available as SaaS instances. Examples of such work flow applications include a Personalized Twitter-like, Messaging Service, a location-aware Media Streaming service, a Commodity Exchange service and a privacy preserving Map service. [2]

[1] Personalization technology enables the dynamic insertion, customization or suggestion of content in any format that is relevant to the individual user, based on the users implicit behavior and preferences, and explicitly given details

[2] Such a service can process sensor data gathered by a mobile user by means of their smart-phone in a way that they can still be visualized in a map but without disclosing the identification and associated location of the users that gathered them, thus breaching user privacy.

Here as well, the SOA paradigm appears to be a promising solution for the creation of these services, their discovery and their composition in work-flows. The research issues therefore relate to the design and implementation of the above services so that they are *cloud-enabled*, i.e. service-oriented, providing an external, searchable interface and be distributable over more than one VMs or operate over distributed data. They also may need to be virtualized so that they can be used by multiple users or migrated closer to the data or the user, if required. Furthermore, the above services need to be composable with application services, thus providing added value on existing services.

2.3 Allowing for the Elastic Use of Cloud Services by Ad-Hoc Networks

In this level of integration user devices behave as Thin (or Smart Clients) that connect to the fixed cloud infrastructure either directly, or through a multi-hop path, using an appropriate routing scheme (e.g. epidemic routing). This model allows them to use Cirrus Cloud Services instead of developing their own custom solutions, thus allowing for correctness, standardization and flexibility. Clients can instantiate, use in an elastic manner and later deallocate VMs and associated capabilities on the fixed cloud. Client mobility and isolation both play a significant role in the way cloud services are designed. For example, a user with a laptop in a car that picks up ambient pollution data, can store the data continually on the cloud, each time at the current closest available location in order to save laptop battery. Similar arrangements may need to be made in the case of data streaming from isolated sensor networks that are only connected to the cloud periodically by means of satellite connection or when a user streams video clips or movies from optimal locations in terms of QoS, while traveling on a train. Note that these scenarios do not necessarily mean that other resources are in a data center other mobile devices could serve as resources (e.g. streaming video from nearest passenger) (see next Section).

In order to realize these scenarios, a federated Data Management needs to be designed, possibly using existing solutions such as RODs [1] or Hadoop [10]. Furthermore, nomadic and resource constrained devices should be DTN-enabled, while fixed infrastructure components need to be either DTN enabled (leading to a cloud where the interconnection is DTN-based rather than IP-based allowing for large scale integration with remote data centres) or connected to a DTN switch.

2.4 Allowing for the Elastic Incorporation of Ad-Hoc Resources in the Cloud

This level of interaction involves the hosting a part of a Cloud service by nomadic devices resulting in cloud services that are fully distributed over dynamic topologies of DTN-enabled nodes. For example, the Cloud's Distributed Storage Service can be built on top of mobile nodes allowing for the construction of distributed applications that use the MAP-REDUCE [15] paradigm. The Data

Aggregation Service as well as the Twitter-like Messaging Service can also be distributed. Aggregation and filtering algorithms can be integrated with the publish-subscribe[7] event paradigm and the performance of this "marriage" over an infrastructure-less DTN network will be investigated. Commodity Exchange involves the development of distributed algorithms that enable the exchange of goods in a fair-trade manner.

Inverting the problem, it is also interesting to investigate the placement of online stream processing and filtering tasks on special devices that behave like "data mules" running on a scheduled trajectory and providing connectivity between otherwise isolated components, i.e. UAV, satellites, submarines. Such devices are also relatively secure from threat, being physically airborn (or emerged).

In terms of research challenges, these include: a light-weight version of the CN hypervisor node (LCN) that can run on the above devices; an integration of the LCN and the DTN Bundle Protocol; investigating how the SOA paradigm can be applied to bandwidth-deprived nomadic devices scenarios (some work has already been done in the area of sensor networks, in the scope of an EU project [13]); An appropriate ontological approach trying to cater for semantic differences among data and services; novel privacy-preserving, forensic algorithms for the extended Cirrus cloud.

3 Conclusions and Future Work

Summarizing the above, mobile users of modern data-centric applications, expect computable, useful answers and they expect them "now"! At the same time, current technology suffers from resource poverty and lack of maturity. Although the Cloud industry is emerging it is only at the beginning of standardization. There exist too many choices of mobile devices and sensor networks, each supporting different operating systems and all prone to disrupted service. Cirrus aims to provide a solution for the above issues by investigating both middleware and algorithmic approaches. In terms of middleware we are considering: the design and implementation of the virtual Cirrus Node (CN) and the light-weight Cirrus Node (LCN), the integration of CN with a SOA technology, such as OSGI [4], the integration of CN with an event technology such as JMS [16], and an integration of all the above with the Bundle Protocol. Also we plan to investigate Thin or Smart Clients, and the composition/decomposition of VMs to allow for elasticity. In terms of algorithms, we are considering optimisations that related to performing distributed aggregation, personalized filtering and media streaming integrated with publish-subscribe over the DTN routing protocols, as appropriate, Map-Reduce processing algorithms over the same protocols, elasticity and service measurement algorithms, and interoperability mechanisms, privacy-preserving forensic algorithms. Cirrus's potential impact is significant. Apart from promoting quality and richness of computed knowledge, it enables functionality similar to that of the Internet of Things [14], such as Machine-to-Machine intelligence, augmented reality, location-based and personalized services.

Acknowledgments. This work was carried out under the auspices of the Space Internetworking Centre (SPICE) Project, which is led by Prof. V. Tsaoussidis of the Democritus University of Thrace.

References

1. IRODS:Data Grids, Digital Libraries, Persistent Archives, and Real-time Data Systems.https://www.irods.org/index.php/IRODS:Data_Grids,_Digital_Libraries,_Persistent_Archives,_and_Real-time_Data_Systems
2. Mobile Cloud Computing: Devices, trends, issues, and the enabling technologies, http://www.ibm.com/developerworks/cloud/library/cl-mobilecloudcomputing/
3. The NIST Definition of Cloud Computing. National Institute of Standards and Technology 53(6), 50 (2009)
4. OSGI Alliance, http://www.osgi.org/About/Technology
5. Amazon Elastic Compute Cloud, http://aws.amazon.com/ec2/
6. Booth, D., Haas, H., McCabe, F., Newcomer, E., Champion, M., Ferris, C., Orchard, D.: Web Services Architecture. Technical report, W3C (2004)
7. Eugster, P.T., Felber, P.A., Guerraoui, R., Kermarrec, A.M.: The Many Faces of Publish/Subscribe. ACM Computing Surveys 35(2), 114–131 (2003)
8. Smart Grid, http://energy.gov/oe/technology-development/smart-grid
9. Delay Tolerant Networking Research Group, http://www.dtnrg.org/wiki/home
10. Hadoop, http://hadoop.apache.org/
11. NASA Nebula in Action: Cloud Computing Case Examples, http://nebula.nasa.gov/media/uploads/nasa-nebula-in-action.pdf
12. Nichols, K., Holbrook, M., Pitts, R.L., Gifford, K., Jenkins, A., Kumzinsky, S.: Dtn implementation and utilization options on the international space station. In: SpaceOps 2010 Conference "Delivering on the dream", Huntsville, Alabama, Springer (April 2010)
13. Leguay, J., Lopez-Ramos, M., Jean-Marie, K., Conan, V.: An efficient service oriented architecture for heterogeneous and dynamic wireless sensor networks. In: 33rd IEEE Conference on Local Computer Networks, LCN 2008, pp. 740–747 (October 2008)
14. The Internet of Things. Executive Summary. Itu internet reports (2005), http://www.itu.int/osg/spu/publications/internetofthings/InternetofThings_summary.pdf
15. MapReduce: Simplified Data Processing on Large Clusters, http://research.google.com/archive/mapreduce.html
16. Java Messaging Service Tutorial, http://docs.oracle.com/javaee/1.3/jms/tutorial/1_3_1-fcs/doc/jms_tutorialTOC.html
17. Twitter, https://twitter.com/
18. Warner, S.A., Karman, A.F.: Defining the mobile cloud. NASA IT Summit 2010 (August 2010)

Reliable Data Streaming over Delay Tolerant Networks

Sotirios-Angelos Lenas [1], Scott C. Burleigh [2], and Vassilis Tsaoussidis[1]

[1] Space Internetworking Center (SPICE),
Democritus University of Thrace, Xanthi, Greece
{slenas,vtsaousi}@ee.duth.gr
[2] NASA / Jet Propulsion Laboratory (JPL),
California Institute of Technology, Pasadena, California, USA
scott.c.burleigh@jpl.nasa.gov

Abstract. Data streaming over Delay-Tolerant Networks (DTN) is a challenging task considering jointly the specific characterisics of DTN environments, the demanding nature of streaming applications and their wide applicability. Presently, there are not any advanced mechanisms available to support this functionality and typical configurations fail to efficiently transfer data streams. In this paper, we present our ongoing work in data streaming over DTNs and propose the Bundle Streaming Service (BSS) as a framework to improve the reception and storage of data streams. Our proposed framework exploits the characteristics of Delay Tolerant Networks to allow for reliable delay-tolerant streaming. Here, we present a simple usage scenario along with the proposed framework and evaluate it experimentally at a preliminary stage which, however, suffices to demonstrate its potential suitability for both terrestrial and Space environemnts.

Keywords: Data streaming, Delay Tolerant Networks, Interplanetary Internet.

1 Introduction

After several years of systematic research in various aspects of Delay/Disruptive Tolerant Networking (DTN) such as routing, transport protocols and convergence layers, DTN technology has reached a higher level of maturity. The development of a reliable set of working solutions and associated standards under the auspices of the Consultative Committee for Space Data Systems (CCSDS) and the Internet Research Task Force's (IRTF's) DTN research group [1] has boosted the applicability of DTN architectures, which now present themselves as prominent solutions for global internetworking. Based on that progress, several studies [2, 3, 4] promote the benefits of DTN architectures [5] and highly suggest their use in disruptive environments through the Bundle protocol [6], which encodes most functionalities that an overlay network requires.

Our work here deals with a relevant topic that has not yet seen much progress, despite its potential applicability: data streaming over DTNs. Data (and especially live) streaming in delay/disruptive tolerant environments becomes a particularly challenging task since the presence of high delays, frequent disruptions and variable

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 358–365, 2012.

bandwidth acts inevitably against the basic application principles of data streaming that call for mechanisms that guarantee smooth viewing experience of end-users.

In this paper, we present the results of our ongoing effort to provide a framework that enables the efficient management of real-time traffic in delay/disruptive tolerant environments in a manner that avoids the overexploitation of available network resources. In this context, we propose the Bundle Streaming Service (BSS) as a practical approach that addresses most of the networking challenges related to streaming over DTNs. BSS is a framework that enables "streaming" data to be conveyed via DTN "bundles" in a manner that supports in-order stream processing with minimal latency while still ensuring reliable delivery of all data to enable ad-hoc "playback" review of recently received information. Potential examples of real-time applications that could exploit the capabilities provided by this framework are one-way voice, video or continuous telemetry streaming.

BSS was designed with Interplanetary Internet (IPN) [7], and its associated issues, in mind. Despite its initial target though, our proposal could also fit in terrestrial delay/disruptive tolerant environments in which network nodes either follow a fixed predetermined route or move freely within a dynamic topology and hence experience occasional disruptions each time they move beyond the communication range; such networks exhibit properties similar to those of space internetworks, in terms of bandwidth capacity and connection availability, and hence may be serviced by a common solution framework.

The rest of the paper is organized as follows. In section 2, we present the related work and highlight the contribution of our proposal. In section 3, we analyze the operation of BSS and present its capabilities. In section 4, we describe a usage scenario that demonstrates how BSS can be exploited in a real world environment. In section 5, we detail the implementation of BSS and present some preliminary experimental results, both for terrestrial and Space environments. Finally, in section 6 we conclude this paper and discuss the framework for future work.

2 Related Work

Mobile Ad-hoc Networks (MANETs) are closely related to DTNs since they share several common characteristics such as network disruptions, high error rates and variable capacity links. A substantial amount of prior works that address several data streaming issues have already been proposed for MANETs. In general, the majority of the efforts are moving in two main directions; i) efficiency improvement and ii) redundancy. Among the most popular approaches suggested so far for improving efficiency are: i) the dynamic optimization of data coding, throughout the streaming session, so that the encoding bitrate does not surpass the available bandwidth of the network [8], ii) routing through multiple paths in order to increase delivery probability [9], iii) packet prioritization to minimize queuing delay and iv) specially adapted transport layer mechanisms that aim in reducing recovery delay of lost data. Redundancy on the other hand, is achieved through the use of FEC codes or by applying content summarization and error spreading techniques in order to provide error resilience. A few cross-layer approaches have also been proposed that combine several of the aforementioned techniques in order to enhance viewers' experience [10, 11].

Yet, to the best of our knowledge, the issue of data streaming in the context of DTN hasn't been studied extensively. Few initial works adopt similar approaches with the ones used in MANETs. In [12], T. Liu and S. Nelakuditi use erasure coding techniques in order to construct a disruption-tolerant video sequence so that in the event of disruption, helpful video content is still provided to clients by injecting additional "summary frames" to the original stream, while in [13], P. U. Tournoux et al. introduce Tetrys, a transport level mechanism based on an on-the-fly coding scheme which provides full reliability under the assumption that the encoding ratio used by Tetrys is higher than the average loss rate.

Due to the fact that the characteristics of each type of DTN may vary and the objective each time may be different, most of the aforementioned approaches cannot be applied in the context of delay/disruptive tolerant networking. In most cases, network functions such as routing, error recovery and congestion are usually located in the source or destination, a fact that mandates end-to-end connectivity. It's clear that any end-to-end approach cannot be considered due to the disruptive nature of DTNs. Therefore, each DTN node should be fully aware of how to handle a bundle that carries frames of a stream without the need for the application to run on every node. The use of FEC codes in the bundle layer also presents drawbacks since it might create conflicts with lower network layers, e.g. in cases that FEC codes are also used in the MAC layer. These conflicts usually lead to increased demand for network resources, mainly bandwidth, without guaranteeing any reliable delivery of the frames in cases where the coding rate is not sufficient to replace the losses imposed by the error rate of the communication channel. Finally, reliability should also be taken into consideration, especially for critical applications, which handle time-sensitive data.

A key observation that also inspired this work is that none of the aforementioned approaches exploit the most appropriate property of the Bundle protocol which naturally allows every DTN-node to temporarily store bundles. In that case, and based on the fact that disruptions are usually localized and experienced only by a few among many receivers, the retransmission effort could be minimal since it is limited to a certain area of the network. Considering this fact, the key concept behind BSS is to employ in the forwarding process of each DTN node both a best effort along with a reliable transfer protocol, in order to achieve minimal latency but also ensure reliable delivery of the whole stream. An additional advantage of our approach is that it does not confine future deployments of other sophisticated mechanisms on top of BSS, but instead, it grafts flexibility that further enhances synergistic application mechanisms.

Finally, as a design choice, we confine all network functionality of BSS within the bundle layer while preserving the closely related to the application functionalities, such as the re-ordering of packets, at the application level, in order to ease the burden of application developers.

3 Bundle Streaming Service Analysis

BSS consists of two basic components: a forwarder daemon and a library for building streaming-oriented applications. Figure 1, depicts the architecture of BSS.

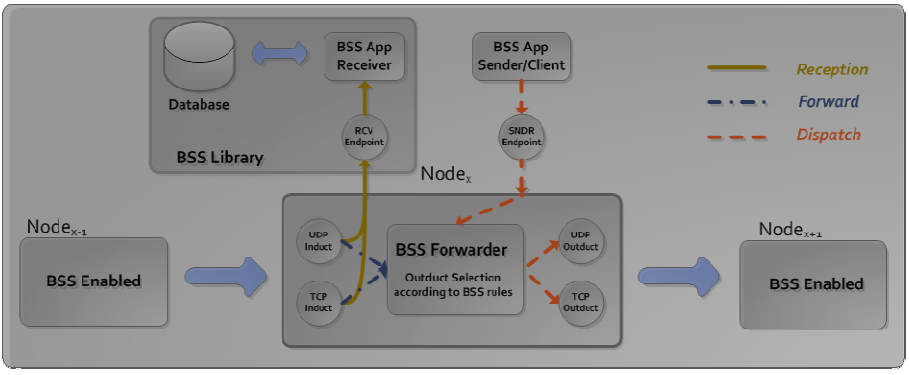


Fig. 1. BSS Architecture

Bundle creation time is the decisive criterion for all forwarding actions. BSS forwarder keeps track of the creation times of the bundles flowing from node X to node Y. Each of the forwarder's neighbors must have two inducts, one for a "best-efforts" convergence-layer protocol such as UDP (or "green" LTP) and one for a reliable convergence-layer protocol such as TCP (or "red" LTP). BSS selects the appropriate *outduct* for forwarding the bundle pending dispatch by applying the following rule: "Each bundle whose creation time is greater than that of any other bundle seen on this stream so far is forwarded to the "best-effort" *outduct*".

A prerequisite of BSS is that every bundle sent by a BSS-enabled node has to be custodially acknowledged. If a bundle's custody-accepted signal does not arrive prior to the timeout, the bundle is re-forwarded; in that case its creation time is not greater than that of any other bundle seen on this stream so far, so it is forwarded to the reliable induct of the next neighbor. Note that non-BSS forwarders will forward streaming traffic, but streaming display performance will be somewhat degraded by every node on the end-to-end path that is not running the BSS forwarder. Whenever the BSS forwarder does not identify the flow of bundles as BSS traffic, it treats them as normal traffic and forwards them accordingly as typical bundles.

The design of BSS ensures that eventually all bundles in the stream are delivered to their final destination, and beyond that, they are delivered in order and synchronized. The flow of streaming data never waits for retransmissions to succeed to avoid degrading the viewing experience of participants in a data streaming session. In the event that a bundle sent over the non-reliable convergence-layer protocol does not arrive at its next-hop node, that bundle simply will not be included in the flow of streaming data displayed at the destination. From this point on, it is identified as out-of-stream because its creation time is out of order. It eventually ends up at the destination, but because it's out-of-stream it does not get included in the displayed flow of streaming data; it just goes into the database at the destination, along with all of the successfully streamed data that has already been displayed. This enables the user to employ the "replay" features of the streaming service library to rewind back through the database and replay the data that were missed – merged with the data that originally were successfully received in transmission sequence; at the same time the current stream continues to be processed, e.g., displayed in another window.

The receiver's application is built using the BSS library, which initiates a background thread that receives all the bundles. Whenever that thread receives a bundle, it inserts the bundle into the BSS database (in creation-time order, for replay on demand) and it also checks bundle's creation time in order to decide, based on the above-described rule, if it will pass the bundle to an application-provided callback function for real-time display or to other stream processing. Meanwhile, the main thread of the application can be responding to user commands by calling BSS library functions that retrieve data from the time-ordered database for replay, with support for running *forward* or *backward*, *fast-forward*, *freeze*, etc.

The result of the above-described process is unimpeded real-time streaming of all the data that don't get dropped, together with comprehensive replay and review of all the data in the stream. And, because BSS does not operate by modifying bundle priorities, it can handle multiple concurrent streams of bundles at different priorities over the same links with notable flexibility: the high-priority streams will be closest to real-time, while the low-priority data will be available a little later.

4 Usage Scenario

In a real-world scenario BSS could be used for streaming video from the Moon to Earth. For simplicity, let's suppose the network topology is comprised of two nodes, the stream's source node (the camera) and the destination node (the user).

The user is sitting at a terminal on which three windows are presented. Window A is the real-time view from the camera. Window B is a GUI comprising VCR-like control widgets for replaying the video stream. Window C is the replay video view, controlled from window B.

Window A shows the view from the camera on the Moon exactly as it was 1.28 seconds ago (plus a few milliseconds of queuing, transmission, and processing latency), except that the view may "freeze" once in a while because one or more video frames were lost or corrupted somewhere along the end-to-end DTN path from the camera to the display. When there is such an outage, the missing frames never show up in this window; the displayed image simply remains unchanged until the next frame received in real time arrives. The view in this window is never delayed by any more than the one-way light time (plus processing, etc. latency), and it never regresses.

The user controls the replay display from window B, commanding the replay view to start N seconds ago and then roll forward or perform other available playback features such as pause, rewind, roll backward, etc.

Window C shows the replay view. This may be no more than the frames that originally were displayed in window A just a few seconds or minutes ago. But in the event that there were some outages in the real-time view in window A, the replay may show more than what was originally displayed. That's because the replay view includes frames that arrived out of ascending bundle creation time order at the user's terminal, due to retransmission of lost/corrupt frames or to arrival on different length paths. So, the replay view will always be at least as complete as the real-time view and it may be more complete; moreover, replaying a second time a bit later one may even reveal a more complete view as late-arriving lost bundles, which perhaps were lost again, finally arrive.

5 Implementation Details and Preliminary Experimental Results

BSS forwarder was implemented as a modified forwarder daemon, adapted from the standard "ipn" forwarder daemon included in ION [14]. Following implementation, our first goal was to establish the baseline performance characteristics of BSS over a simple streaming session under various network conditions, including variable propagation delays (PD) and high packet error rates (PER). Up till now, we have conducted some initial experiments, using a simple single-hop scenario (Fig. 2). The time needed for the complete reception of a stream consisting of 5000 frames was evaluated by using two basic sets of transport protocols in combination with BSS in order to evaluate its performance under terrestrial and Space environments.

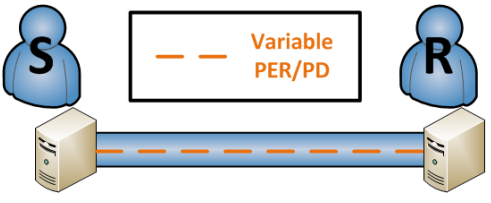


Fig. 2. String topology

The network stack was emulated on the DTN testbed of SPICE center [15] while the streaming process was simulated by developing *bssStreamingApp* and *bssRecv* applications. *bssStreamingApp* simulates the functionality of a media device that produces 30fps, 20866 bytes each. It also wraps each of these frames in bundles and hands them to the bundle layer for transmission. The transmission ratio achieved by *bssStreamingApp* is about 5000kbps which is considered more than enough to simulate the transmission of a high definition H.264 video quality stream [16]. At the other end, *bssRecv* is developed based on the API functions provided by BSS library. It presents two basic functionalities; firstly it immediately displays any in-order frames arriving at the destination; secondly, it saves in a specially-designed database the received stream, both in-order and out-of-order frames, in the appropriate order. The results of our initial experiments are shown in Figure 3.

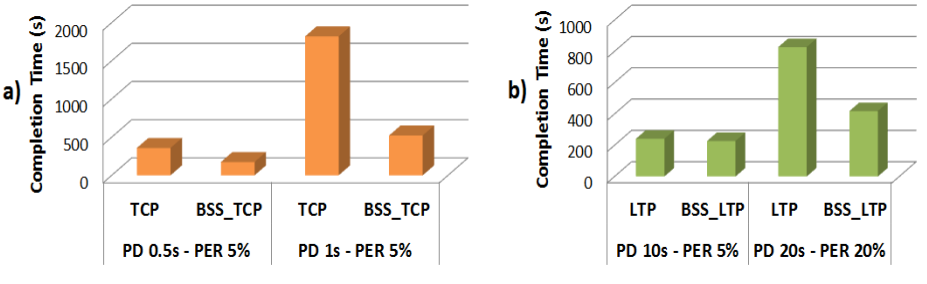


Fig. 3. Preliminary experimental results

Due to the poor performance that TCP exhibits in high error rate and long propagation delay environments, BSS manages to reduce the total requested time of receiving 5000 frames by almost 80% in the worst case. In Space environments, where LTP "red" transmission is used in place of TCP, BSS achieves better results only in cases where the error rate of the channel is above 10%. Furthermore, based on a different set of experiments that due to lack of space we cannot present here, we note another interesting property of BSS: it manages to reduce the total number of out-of-order received packets in comparison with the normal ION configuration using LTP alone.

6 Conclusion

In this initial phase of our study, a preliminary performance evaluation was conducted under various network conditions. The results obtained so far show that the suggested framework has the potential to improve stream reception in both terrestrial and Space environments.

As future work, we plan to extend this preliminary evaluation by conducting more tests and employing several other metrics, such as out-of-order delivered packet ratio, packet loss ratio, frame loss ratio and peak signal-to-noise ratio, in order to accurately assess the efficacy of BSS and evaluate the impact of frame size, hop count and mobility on its performance.

Armed with the knowledge acquired by these evaluations, we plan to proceed by modifying *bssStreamingApp* and *bssRecv* applications to support real video traffic, probably by importing/exporting video frames from/to VLC and deploying BSS in embedded devices in order to confirm our emulation results through tests in real wireless environments.

Acknowledgments. The research leading to these results has received funding from the European Community's Seventh Framework Programme ([FP7/2007-2013_FP7-REGPOT-2010-1, SP4 Capacities, Coordination and Support Actions) under grant agreement n° 264226 (project title: Space Internetworking Center-SPICE). This paper reflects only the authors' views and the Community is not liable for any use that may be made of the information contained therein.

References

1. Internet Research Task Force Delay Tolerant Networking Research Group, `http://www.dtnrg.org/`, `http://www.irtf.org/charters/dtnrg.html`
2. Farrell, S., Cahill, V., Geraghty, D., Humphreys, I., McDonald, P.: When TCP Breaks: Delay and Disruption Tolerant Networking. IEEE Internet Computing, 72–78 (2006)
3. Fall, K.: A delay-tolerant network architecture for challenged internets. In: Proceedings of ACM SIGCOMM (2003)
4. Burleigh, S., Hooke, A., Torgerson, L., Fall, K., Cerf, V., Durst, B., Scott, K., Weiss, H.: Delay-tolerant networking: An approach to interplanetary Internet. IEEE Communications Magazine 41(6), 128–136 (2003)

5. Cerf, V., Burleigh, S., Hooke, A., Torgerson, L., Durst, R., Scott, K., Fall, K., Weiss, H.: Delay–Tolerant Networking Architecture. Internet RFC 4838 (April 2007)
6. Scott, K., Burleigh, S.: Bundle Protocol Specification. RFC 5050 (November 2007)
7. Cerf, V., Burleigh, S., Hooke, A., Torgerson, L., Durst, R., Scott, K., Travis, E., Weiss, H.: Interplanetary Internet (IPN): Architectural Definition, http://www.ipnsig.org/reports/memo-ipnrg-arch-00.pdf
8. Qin, M., Zimmermann, R.: Improving mobile ad-hoc streaming performance through adaptive layer selection with scalable video coding. In: Proceedings of the 15th International Conference on Multimedia, MULTIMEDIA 2007, New York, NY, USA, pp. 717–726 (2007)
9. Calafate, C., Malumbres, M., Manzoni, P.: Mitigating the impact of mobility on H.264 real-time video streams using multiple paths. Journal of Communications and Networks 6(4), 387–396 (2004)
10. Zhao, Z., Long, S., Shu, Y.: Rate Adaptive Live Video Streaming in Manets. In: Communications and Networking in China, ChinaCom 2006, pp. 1–5, 25–27 (2006)
11. Delgado, G., Frias, V., Igartua, M.: Video-streaming transmission with qos over cross-layered ad hoc networks. In: Proceedings of SoftCOM 2006, Dubrovnik, Croatian, pp. 102–106 (2006)
12. Liu, T.,Nelakuditi, S.: Disruption-tolerant content-aware video streaming. In: Proceedings of ACM, Multimedia (2004)
13. Tournoux, P., Lochin, E., Leguay, J., Lacan, J.: Robust streaming in delay tolerant networks. In: Proc. IEEE ICC, Cape Town, South Africa, pp. 1–5 (2010)
14. Burleigh, S.: Interplanetary Overlay Network: Design and Operation V1.13. JPL D-48259. Jet Propulsion Laboratory, California Institute of Technology (2011)
15. Samaras, C., Komnios, I., Diamantopoulos, S., Koutsogiannis, E., Tsaoussidis, V., Papastergiou, G., Peccia, N.: Extending Internet into Space – ESA DTN Testbed Implementation and Evaluation. In: Granelli, F., Skianis, C., Chatzimisios, P., Xiao, Y., Redana, S. (eds.) MOBILIGHT 2009. LNICST, vol. 13, pp. 397–404. Springer, Heidelberg (2009)
16. High-definition video, Wikipedia (2012), http://en.wikipedia.org/wiki/High-definition_video

Space Mission Characteristics and Requirements to be Addressed by Space-Data Router Enhancement of Space-Data Exploitation

Ioannis A. Daglis[1], Olga Sykioti[1], Anastasios Anastasiadis[1], Georgios Balasis[1], Iphigenia Keramitsoglou[1], Dimitris Paronis[1], Athanassios Rontogiannis[1], and Sotiris Diamantopoulos[2]

[1] National Observatory of Athens, Athens, Greece
{daglis,sykioti,anastasi,gbalasis,ikeram,paronis,tronto}@noa.gr
[2] Democritus University of Thrace, Greece
sdiaman@ee.duth.gr

Abstract. Data distribution and access are major issues in space sciences as they influence the degree of data exploitation. The project "Space-Data Routers" (SDR) has the aim of allowing space agencies, academic institutes and research centres to share space data generated by single or multiple missions, in an efficient, secure and automated manner. The approach of SDR relies on space internetworking – and in particular on Delay-Tolerant Networking (DTN), which marks the new era in space communications, unifies space and earth communication infrastructures and delivers a set of tools and protocols for space-data exploitation. The project includes the definition of limitations imposed by typical space mission scenarios in which the National Observatory of Athens (NOA) is currently involved, including space exploration, planetary exploration and Earth observation missions. In this paper, we present the mission scenarios and the associated major SDR expected impact from the proposed space-data router enhancements.

Keywords: space science, spacecraft data distribution, internetworking, space communications.

1 Introduction

Vast quantities of space data have to be transferred from space to the operation centres and, beyond, to the research institutions in order to be analysed and exploited. The basic aim of Space Data Routers project is to allow space agencies, universities and research centres to share space data generated by a single or multiple missions, in a more flexible, secure and automated manner. Currently, efficient space-data exploitation faces two major obstacles: Firstly, research institutions have limited access to scientific data since their limited connectivity time to satellites directly confines their scientific capacity. Secondly, space-data collection centres, such as ESOC, lack sufficient mechanisms for efficient communication with interested end-users, let alone the lack of mechanisms for efficient data dissemination. The result is frequently quite

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 366–373, 2012.

disappointing: space data remain stored and unexploited, until they become obsolete or useless and consequently are being removed. In the context of space-data exploitation, the situation is expected to aggravate in the near future: space data volume will increase (consider the upcoming Sentinel missions, for example), but the mechanisms for disseminating and exploiting data are not yet in place. Therefore, the efficient exploitation and dissemination of space data should not be considered as a peripheral issue, but rather as an important missing mechanism from the European Infrastructure. The Space-Data Router implements a dual role: It increases communication flexibility in Space and forms a mission-/application-oriented communication overlay for data dissemination, on Earth. These goals will be realized in three stages:

- Design and implementation of a Space-Data Router; a crucial component for space internetworking.
- Integration of the Space-Data Router within a core existing testbed, tested and evaluated in terms of specific space mission scenarios' requirements, overlay architectural design, compatibility with ESA equipment, protocols and policies, scalability in communicating with deep-space components, and interoperability with NASA's equipment.
- Development of a pilot application to integrate thematically various practical space mission scenarios. A cross-mission approach will benefit scientific centres worldwide and also allow for more accurate and timely data analysis. The application will allow for investigating in depth the potential of creating thematically-oriented space-data overlays in the future.

The application scenarios, presented in this paper, have been properly selected to match with the scientific objectives of the project. The major scientific objectives along with the corresponding scenarios are for Space Data Routers to:

- Boost the dissemination capability for space data on Earth by extending end user access to space data through communicating Ground Stations and Space Research Centres.
- Allow for exploiting data from Deep Space and disseminating them naturally through unified communication channels.
- Exploit European Scientific Capacity as well as ESA's existing infrastructure, resources, protocols policies and assets.
- Allow for cross-mission scientific applications, by demonstrating the capability of the DTN space-data overlays to administer thematic cross-mission space-data.

The original scenarios include operational and application information, as they constitute the reference of comparison for the new architecture [14]. Additionally, the expected impact on each scenario by using the Space Data Routers based architecture is foreseen from the proposed SDR enhancement. In a later stage of the project, this impact will be evaluated in terms of data transfer, dissemination capacity, and operative burden.

2 Space-Data Routers Enhancements

The main advantage of the Space-Data Router is that it operates on top of existing network protocols and technologies, creating a DTN overlay [3] that interconnects networks with very diverse characteristics, such as space and terrestrial. Therefore

DTN provides the basic functionality for efficient space-to-earth data dissemination. In addition the router is developed on top of real space protocols allowing for the direct interoperation with current space infrastructure. Furthermore, a sophisticated application will also be implemented in order to support, highlight and assess system's capabilities.

A novel routing mechanism is being implemented to support communications in Space and on Earth. More specifically, we are designing a routing algorithm that is based on Contact Graph Routing algorithm [2] with several additions and modifications. In particular, we propose a cost calculation function that takes into account parameters such as the bandwidth, the delay, the financial cost, the node storage capability, and the node or owner agency's trustability. These parameters are included into the routing decision mechanism with specific weights that are configured based on specific mission objectives. For example, in a transmission session of vast amounts of data, storage limitations of intermediate nodes is an important parameter that should be given more weight than others. In essence, data will be optimally routed towards their final destination, depending on the specificities of each mission and each scientific instrument.

As far as the actual transmission of data is concerned, sophisticated transport protocols, such as LTP [12], are used to mitigate losses due to error-prone deep-space communication channels. Furthermore, we are in the process of designing and implementing a protocol for Delay Tolerant Payload Conditioning (DTPC), which is used as an end-to-end protocol on top of the DTN Architecture. DTPC protocol complements the DTN services offered by the Bundle Protocol [16] and operates only at the endpoints of the communication system enabling end-to-end services, such as application data aggregation and elision, in-order delivery of application data, end-to-end retransmission-based reliability and duplicate suppression.

Finally, a major technical contribution of the Space-Data Routers project is the implementation of a web-based application that interconnects data providers, user institutions, infrastructure providers and administrators. A graphical user interface will be developed, to facilitate easy and automated access to the application database. Given the wide user community, specific focus is given on providing a user-friendly environment that will allow for fast browsing between mission profiles. Furthermore, the cross-mission categorization will facilitate the interdisciplinary approach on many data sets.

3 Application Scenarios

3.1 Demonstrate the Capability of the Space-Data Routers to Extend End-User Access to Space Data. Case: Combining Real-Time/Archived AVHRR HRPT Data from Local HRPT Receiving Stations Worldwide

Description.

The Advanced Very High Resolution Radiometer (AVHRR), aboard the NOAA meteorological satellites, is a cross-track scanning system with five spectral bands having a ground resolution of 1.1 km and a frequency of earth scans twice per day. Each

pass of the satellite provides a 2399 km wide swath. AVHRR data are used for retrieving various geophysical parameters such as sea surface temperatures, energy budget, and vegetation content. Through the High Rate Picture Transmission (HRPT) service (1700MHz, at a transmission rate of 665,400 bps) installed on the NOAA satellites, user stations throughout the world can acquire data from three or more consecutive overpasses. In the specific scenario, as a demonstration, AVHRR data will be gathered from various ground stations and disseminated as a composite dataset in real-time via network nodes, which will incorporate the concepts and protocols of Delay Tolerant Networking. This scheme is similar to the one described in the currently running WMO's (World Meteorological Organization) RARS project which is focused on delivering NOAA ATOVS data (AVHRR and ATOVS sensors are both mounted aboard NOAA polar satellites) within no more than 30 minutes from acquisition [13].

SDR Expected Impact.
The expected impact is the increase of data availability and delivery throughput for real-time access of satellite data. Moreover, the deployment of the DTN nodes is expected to contribute to an effective utilization of the ground communication infrastructures, enhancing thus the data sharing mechanisms, circumventing the downlink constraints. At the same time, the scalability potential of the SDR concept will be assessed. Applicability of the approach to other types of direct readout broadcasting systems (e.g. MODIS) will be further examined.

3.2 Demonstrate the Potential of Exploiting Data from Deep Space and Disseminate It Naturally through Unified Communication Channels. Case: Hyperspectral Images Captured by MEx/OMEGA

Description.
MEx (Mars Express) is the first 'flexible' mission of ESA's long-term science exploration programme. Hyperspectral images are captured by the OMEGA sensor on-board MEx. The OMEGA data type is a hyperspectral image cube of ~120Mbytes size [5]. The European Space Operations Control Centre (ESOC) in Darmstadt communicates with MEx via ESA's New Norcia ground station (DSA-1) in Perth, Australia [7]. DSA-1 coverage depends on the actual distance between Mars and Earth. The Mars visibility window at DSA-1 is of the order of 10-12 hours. Within this visibility period there is a nominal tracking time of 8 hours per day, in which MEx communicates with DSA-1. Downlink communication sessions are continuous. Data are then transferred from ESOC/DSA-1 to IAS (Institut d'Astrophysique Spatiale, France) on a dedicated server, where they are decompressed, Planetary Data System (PDS) formatted and archived. Data processing is restricted to the IAS team during a proprietary six months period. After this period, the calibrated corrected image cubes become available to the scientific community through the ESA Planetary Science Archive. Currently, 5-6 image cubes are received per year at NOA via a slow ftp connection.

SDR Expected Impact.
The use of the proposed SDR architecture for MEx/OMEGA data is expected to:

- Increase the data volume received from Mars Express, by increasing the connectivity and downlink time of Mars Express with ground stations.
- Provide access to current (possibly raw) image cubes.
- Increase the access speed to high-volume data.

3.3 Demonstrate the Sufficiency of DTN Space-Data overlays to Administer Thematic Cross-Mission Space Data

Case 1: Multi-mission Study of the Sun-Earth Connection

Description.
The term "space weather" refers to conditions on the Sun and in the solar wind, Earth's magnetosphere, ionosphere, and thermosphere that can influence the performance, efficiency, and reliability of space- and ground-based infrastructure and can endanger unprotected humans in space conditions or above the Earth's poles. Nowadays, information from a single spacecraft vantage point can be replaced by multi-spacecraft distributed observatory methods and adaptive mission architectures that require computationally intensive analysis methods. Future explorers far from Earth will be in need of real-time data assimilation technologies to predict space weather at different solar system locations.

SDR Expected Impact.
The main requirements for this application scenario are the real-time availability of electric field, magnetic field and charged particle data as recorded by multiple missions in geospace and in the solar wind. The use of the DTN architecture could provide/improve:

- Real-time data acquisition from multiple missions for monitoring ULF/VLF wave occurrence and its effects on radiation belt dynamics.
- Successful data transmission even in harsh/challenged communication conditions.

The objective of this scenario is to now-cast and, ultimately, forecast the influence of solar disturbances (which propagate through interplanetary space and impinge on the terrestrial magnetosphere) on the development of electromagnetic waves in the magnetosphere and the wave effect on radiation belt variability.

We plan to test the capability of Space-Data Routers to efficiently distribute to registered end-users the relevant data streams from an ongoing NASA mission (THEMIS), an ESA mission (Cluster) and an upcoming NASA mission (Radiation Belt Storm Probes) [1,6]. The simultaneous real-time sampling of space plasmas from multiple points with cost-effective means and measuring of phenomena with higher resolution and better coverage will further address outstanding science questions.

Case 2: Land Surface Temperature

Description.
"LST" is a multi mission, single parameter case study. Knowledge of surface temperature and its temporal and spatial variations within a city environment is of prime importance to the study of urban climate and human–environment interactions [10, 11, 17, 18]. For the purposes of the SDR project, the satellites that carry thermal infrared sensors useful for the study of LST distribution are considered. Overall, three different spatial resolutions of 3km, 1km and 100 m, respectively, provide a different perspective to the study and characterization of the Urban Heat Island (UHI) phenomenon. In particular, 1km spatial and few images per day temporal resolution (e.g. MODIS, AVHRR and (A)ATSR) is an adequate compromise which gives the general picture of the hot spots and relevant patterns at a regional scale. If one wishes to investigate the phenomena in a finer scale, then one should use the high-resolution images (90/120m, e.g. Landsat TM and ASTER) for local/municipality level studies for long-term planning. However, the diurnal variation of the phenomenon is only possible with geostationary satellites (MSG-SEVIRI). Currently, one of the main problems is the different location of the data as they come from different providers.

SDR Expected Impact.

- The application allows for data gathering from multiple missions for one scientific objective
- Same storage location for all data as well as for real time and on demand.
- Ensure real-time data acquisition.

3.4 Demonstrate the Capability of Space-Data Routers to Deliver Efficiently to End-Users Vast Volumes of Data over Terrestrial Internetworks. Case: New Deployments in Space: Sentinels

Description.
This is an upcoming mission scenario. At the moment (2012), ESA is developing five new missions called Sentinels. Each Sentinel mission is based on a constellation of two satellites to fulfil the revisit and coverage requirements to provide robust datasets for the Global Monitoring for Environment and Security (GMES) Services. This scenario focuses on two Sentinels, Sentinel-1 corresponding to C-band synthetic aperture radar (SAR) applications [8,9] and Sentinel-2, which will provide high-resolution optical observation for land and emergency services [15]. The GMES Sentinels Flight Operations Segment (FOS) is being established at ESOC, Darmstadt, Germany. The Payload Data Ground Segment (PDGS) operation baseline imposes strong constraints on the product timeliness provided to end-users. The PDGS is designed as a network of distributed ground stations and complementary centres imposing product data exchanges amongst PDGS remote locations and between the PDGS and its users. This implies an adequate dimensioning of internal data circulation resources between centres (electronically or else), complemented by data dissemination resources between the various archive locations and end users (electronic only).

SDR Expected Impact.
The challenge of this scenario is that it refers to an upcoming mission and it calls for the systematic acquisition with a couple of twin satellites of all land surfaces. On ground, this implies a very large data volume to manage with appropriate processing, archiving and networking resources. As this data will be used, among other operation modes, for emergency situations SDR will be called to distribute vast volumes of data over terrestrial internetworks to the appropriate user locations on time.

4 Summary

The ultimate goal of "Space-Data Routers" is to boost collaboration and competitiveness of European Space Agency, European Space Industry and European Academic Institutions towards an efficient architecture for exploiting space data. The proposed approach in this project relies on space internetworking – and in particular in Delay-Tolerant Networking (DTN), which marks the new era in space communications, unifies space and earth communication infrastructures and delivers a set of tools and protocols for space-data exploitation within a single device.

Acknowledgements. The project "Space-Data Routers for Exploiting space data" has received funding from the European Community's Seventh Framework Programme (FP7-SPACE-2010-1, SP1 Cooperation, Collaborative project) under grant agreement n° 263330. This paper reflects only the authors' views and the Union is not liable for any use that may be made of the information contained therein.

References

1. Angelopoulos, V.: The THEMIS Mission. Space Sci. Rev. 141, 5–34 (2008)
2. Burleigh, S.: Dynamic Routing for Delay-Tolerant Networking in Space Flight Operations. In: SpaceOps 2008 Conference on Protecting the Earth, Exploring the Universe, Heidelberg (2008)
3. Cerf, V., Burleigh, S., Torgerson, L., Durst, R., Scott, K., Fall, K., Weiss, H.: Delay-Tolerant Network Architectur. IETF RFC 4838, Internet Engineering Task Force (April 2007), http://www.ietf.org/rfc/rfc4838.txt
4. Daglis, I.A. (ed.): Space storms and space weather hazards. Kluwer, Dordrecht (2001)
5. ESA Mars Express Operations, http://sci.esa.int/science-e/www/area/index.cfm?fareaid=9
6. Escoubet, C.P., Fehringer, M., Goldstein, M.: Introduction: The Cluster mission. Annales Geophysicae 19, 1197–1200 (2001)
7. ESA Operations and Situational Awareness - Mars Express operations, http://www.esa.int/esaMI/Operations/SEM0RMQJNVE_0.html
8. GMES Space Component Sentinel-1 Payload Data Ground Segment System Technical Budget, GMES-GSEG-EOPG-TN-08-0011, ESA (2009)
9. GMES Space Component Sentinel-1, Payload Data Ground Segment (PDGS) and Operations Concept Document, GMES-GSEG-EOPG-TN-08-0012, ESA (2010)

10. Hung, T., Uchihama, D., Ochi, S., Yasuoka, Y.: Assessment with satellite data of the urban heat island effects in Asian mega cities. International Journal of Applied Earth Observation and Geoinformation 8, 34–48 (2006)
11. Keramitsoglou, I., Kiranoudis, C.T., Ceriola, G., Weng, Q., Rajasekard, U.: Identification and Analysis of Urban Surface Temperature Patterns in Greater Athens, Greece, Using MODIS Imagery. Remote Sensing of Environment 115, 3080–3090 (2011)
12. Ramadas, M., Burleigh, S., Farrell, S.: Licklider Transmission Protocol – Specification. IETF RFC 5326, experimental (2008), http://www.ietf.org/rfc/rfc5326.txt
13. Regional ATOVS Retransmission Services (RARS), World Meteorological Organization, http://www.wmo.int/pages/prog/sat/rars_en.php
14. Scenario Requirements Report, Space-Data Routers (March 2011), http://www.spacedatarouters.eu/wp-content/uploads/2010/12/D21.pdf
15. Sentinel-2 Payload Ground Segment, System Technical Budget Document, GMES-GSEG-EOPG-TN-09-0031, ESA and Operations Concept Document, GSEG-EOPG-TN-09-0008, ESA (2010)
16. Scott, K., Burleigh, S.: Bundle Protocol Specification. IETF RFC 5050, Internet Engineering Task Force (November 2007), http://www.ietf.org/rfc/rfc4838.txt
17. Stathopoulou, M., Cartalis, C.: Downscaling AVHRR land surface temperatures for improved surface urban heat island intensity estimation. Remote Sensing of Environment 113, 2592–2605 (2009)
18. Weng, Q.: Thermal infrared remote sensing for urban climate and environmental studies: Methods, applications, and trends. ISPRS Journal of Photogrammetry and Remote Sensing 64, 335–344 (2009)

DTN-tg
A DTN Traffic Generator

Theodoros Amanatidis and Anastasios Malkotsis

Space Internetworks Ltd, Athens, Greece
{tham,anastasios.malkotsis}@spaceinternetworks.com

Abstract. Delay-Torelart Networking becomes a rapidly developing network technology in the area of Challenged Networks. Research community has made significant progress towards developing a set of protocols that enhance the reliability and efficiency of DTNs. Despite the broad acceptance of DTN architecture, there is not yet any reliable testing tool that could become the reference point of DTN protocol developers. In this context, we develop a testing tool called "DTN Traffic Generator - DTN-tg" that will be useful for every developer in the DTN area. DTN-tg generates multiple sessions of constant bundle rate traffic, allowing the testing and validation of any intermediate DTN node or a number of interconnected DTN nodes.

Keywords: Delay-Tolerant Networking, DTN Traffic Generator, testing tool.

1 Introduction

DTN architecture [1] has been widely adopted as the network architecture that addresses the problems of Challenged Networks [2]. Deep space communications, sensor networks, ad-hoc networks and underwater acoustic networks are typical cases of which DTN enhances the operation, as they are characterized by long delays, intermittent connectivity and high bit error rates. The core functionality of the DTN architecture is implemented by the Bundle Protocol (BP) [3], designed within the IRTF's Delay Tolerant Networking Research Group [4]. Bundle Protocol forms an overlay network on top of heterogeneous internets, implementing the store-and-forward mechanism, custody transfer mechanism, addressing and naming. During the evolution of DTN architecture, appropriate protocols of lower layers - following the OSI model - have been developed in order to address the needs of challenged environments and work effectively with the BP. On the transport layer, DTN-oriented protocols, like the Licklider Transmission Protocol (LTP) [5] and the Deep-Space Transport Protocol (DS-TP) [6], with their corresponding convergence layers, ensure the operation in environments where the traditional transport protocols of the Internet, TCP and UDP, do not cope well with the conditions of the stressed environments [7]. Although great effort has been expended to develop a reliable and efficient protocol stack, an equally reliable testing tool has not yet been developed and adopted by the research community.

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 374–379, 2012.

In Space Internetworks (SI) [8] we have recognized the need for a testing tool that will help DTN protocol developers to evaluate the reliability and performance of their protocols. In this context, we develop a testing tool called "DTN Traffic Generator" (DTN-tg), which will allow developers to test their protocol implementation, as well as the whole DTN protocol stack in connection with the hardware on which it is installed.

2 DTN-tg Functionality

The core functionality of DTN-tg includes the creation of multiple sessions of constant bundle rate traffic. Bundles are sent from a DTN-tg instance that acts as a client and they are received by another instance of the DTN-tg that acts as a server. The rate with which the bundles are sent is user configurable and remains constant during the testing session. Bundle rate is measured in kilobits per second (Kbps). Other user configurable options of a client that form the bundle traffic are: server's endpoint ID (EID), bundle's payload size in bytes, class of service, status report request flags and lifetime in seconds. On the other end, user is able to configure the server based only on the corresponding client's EID.

Generating constant bundle rate traffic, we are able to monitor the performance of a DTN node that is connected between two nodes equipped with the DTN-tg software (Fig. 1). One end acts as a client and the other end as a server, creating one or more streams of bundle traffic.

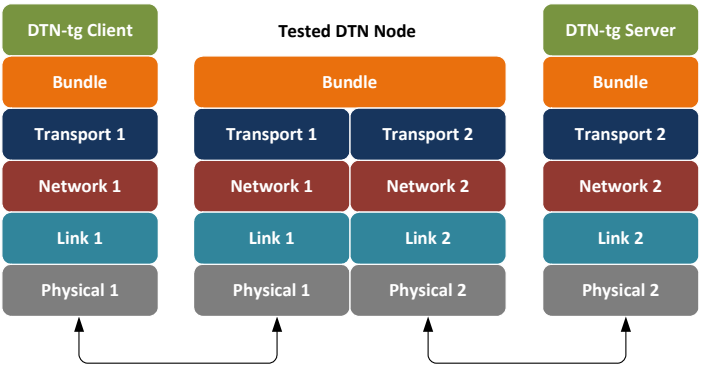


Fig. 1. A testing setup with the corresponding protocol stacks

The creation of multiple streams between the two ends allows for the emulation of bundle traffic generated by one or more applications with different specificities. In a real situation, the DTN node could be an Earth-orbiting satellite and the applications that generate bundle traffic could be an earth observation camera, a satellite's subsystems monitoring application or a deep-space relay system. In this way, we can monitor in real time the ability of the tested DTN node to forward the generated bundle traffic. More precisely, we are able to monitor the throughput of the bundle traffic that

is forwarded by the tested DTN node over time, and thus identify any performance-limiting factor that degrades the throughput. For example if a DTN node cannot forward all the bundle traffic that it receives because of the limited CPU performance or the low-rate link that connects it with the next node, it will have to store the received bundles on its internal memory. The number of stored bundles will increase over time, reaching the point of memory exhaustion.

The DNT-tg allows us to test both the hardware and software performance of a DTN node. The software implementation of the protocol stack can be a performance-limiting factor. Conducting experiments for the performance comparison of different DTN protocol stack implementations, using the same hardware setup, we can identify parts of the software code that reduce the overall performance. Some efforts have already been done to compare the performance of different DTN Bundle Protocol implementations, using the same hardware configuration during the experiments, and their results show the variation of the performance measurements of the different BP implementations [9]. With DTN-tg we can test and evaluate the software implementation of protocols of different OSI layers, for a given hardware platform. Additionally, we are able to detect hardware components of a DTN node which cause the overall system to underperform. For example, memory capacity and speed, HDD capacity and speed, CPU speed and network links capacity are factors whose impact on the overall system performance we can quantify with the DTN-tg.

3 DTN-tg Functional Components

DTN-tg consists of a DTN protocol stack and a Graphical User Interface (GUI) application. The functionality of the DTN protocol stack is provided by the DTN2 Reference Implementation [10]. The user will have to configure and compile the DTN2 source code, as it is not included in the DTN-tg software package. The GUI application is responsible for the configuration and creation of the bundle traffic as well as for the graphical representation of the bundle traffic in real time. DTN-tg can be installed in any kind of Linux-based PC or laptop, given that it is able to generate and consume the requested rate of bundle traffic and thus it will not be the bottleneck of the network.

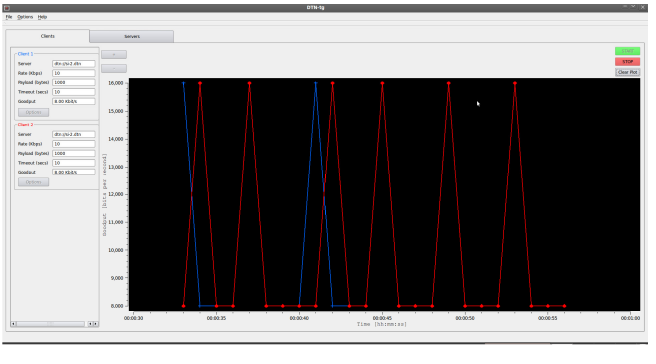


Fig. 2. DTN-tg GUI, Clients tab

The main GUI has two separate tabs: “Clients” and “Servers”. In the Clients tab, the user can add and remove a client. The maximum number of clients is currently restricted to five, for performance reasons. For each client, the user has to define: the corresponding server’s EID, the requested constant rate of bundle traffic in Kbps, the bundle payload length in bytes, the bundle Class of Service (CoS), the bundle lifetime in seconds and the requested bundle status reports. Two buttons called “Start” and “Stop” are responsible for the initiation and termination of all the clients respectively. When the clients are initiated, an individual curve with the corresponding color is drawn in the plot area for each client. The plot depicts the goodput over time, based on the received bundle status reports which were sent from the server and indicate that the bundle delivered to the server.

In the Servers tab, the user can add and remove a server in the same manner. The maximum number of servers is also restricted to five. For each server, only one setting is required: the corresponding client’s EID. The pair of the local EID and the corresponding client’s EID allows us to uniquely identify bundle traffic between two DTN nodes, and thus to build an extended topology with multiple instances of DTN-tg acting either as clients or as servers. Using two buttons, called “Start” and “Stop”, we can initiate and terminate all the servers respectively. Servers’ tab includes also a plot area where an individual curve with the corresponding color is drawn for each server. The plot depicts the goodput over time, based on the received bundles which were sent from the corresponding client. In this way, we are able to monitor in real time the achieved goodput and compare it with the rate that the user set in the client’s side.

4 DTN-tg Extended Functionality

DTN-tg can also be used in order to test a wider topology of DTN nodes. We can test any path of a network or the whole network by establishing multiple pairs of DTN-tg clients and servers.

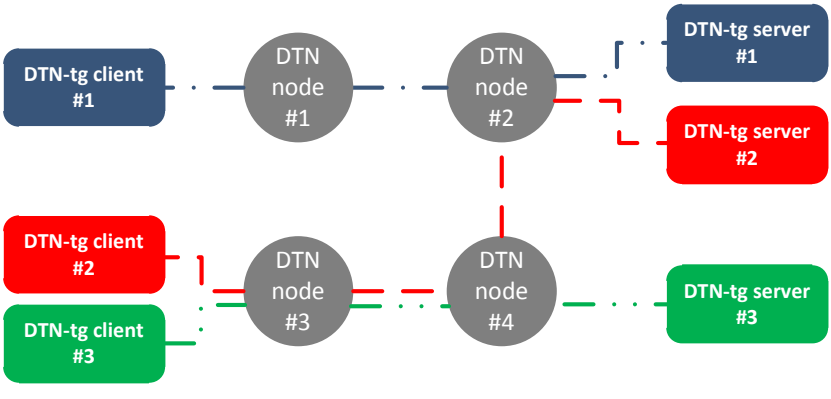


Fig. 3. An example of an extended topology

In Fig. 3 we have an example of a network with four DTN nodes and three bundle traffics that are generated between the three individual pairs of DTN-tg clients and servers. Each of the three bundle traffics can have its own characteristics. For example, they could have different bundle payload size and they could request different bundle status reports from the intermediate nodes, while keeping the same rate of bundle traffic. In this case, we could examine the impact of the bundle payload size on the network goodput as well as the impact of the increased bundle traffic due to the bundle status reports on the overall network performance. Additionally, in case of DTN nodes with different hardware capabilities, we are able to identify the impact of a resource-limited node on the bundle traffic. In summary, in case of an extended topology, DTN-tg allows us to test the ability of different paths of the DTN network to forward bundle traffics with different characteristics.

5 Future Work

One of our major next steps is to develop a version of DTN-tg which will work over Interplanetary Overlay Network (ION) [11]. We intend to keep the core functionality of DTN-tg and use the ION instead of DTN2 as the Bundle Protocol Implementation as well as the convergence layers and lower-layer protocols that ION provides. We also plan to evolve the Graphical User Interface in order to allow users to configure in more detail the bundle traffic and provide richer feedback to the user about the progress of the bundles through the network.

Acknowledgements. The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7-SPACE-2010-1, SP1 Cooperation, Collaborative project) under grant agreement n° 263330 (project title: Space-Data Routers for Exploiting Space Data). This paper reflects only the author's views and the Union is not liable for any use that may be made of the information contained therein.

References

1. Cerf, V., et al.: Delay-Tolerant Network Architecture, IETF RFC 4838, information (April 2007)
2. Fall, K.: A Delay-Tolerant Network Architecture for Challenged Internets, IRB-TR-03-003 (February 2003)
3. Scott, K., Burleigh, S.: Bundle Protocol Specification, IETF RFC 5050, experimental (November 2007)
4. Delay-Tolerant Networking Research Group, IRTF, `http://www.dtnrg.org`
5. Ramadas, M., Burleigh, S., Farrell, S.: Licklider Transmission Protocol - Specification, IETF RFC 5326, experimental (September 2008)
6. Papastergiou, G., Psaras, I., Tsaoussidis, V.: Deep-Space Transport Protocol: A Novel Transport Scheme for Space DTNs. Computer Communications 32(16), 1757–1767 (2009)

7. Akan, O.B., Fang, J., Akyildiz, I.F.: Performance of TCP protocols in deep space communication networks. IEEE Communications Letters 6(11), 478–480 (2002)
8. Space Internetworks Ltd., http://www.spaceinternetworks.com
9. Pöttner, W., et al.: An Empirical Performance Comparison of DTN Bundle Protocol Implementations, Technische Universität Braunschweig, Informatikbericht (2011)
10. DTN2 Reference Implementation, http://sourceforge.net/projects/dtn/
11. Interplanetary Overlay Network (ION), Jet Propulsion Laboratory, NASA, https://ion.ocp.ohiou.edu/

Author Index