

# SECOND ERCIM WORKSHOP ON EMOBILITY

Torsten Braun, Geert Heijenk,  
Dimitri Konstantas, Markus Wulff (Eds.)

Tampere, Finland, May 30, 2008



## Preface

ERCIM, the European Research Consortium for Informatics and Mathematics, aims to foster collaborative work within the European research community and to increase co-operation with European industry. In the ERCIM eMobility workshop, current progress and future developments in the area of eMobility should be discussed and the existing gap between theory and application closed.

This volume contains all accepted papers of the second eMobility workshop, which has been held in Tampere, Finland, on May 30, 2008. Papers from three main areas have been selected for this workshop. The contributions discuss several topics of the ERCIM eMobility working group, namely

- Transport protocols
- Multi-hop networks
- Services, user interfaces and mobility.

At this point, we want to thank all authors of the submitted papers and the members of the international program committee for their contribution to the success of the event and the high quality program. The proceedings are divided into two sections, full papers and short papers. While the latter present work in progress and ongoing research, the full papers have been carefully selected in a peer review process. The reviewers evaluated these papers and sent the authors the comments on their work.

The invited talk of the workshop by Antonio Peinado on “Speech-based user interaction for mobile devices” opened the session “Services, User Interfaces and Mobility”. The speaker gave a review of the technologies devoted to provide a faster, more comfortable and flexible user interaction with mobile devices through voice.

Tampere University of Technology (TUT) is the second-largest of the universities in engineering sciences in Finland. It is located in Tampere, which is a city in southern Finland located between two lakes, Näsijärvi and Pyhäjärvi. Here, we found an inspiring environment for a lot of intensive and fruitful discussions. The next ERCIM eMobility workshop is scheduled for 2009. We hope that a lot of our participants and many new colleagues will take this opportunity to continue exchanging their knowledge and experiences devoted to the development and use of eMobility.

Torsten Braun  
Geert Heijenk  
Dimitri Konstantas  
Markus Wulff

## **General chairs**

Torsten Braun, University of Bern, Switzerland  
Dimitri Konstantas, University of Geneva, Switzerland

## **TPC chairs**

Geert Heijenk, University of Twente, The Netherlands  
Markus Wulff, University of Bern, Switzerland

## **Technical program committee**

Francisco Barceló-Arroyo, Universitat Politecnica de Catalunya, Spain  
Raffaele Bruno, Italian National Research Council, Italy  
Giovanni Giambene, University of Siena, Italy  
Geert Heijenk, University of Twente, The Netherlands  
Jean-Marie Jaquet, University of Namur, Belgium  
Andreas J. Kassler, Karlstad University, Sweden  
Yevgeni Koucheryavy, Tampere University of Technology, Finland  
Edmundo Monteiro, University of Coimbra, Portugal  
Maria Papadopouli, University of Crete, Greece  
Antonio M. Peinado, University of Granada, Spain  
Vasilios Siris, University of Crete, Greece  
Dirk Stähle, University of Wuerzburg, Germany  
Do van Thanh, NTNU, Trondheim, Norway  
Mari Carmen Aguayo Torres, University of Malaga, Spain  
Vassilis Tsoussidis, Democritus University of Thrace, Greece  
Jean-Frédéric Wagen, College of Engineering and Architecture of Fribourg,  
Switzerland

# Table of Contents

---

## I Full papers

---

DTTP: a Delay-Tolerant Transport Protocol for Space Internetworks . . . .	3
<i>C.V. Samaras, V. Tsaoussidis, N. Peccia</i>	
Improving Sensor Network Robustness with Multi-Channel Convergecast .	15
<i>T. Voigt, F. Österlind, A. Dunkels</i>	
Evaluation of Routing Overlay Solution for a Hybrid Mobile ad hoc Network . . . . .	25
<i>J. Latvakoski, T. Aapaoja, J. Kärnä</i>	
Multi-hop Cellular Networks: Integrated IEEE 802.11 Ad hoc and Universal Mobile Telecommunications System (UMTS) Networks . . . . .	37
<i>A. Lo, J. Zhou, I. Niemegeers</i>	
Speech-based user interaction for mobile devices . . . . .	49
<i>A.M. Gómez, V. Sánchez, J.A. González, A.M. Peinado</i>	
Hovering Information: Infrastructure-Free Self-Organising Location- Aware Information Dissemination Service . . . . .	65
<i>A.A.V. Castro, G. Di Marzo Serugendo, D. Konstantas</i>	
Reducing Handover Latency in Future IP-based Wireless Networks: Fast Proxy Mobile IPv6 . . . . .	79
<i>G. Heijenk, M.S. Bargh, J. Laganier, A.R. Prasad</i>	

---

## II Short papers and abstracts

---

TCP Performance and Packet Aggregation in Wireless Mesh Networks under Low Link Quality . . . . .	95
<i>J. Karlsson, A. Kassler</i>	
TCP Performance over Wireless Channels with Unreliable ARQ . . . . .	99
<i>D. Moltchanov, R. Dunaytsev, Y. Koucheryavy</i>	
ATOM: Adaptive Transport over Multipaths in Wireless Mesh Networks .	101
<i>T. Staub, T. Braun</i>	
Analysis of cell residence time in WLAN under different mobility models .	107
<i>E. Zola, F. Barcelo-Arroyo</i>	
<b>Author Index</b> . . . . .	117



**Part I**

**Full papers**





# DTTP: a Delay-Tolerant Transport Protocol for Space Internetworks

Christos V. Samaras<sup>1</sup>, Vassilis Tsaoussidis<sup>1</sup> and Nestor Peccia<sup>2</sup>

<sup>1</sup>Department of Electrical and Computer Engineering  
Democritus University of Thrace  
12 Vas. Sofias Str., Xanthi 67100, Greece  
{csamaras, vtsaousi}@ee.duth.gr

<sup>2</sup>ESA/ESOC  
Robert Bosch Strasse 5, D-64293 Darmstadt, Germany  
nestor.peccia@esa.int

**Abstract.** We propose Delay-Tolerant Transport Protocol (DTTP) to address reliable data transfer in stressed network environments, such as space communications. Since existing TCP mechanisms do not work well (or at all) for such networks, new transport schemes are required. Intermittent connectivity in space environments calls for new transport approaches that smoothly adapt to the special networking conditions. DTTP is primarily a transport layer protocol and satisfies the inherent architecture requirements of Delay Tolerant Networking (DTN) in the absence of IP network infrastructure. It allows for reliable, efficient data transfer offering a number of application-oriented transmission strategies. Otherwise, when an IP architecture exists, DTTP operates as a stand-alone transport entity which interfaces with IP directly. We introduce the protocol's properties and functionality that enable its deployment in challenged networks. We conduct simulations that demonstrate the protocol's efficiency in scenarios with: (i) long propagation delays, (ii) minimum to relatively high packet error-rate, and (iii) intermittent connectivity.

**Keywords:** Delay-Tolerant Networking, Reliable Transport Protocol, Space Communications

## 1 Introduction

The success of space missions primarily relates to fulfilling their communication needs. However, as space missions increase and grow in complexity, there is an accompanying desire for interoperable missions. The Consultative Committee for Space Data Systems (CCSDS) [1] and other research groups active in the space communications area, are investigating ways to shift to such an interoperable communication platform [2]. Ability to exploit different combinations of spacecraft as data relays to Earth can, indeed, offer alternate communication opportunities; increase data return volumes; and reduce mission operating costs.

Delay-Tolerant Networking (DTN) architecture has been proposed as a means to interconnect heterogeneous network regions, which feature long message round-trip

times and/or interruptions in connectivity [3, 4]. DTN essentially conceals the underlying protocol stack and lays the communication standards for data exchange interoperability.

Other research endeavors concentrate on applying standard Internet technologies into space-based communication networks [5, 6]. The Internet paradigm provides a successful example of how heterogeneous networks cooperate and jointly offer services to each other. It is still under investigation whether proper adaptation of the Internet protocol stack is feasible for space communications as well. However, integrating the space-based infrastructure with the terrestrial Internet can seamlessly extend IP services in space, and spread the use of well-known Internet applications on-board spacecraft.

Whichever approach will eventually prevail (whether separate or combined deployment of DTN and IP, or even a different network scheme), an appropriate transport approach should be utilized. Space communication resources are limited, and are not expected to grow substantially in the foreseeable future. Along with the requirement for efficient use of scarce bandwidth resources, another major objective for space communications is reliable data transfer. Reliability is typically obstructed by high error rates present in space communication channels.

In this paper, we present Delay-Tolerant Transport Protocol (DTTP): a reliable transport protocol specialized for space communications. DTTP is important for space environments because it supports in-network storage that strengthens it against intermittent connectivity. Indeed, DTTP operates efficiently in intermittently available network paths where TCP-like transport approaches cannot even operate.

However, network entities (such as spacecraft, base stations on Moon or elsewhere, relay satellites, Earth ground stations, etc.) are the building blocks for internetworking into Space. On the basis of scheduled or predicted connectivity, and even during opportunistic contacts, stateful sessions take place. In such stateful connections, a DTTP agent is signaled as to when it can communicate to a peer node, and available information (such as the capacity of the communication link and the duration of the connection) is utilized by DTTP's rate-based transmission mechanism. Given that aggregate data return volumes for space-to-space or space-to-earth connections are limited, DTTP's ability to efficiently utilize individual links (as they become available) coupled with its parallel data transfer functionality, underline DTTP's importance, performance-wise.

In addition, DTTP's transmission behavior acts in an asynchronous manner, in contrast to standard TCP's strict compliance to closed-loop communication rules. The reason being that a space entity (at least in current predetermined space connections) solely uses a link (or a percentage of its capacity) for a certain period of time. Thus, DTTP acquires available bandwidth resources via its rate-based transmission behavior, and only adapts its sending rate upon receipt of explicit notification that congestion is present (possibly in the form of storage resources depletion). As a result, DTTP proves robust against packet error rates that can reach relatively high levels. In essence, DTTP primarily targets reliability, efficiency, and interoperability among space missions. We focus on space communications, though applicability of DTTP in other delay-tolerant network environments might be possible as well.

The remainder of the paper is organized as in the following. Section 2 presents work related to transport protocols for space communications. In Section 3, we de-

scribe DTTP features, potential network architecture, and implementation issues. Experimental evaluation of DTTP is presented in Section 4. Finally, Section 5 concludes the paper.

## 2 Related Work

TCP, the Internet's Transmission Control Protocol, is widely recognized to perform poorly across geostationary satellite links of around 0.5sec round-trip time [7]. This is due to the TCP's exponentially-growing probing of available capacity during slow-start, and its understanding of fairness as it interprets every packet loss as being a sign of congestion and thus slows its sending rate. TCP is shown to be less suited to larger delays due to the interaction of various timers present in TCP implementations [8]. TCP could be used in a direct Earth/Moon communication, though it could not effectively use the available link capacity. In greater distances and especially in scenarios characterized by intermittent connectivity, TCP essentially cannot function at all. In the following, we briefly review some transport approaches designed to function in space environments.

TP-Planet [9] is a reliable transport protocol designed for space environments that tries to capture the available link resources. It presumes the presence of IP infrastructure in space, and deploys a rate-based additive-increase multiplicative decrease congestion control. Yet, TP-Planet does not fully exploit available channel capacity because it *emulates* Slow Start and Congestion Avoidance algorithms of conventional TCP. TP-Planet is not tested over a multihop scenario but pertains to a single deep-space link. Also, it relies on conversational functions in order to set data rates. Thus, it implicitly expects continued bidirectional connectivity between the sender and the destination (which is not applicable in deep space communications).

SCPS-TP [10, 11] is the proposal of the Consultative Committee for Space Data Systems (CCSDS) for space communications. It consists a transport protocol based on TCP with appropriate extensions for space environments. SCPS-TP uses header compression and Selective Negative Acknowledgement (SNACK) options, to address limited bandwidth and provide more efficient loss recovery. When there is indication of congestion, either standard TCP or TCP-Vegas congestion control mechanisms can be used to provide congestion control. Otherwise, SCPS-TP operates in open loop rate control mode, where the transmission rate of outbound traffic is limited on the basis of a fixed rate parameter or on the basis of feedback from remote systems. In other words, SCPS-TP is based on existing TCP protocols with some modifications and extensions, which are shown to be inadequate for addressing the challenges in interplanetary networks (for example, TCP-Vegas cannot fully utilize space links due to its window-based nature and the apparent difficulty to accurately measure the variation in RTT for such long distances).

CFDP [12] is a protocol suitable for transmission of files to and from spacecraft data storage. In essence, it operates by copying files from a source storage medium to a target storage medium. Both unreliable and reliable (based on Negative Acknowledgements) services can be offered. The *core* functionality is the simplest form of operation and pertains to file delivery across a single link. The *extended* mode of opera-

tion refers to more complicated scenarios, and provides store-and-forward functionality across a network containing multiple links with disparate availability.

Authors of [13] propose LTP-T to target deep space applications. LTP-T uses the notion of *custody transfer*: each LTP-T entity must accept custody for all blocks it successfully receives. Custody is thus passed from host to host until the final destination is reached. This can minimize end-to-end data delivery time, though the requirement of accepting custody for all received blocks can also lead to storage exhaustion problems. The protocol supports delay-tolerant transport and congestion notification. However, LTP-T is only defined in a generic way, and further details of how it would operate in reality are not provided in the paper.

### 3 Delay-Tolerant Transport Protocol

#### 3.1 Features

Delay-Tolerant Transport Protocol (DTTP) provides reliable data transfer through challenged network environments. It is a transport protocol suitable for network environments characterized by intermittent connectivity, high bit-error rates, and long propagation delays. DTTP comprises a packet-oriented transfer approach that is designed to accelerate data transfers. Indeed, a fundamental design principle for DTTP is to improve data transfers' speed via efficient use of (possibly limited) communication resources.

Basic features of DTTP are summarized as follows:

(i) *Reliability*.

Though DTTP can operate in unreliable mode upon demand, ensuring and accelerating reliability is one of the core design goals for DTTP. Reliable transfer is assured by acknowledgments sent by the receiver. However, unlike TCP's ACK-clocking algorithm, DTTP exploits ACK information in an asynchronous manner: sending and acknowledgment processes in DTTP are *loosely* coupled when compared to standard TCP's behavior. In particular, acknowledgment procedures do not confine the sending rate, which is set autonomously based on bandwidth availability, presence or absence of storage capacity, congestion, and other network conditions. This property makes DTTP suitable for intermittently connected environments, such as space communications.

(ii) *Custody transfer*.

DTTP adopts the notion of custody transfer, upon which, the Delay-Tolerant Network architecture [3] is built. In essence, reliable transfer responsibility is delegated from original sender to next available intermediate node across the communication path, and this process can further be repeated until the final destination receives all data sent. Splitting the communication path into a distinct set of consecutive, reliable connections can serve a number of purposes; most notably, it alleviates senders from having to buffer large amount of data. Once data is received by next node on the path and custody is accepted, that data can be deleted from sender's buffer even if it has not reached its final destination. Consider, for example, the case of a robot on Mars with limited storage and energy resources that is able to dispense its data load to a

base station in range; or a rover on the Moon that distributes its data to a satellite crossing by. DTTP comprises a packet-oriented transport approach. Thus, in order to maximize efficiency, a packet that is accepted by next available custodian and which has yet to be acknowledged to the originating peer, can be immediately forwarded to the next-hop. This approach minimizes the time required for complete end-to-end delivery of the data.

(iii) *Parallel data transfer.*

Space agencies and industrial associates worldwide are involved in structuring standard technologies to achieve interoperable missions. In the years to come, space communications are expected to shift from current statically-organized communication sessions toward a more flexible network architecture. To exploit this upcoming networking framework, DTTP is equipped with the option of parallel data transfer. Thus data transfer can be accomplished in parallel data paths, exploiting various communication opportunities. Sequence of application data is resumed at the receiver. Parallel data transfer can be implemented by preserving the original sequence number space. For instance, assume a rover on the Moon is about to transmit a file (partitioned in 10,000 packets) to a ground station on Earth, and decides to accomplish this via two separate orbiting satellites. The rover splits the data payload, and forwards one portion (sequence numbers 0 through 4,999) to one satellite, and the remaining data (sequence numbers 5,000 through 9,999) to the second satellite, when a contact opportunity appears. This feature requires explicit definition in the protocol header, so that the final destination can anticipate and merge data packets coming from different paths.

(iv) *(Time periods with) constant sending rate.*

DTTP is a rate-based protocol. It employs a constant sending rate in order to exploit the available bandwidth at its full capacity. During scheduled (or even opportunistic) contacts, the protocol steadily fills the pipe. A primary design goal of DTTP is thus to efficiently exploit the valuable space communication resources. As explained next, sending rate can be adjusted according to network conditions.

(v) *Sending rate adaptivity.*

DTTP's sending rate can be accurately characterized as *temporarily constant*. Sending rate adaptation can follow network events such as storage capacity exhaustion. These network events can be either *perceived* by senders through advanced mechanisms or explicitly signaled by receivers; however, in this paper, we do not elaborate on relevant network conditions and how they should affect sending rate.

(vi) *Application-oriented transmission behavior.*

Depending on application type, DTTP can select (or better be instructed to use) a suitable transmission strategy. When no application hint is given, DTTP shows a default transmission behavior. However, we consider that various application types can benefit from customized transmission strategies. The notion of adaptive transport behavior is further explained in Section 3.3, where we present two implementation examples.

### 3.2 Potential Architecture

There is a lot of research effort concentrated on how interoperability between different space agencies can be achieved. Different network protocols, various link layer technologies, static versus dynamic routing, are some of the conflicting approaches regarding space internetworking.

Probably, one of the hot arguments is related with the suitability of IP in space. Indeed, IP extension into space might constitute the common network functionality to interconnect space networks, and integrate them with the terrestrial Internet. However, there exist open issues regarding IP adaptation to space environments. Assuming an IP-everywhere communication infrastructure, DTTP in such a networking framework is shown in Fig. 1.

Recently, a lot of attention has been brought around Delay-Tolerant Network Architecture. DTN's major characteristic is that it can glue together different network protocols. More specifically, DTN acts at the application layer and essentially connects *network regions* that potentially run different protocol stacks. Under this scenario, DTTP's custody transfer functionality is deactivated since it is also offered by the DTN architecture. Figure 2 shows DTTP in a DTN-enabled internetwork.

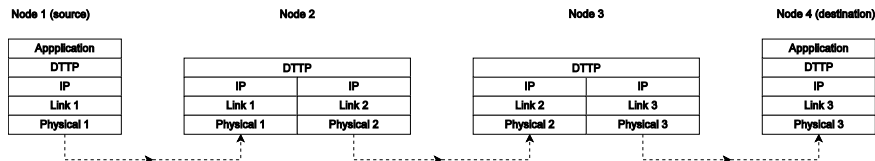


Fig. 1. DTTP deployed in an IP-enabled internetwork.

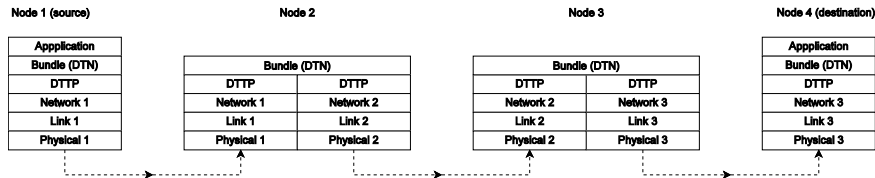


Fig. 2. DTTP deployed in a DTN-enabled internetwork.

### 3.3 Implementation

We have implemented DTTP's primary functionality in Network Simulator ns-2 [14]. Our first implementation of DTTP is based on SACKs. However, we intend to evaluate various acknowledgment mechanisms, such as selective acknowledgments (SACKs), selective negative acknowledgments (SNACKs), delayed acknowledgments, or come up with new sophisticated acknowledgment functions.

In accord with different application scenarios, we present two general tactics to provide reliability. The first targets applications that can benefit from *graduated reliability enhancements*. Suppose, for example, the case where a 30-minute video cap-

tured by a distant satellite is sent back to Earth. Also assume it is desired that each minute is reliably received for instant play-back previously to full-video receipt. An analogous case is a high-resolution photography that can be received in steps, each one providing a better resolution of the photo. Relevant DTTP implementation requires almost immediate use of acknowledgment information as it arrives, and possibly sending *redundant* data (e.g. doubly transmitting the same data segments in order to avoid or minimize data loss due to bit errors of the wireless channel). Note that this approach might waste some portion of network capacity, due to retransmissions and proactive duplicate-transmissions of data segments. A simplified pseudo-code for this transmission approach is the following:

```

until (all application data is acknowledged)
  start transmitting new application data
  if (acknowledgment info arrives)
    send or multiply-send missing data
  end;
end;

```

The second tactic targets bulk-data transfers, which form a very common application type for stressed networks. Indeed, speed of light limit coupled with rare/short communication opportunities restrict other application types (e.g., real-time applications) from functioning properly, especially when propagation delays become extremely long. A suitable algorithm is to: (i) send all application data, from first to last segment; (ii) exploit available acknowledgment information and retransmit missing segments; and (iii) iterate the receive-ACKs and retransmit processes until done. This transmission strategy is described in the following instruction set:

```

send all application data
until (all application data is acknowledged)
  exploit current acknowledgment info
  send or multiply-send missing data
end;

```

Both transmission tactics (and how receipt of packets is affected) are depicted in Figures 3 and 4. Of course, in reality, the number of received packets is orders of magnitude higher. The figures present how gaps of missing packets (at the receiver) are filled.

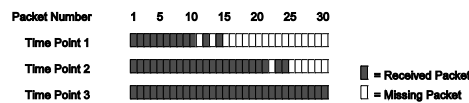


Fig. 3. First transmission tactic.

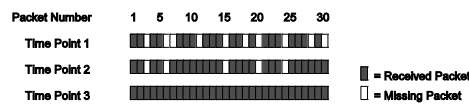


Fig. 4. Second transmission tactic.

## 4 Experimental Evaluation

### 4.1 Evaluation Methodology

We have implemented DTTP’s primary capabilities on the Network Simulator [14]. In particular, our DTTP implementation supports reliability (based on the second transmission tactic described in Section 3.3, and deploying SACK functionality), custody transfer, and rate-based transmission up to the line capacity. We have left parallel data transfer capability as a future extension to the protocol. However, given the static nature of space communications, we have also omitted congestion control. Indeed, each communication session in current space missions involves extensive planning and prior scheduling. Should space communications shift to more flexible architectural schemes, congestion control can certainly be incorporated into DTTP.

In this preliminary study, we evaluate DTTP performance over space network paths varying in distance, data rate, number of hops, and error rate. In our simulations, distance extends through to Mars-Earth communication scenarios. In such distances, data rates are in the order of hundreds of Kbps, such as in ESA’s Mars Express mission and NASA’s Mars Reconnaissance Orbiter. The maximum bandwidth-propagation delay product (which reflects the size of the communication pipe) occurs at shorter distances where high data-rates dominate that product [15]. At this stage, a *comparative* experimental evaluation is not feasible. On the one hand, TCP is not a competitive candidate for comparison due to its inability to function over very long distances with intermittent connectivity, and on the other hand, we do not have sufficiently detailed description to produce reliable simulation code for protocols such as LTP-T.

Table 1 shows the configuration of simulation parameters: combinations of Round Trip Times (RTTs) and data rates considered in our simulation tests; number of hops; and packet error rates. Error models that more precisely capture space network conditions were not explored in our study, and are left for future work. Indeed, wireless channels typically exhibit burst error characteristics. Relevant to distances covered in our tests, space communication channels demonstrate bit error rates of  $10^{-5}$  through  $10^{-8}$ . We just apply a uniform packet error rate that spans up to 10%, and examine the effects on DTTP performance.

**Table 1.** Simulation parameters.

Parameter	Value
{Round Trip Time (sec), Data Rate (Mbps)}	{500, 10} {1200, 2} {2500, 0.5}
Number of Hops	2, 5
Packet Error Rate (%)	0, 1, 5, 10

In our tests, we use DTTP on top of IP, and packets are 1000Bytes long. In every simulation there is one DTTP source (the original sender) that wants to transfer a 10MByte file (i.e., there is no contention for channel capacity). Two topologies are used: a 2-hop and a 5-hop topology, as depicted in Fig. 5. For each direction (downlink or uplink), the one-way propagation delay is uniformly distributed among



all links comprising the path. The packet error rate is equal for both the forward and return path (that carries the application data and the acknowledgments respectively). Along with applying the afore-mentioned error-rate, last link in the forward path exhibits intermittent connectivity: it constantly alternates between *on/off states* (i.e., connected/disconnected) at a fine-grained granularity relevant to each simulation’s duration. On average, that last link is available for 70% of the simulation time, and unavailable for the remaining 30% of time. Also, for this preliminary set of tests, buffer capacity of each DTTP entity is set to relatively high levels (a few MBytes suffice for our tests) so that no data or ACK packet is lost due to a buffer being full. Instead, packet losses are caused only by link errors introduced in the simulations. However, our intention in this paper is to validate DTTP’s properties and provide some initial results for a file transfer scenario.

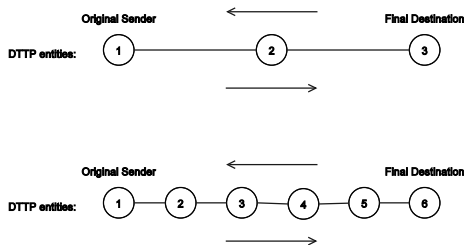


Fig. 5. 2-hop and 5-hop topologies.

## 4.2 Results and Discussion

Graphs in Fig. 6 present completion times for a 10MByte file transfer, both in 2-hop and 5-hop topologies. Each of the graphs refers to a combination of RTT and channel capacity values, as laid out in Table 1.

First noticeable observation is DTTP’s ability to grab available bandwidth. Indeed, in all cases DTTP completes the file transfer in about twice the value of one-way propagation delay. An interesting (and expected) result is that DTTP performs better when the number of hops increases. This is a rational outcome that is justified by the custody transfer capability of DTTP. More specifically, as reliability task is delegated from one node to the next one, shorter SACK-feedback receipt times (now pertaining to distinct links and not to end-to-end paths) facilitate data communication and bring shorter application completion times. Closer look at the graphs in Fig. 6 reveals that as error rate increases, then the greater the number of hops the better DTTP performs. This is depicted by the deviation angle of 2-hop and 5-hop lines in Fig. 6.

We also remark that varying degrees of error rate does only slightly affect the file delivery completion time, when a DTTP session is given full provision of the communication channel. This is explained by the fact that a single DTTP application runs on all links in our simulations. Therefore, DTTP exploits bandwidth resources at the maximum extent: if all application data has been sent at least once, then the (original or intermediate) DTTP sender *fills the pipe* by multiply transmitting missing segments. Of course, this is reflected in the percentage of retransmitted packets (not presented here due to space limitations). The trade-off between file delivery completion

time and percentage of retransmitted packets affects the end-system and overall network performance, and is to be investigated in a future study of ours. We refer the reader to [16] for an automatic retransmission technique that induces early packet retransmissions in order to cope with high bit error rates.

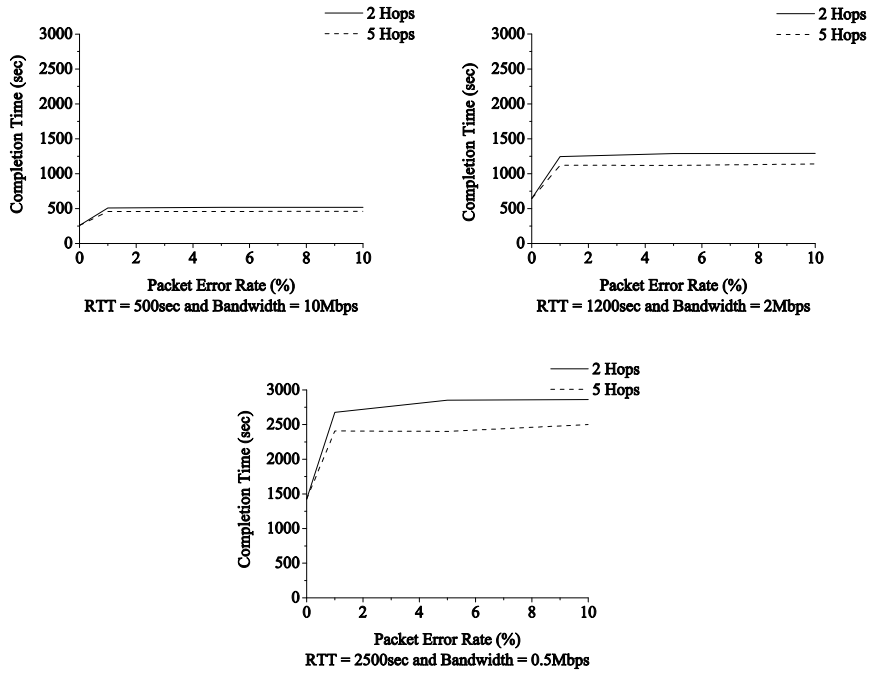


Fig. 6. File delivery completion time using different communication pipes.

The impact of propagation delay on DTTP throughput is shown in Fig. 7. Again, we observe that the greater the number of hops, the better the DTTP throughput achieved. In particular, when the (average) packet error rate increases, DTTP benefits more drastically from greater number of network hops.

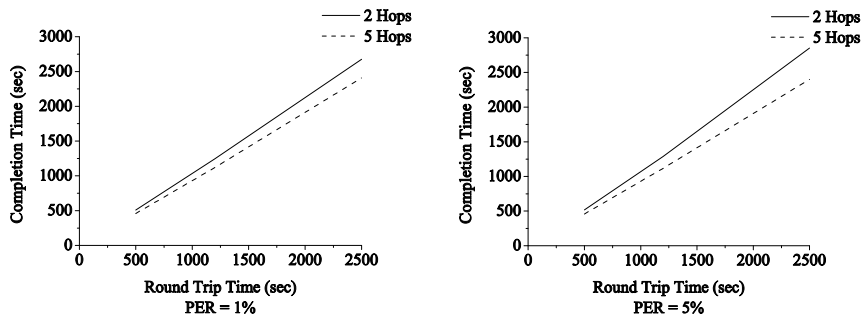


Fig. 7. Round-trip-time impact on file delivery completion time.

## 5 Conclusions

Current and planned activities in space allow for communications strategies that rely on collaborative and multihop end-to-end paths. In this context, a reliable transport service is clearly an important missing component of the communication architecture. We proposed and evaluated DTTP as a potential candidate to fill this gap. We discussed how DTTP satisfies the functionality requirements of both DTN- and IP-oriented space communications. We also demonstrated experimentally that DTTP exhibits satisfying performance, which is enhanced as number of hops increases. In essence, DTTP forms a reliable, transport solution to support flexible multi-node space communications.

As a future task, we are planning to refine and add to DTTP functionality, and validate its properties further. More specifically, we intend to: (i) explore efficient acknowledgment schemes in order to cope with extreme bandwidth asymmetries; (ii) improve its (re-)transmission behavior taking into account the limited bandwidth resources and possibly in relation to delay-bandwidth product; and (iii) supplement DTTP with new capabilities, such as data transfer via parallel paths and explicit signaling for storage resources exhaustion.

## References

1. The Consultative Committee for Space Data Systems (CCSDS), The Official Web Site, <http://www.ccsds.org/>.
2. Ian F. Akyildiz, Ozgur B. Akan, Chao Chen, Jian Fang, and Weilian Su, "InterPlaNetary Internet: state-of-the-art and research challenges," *Computer Networks*, Vol. 43, Issue 2, pp. 75-112, October 2003.
3. V. Cerf et al., "Delay-Tolerant Networking Architecture," RFC 4838, Internet Engineering Task Force, April 2007.
4. K. Scott, S. Burleigh, "Bundle Protocol Specification," RFC 5050, Internet Engineering Task Force, November 2007.
5. K. Hogue, E. Crisculo and R. Parise, "Using standard Internet Protocols and applications in space," *Computer Networks*, vol. 47 no. 5, pp. 603-650, April 2005.
6. Operating Missions as Nodes on the Internet (OMNI), <http://ipinspace.gsfc.nasa.gov/>.
7. C. Partridge and T. Shepard, "TCP Performance Over Satellite Links," *IEEE Network*, vol. 11 no. 5, September/October 1997.
8. Lloyd Wood, Cathryn Peoples, Gerard Parr, Bryan Scotney and Adrian Moore, "TCP's protocol radius: the distance where timers prevent communication," *Third International Workshop on Satellite and Space Communications (IWSSC '07)*, September 2007.
9. I.F. Akyildiz, O.B. Akan and J. Fang, "TP-Planet: A Reliable Transport Protocol for Inter-PlaNetary Internet," *Journal on Selected Areas in Communications*, Vol. 22, Issue 2, pp. 348-361, February 2004.
10. R. C. Durst, G. J. Miller, and E. J. Travis, "TCP extensions for space communications," in *Proceedings of MOBICOM '96*, November 1996.
11. Consultative Committee for Space Data Systems, "Space Communications Protocol Specification-Transport Protocol (SCPS-TP)," *Recommendation for Space Data System Standards, CCSDS 714.0-B-2. Blue Book. Issue 2*. Washington, D.C.: CCSDS, October 2006.

12. Consultative Committee for Space Data Systems, "CCSDS File Delivery Protocol (CFDP)," Recommendation for Space Data System Standards, CCSDS 727.0-B-4. Blue Book. Issue 4. Washington, D.C.: CCSDS, January 2007.
13. Stephen Farrell, Vinny Cahill, "LTP-T: A Generic Delay Tolerant Transport Protocol," Technical Report TCD-CS-2005-69, Trinity College, Dublin, 2005.
14. The Network Simulator - ns-2, <http://www.isi.edu/nsnam/ns/>.
15. Jay L. Gao, John S. SeGuí, "Performance Evaluation of the CCSDS File Delivery Protocol – Latency and Storage Requirement," IEEE Aerospace Conference, March 2005.
16. Ioannis Psaras, Giorgos Papastergiou, Vassilis Tsaoussidis and Nestor Peccia, "DS-TP: Deep-Space Transport Protocol," IEEE Aerospace Conference, March 2008.

# Improving Sensor Network Robustness with Multi-Channel Convergecast

Thiemo Voigt, Fredrik Österlind, Adam Dunkels

Swedish Institute of Computer Science, Box 1263, SE-164 29 Kista, Sweden  
{thiemo,fros,adam}@sics.se

**Abstract.** Most of the existing sensor network deployments are convergecast applications that transmit data from multiple sources to one or more sinks. In this paper, we present the design of a self-organizing, collision-free multi-channel convergecast protocol. We present experiments that demonstrate our protocol's energy-efficiency for low duty cycle applications by comparing it to X-MAC. Our experiments also demonstrate that our protocol's ability to switch channels dynamically increases robustness against interference.

## 1 Introduction

The number of sensor network deployments is increasing rapidly. Many of the current deployments gather environmental data and send them to one or a few sinks. This paradigm is often called convergecast and a number of convergecast protocols have been developed [2, 8, 9, 11, 16].

Most sensor networks operate in license-free bands such as 868 MHz or 2.4 GHz. In the 2.4 GHz band, sensor networks need to co-exist with IEEE 802.11 (WLAN), Bluetooth and other networking technologies which can have a serious impact on the IEEE 802.15.4 network performance if the channel allocation is not carefully taken into account [14]. Modern low-power radios such as the IEEE 802.15.4-compliant CC2420 offer multiple channels which makes it possible to switch channels in order to avoid interference. So far, however, there are only a few attempts to make use of multiple channels. One of the major reasons for this is that algorithms and techniques developed for general wireless networks are not appropriate for wireless sensor networks. Many protocols are designed for more powerful radio hardware and require frequency hopping spread spectrum wireless cards [15]. Another frequent assumption made by existing general approaches is that the hardware is capable of sensing on multiple channels at the same time. However, radios typically found on sensor boards are single radio transceivers that cannot simultaneously transmit and receive. On the other hand, they can operate on different channels at different times.

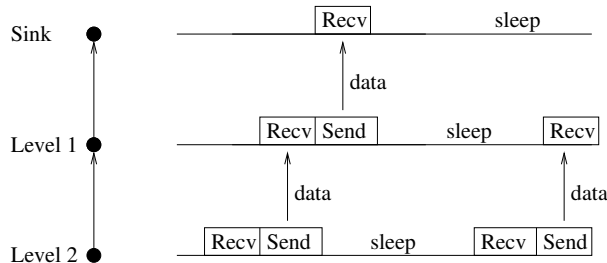
We present an energy-efficient convergecast protocol that improves robustness by using multiple channels available on modern low-power radios such as the CC2420. To the best of our knowledge, this is the first convergecast protocol that uses multiple channels for this purpose. Our protocol builds on the

notion staggered slots introduced by DMAC [11]. We deploy distributed slot assignment to achieve a collision free convergecast protocol without network-wide time synchronization, as well as channel-switching in case a currently used channel is interfered. We use acknowledgements for slot assignment and inter-node synchronized wake-up scheduling. WiseMAC has exploited a similar idea by including the sampling schedule offset into acknowledgements [7]. In addition, by adding control information in the acknowledgement, we provide synchronized channel switching to increase robustness against interference. We present experiments on real hardware that demonstrate the energy-efficiency of our protocol. Simulations with the COOJA simulator [12] as well as experiments on real hardware demonstrate that our scheme increases robustness against interference. The protocol’s ability to synchronize node wake-up without explicit time synchronization is especially useful for low duty cycle data collection applications that demand long lifetime.

The rest of paper is outlined as follows: In the next section we present the design of our convergecast protocol. Section 3 discusses the implementation of our multi-channel algorithm in more detail and presents simulation results. Experimental results are shown in Section 4. Before concluding, we discuss related work in Section 5.

## 2 Convergecast Protocol Design

Our convergecast protocol uses the notion of staggered slots introduced by DMAC [11].



**Fig. 1.** Staggered wake-up a la DMAC

Figure 1 presents the staggered wake-up scheme employed by DMAC. As shown in the figure, different levels of the tree send at different times in order to reduce delay and contention. However, in DMAC there is still contention between nodes on the same level.

Figure 2 shows our basic idea. The receiver of a message sends an ACK that besides acknowledging a packet also states in how many seconds it will turn on its radio again and is ready to receive packets from its children. This way, no

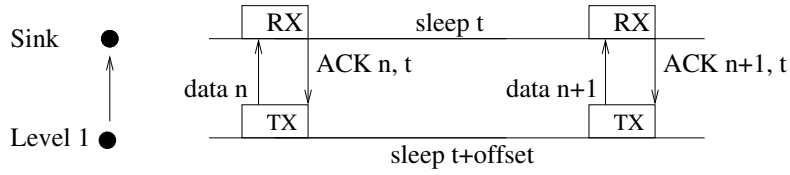


Fig. 2. Basic synchronization scheme

explicit time synchronization needs to be performed since only relative time is of importance. Note that nodes do not need to have their radio turned on during the whole duration of the TX slots.

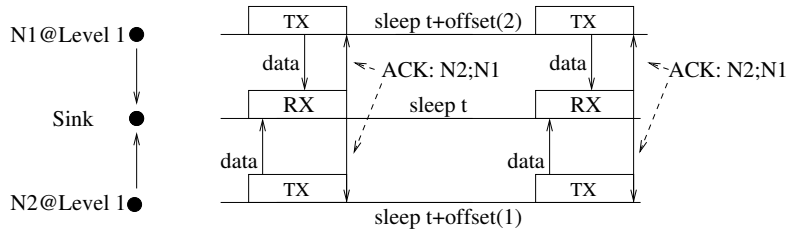
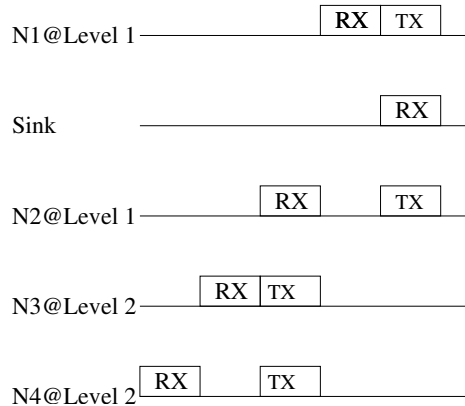


Fig. 3. On-demand slot assignment

By adding offsets for different nodes into the ACK packet, we can extend the basic scheme to let a parent such as the sink node assign slots to its children. Figure 3 demonstrates how the sink assigns different offsets to its children. Each acknowledgement packet contains ACK-fields denoting the positive and negative acknowledgement of the children's data packets in always the same order. This way, the position of the ACK-field can be used by the children to compute their offset into the parent's RX slot. The parent's RX slot must be long enough to allow a maximum number of children to transmit. In the scenario in Figure 3, node  $N2$  may send before  $N1$ .

The scheme can be applied recursively to extend it towards a whole tree. However, when extending the scheme to several levels we need to take care to avoid collisions between nodes on different levels. Towards this end, we introduce a maximum number of children per node. Based on the maximum and its position in the tree, a node can compute its wake-up time and hence offsets for its children. This way, we build a collision-free tree without explicit time synchronization. An example is shown in Figure 4. In this figure,  $N2$  is the parent of  $N3$  and  $N4$ . However, the send and receive offsets may no longer be aligned, i.e. a node's RX slot is not immediately followed by its TX slot. Hence, a node needs to turn its



**Fig. 4.** Slot assignment for larger tree

radio on and off more often which implies a little energy overhead that should be well compensated for by avoiding collisions.

## 2.1 Multiple Channels

Current standards for low power networking allow the usage of multiple channels. For example, in IEEE 802.15.4, there are 16 channels in the 2.4 GHz band. However, only a limited number of protocols leverage the possibility of channel switching in case a channel becomes unusable due to interference. We add channel switching in our design by using the same idea of adding additional control information into the acknowledgement. In our current design, we add information about the next two channels to use into each acknowledgement. If a node does not receive any data message from its children, it blacklists the corresponding channel <sup>1</sup>. More details can be found in the next section.

## 3 Simulating Channel Switching

We have implemented a channel switching algorithm that switches between two channels. In each acknowledgement, a node announces to its children on which channels the next two ACKs will be sent. For example, the first ACK in Figure 5 announces that the sink will send the next ACK on Channel 6 and the successive one on Channel 1. In order to receive the ACK on the correct channel, the level 1 node always performs channel switching after having sent its payload packet.

Figure 5 is produced from the log of a simulation in the COOJA [12] simulator. COOJA supports radio traffic on multiple channels and it is possible to add disturber nodes that interfere with transmissions on a certain channel. The acknowledgements sent by the level 1 node are not shown.

<sup>1</sup> We might also choose a new, not blacklisted channel.



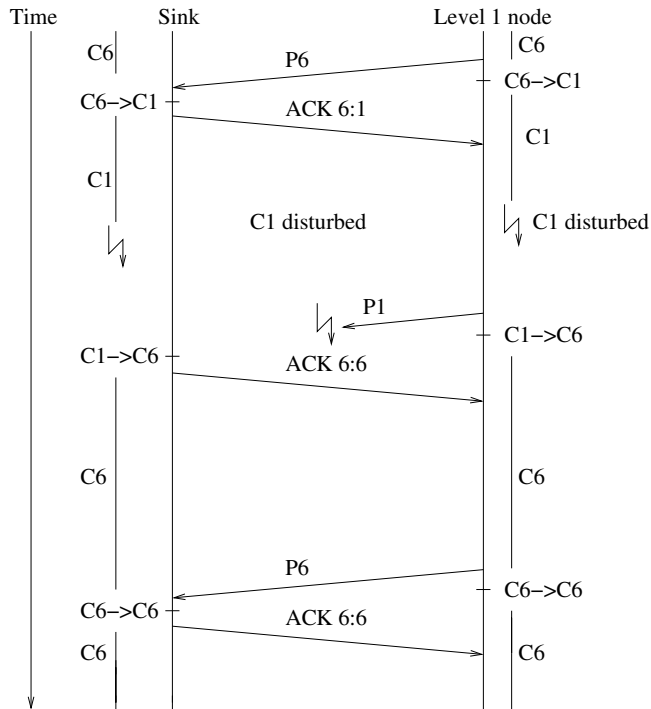


Fig. 5. Channel switching during interference

The figure shows that when the sink notices that it does not receive a message on a certain channel, it announces that it will send future acknowledgements on the channel that is not interfered. This way it is also indicated on which channel the sender should transmit. Note that in the figure only the acknowledgments' fields for channel usage are shown. The acknowledgements also advertise received packets. For example, the second acknowledgement also advertises that the sink has not received the previous packet.

## 4 Implementation and Results

In this section we present results from experiments with real hardware using the Tmote Sky platform. We have implemented the proposed scheme in the Contiki operating system [3] above the broadcast layer of the Rime protocol stack [4], i.e. we turn the radio on and off and change channels at the application layer. The scheme could also be implemented in the MAC layer below the Rime stack. Our current implementation is not optimized in that it does not try to minimize the guard times of the RX slots.

#### 4.1 Energy-efficiency of the proposed approach

We estimate the energy consumption of four nodes deployed in a chain. With our self-organizing approach, we simply set the maximum number of children per node to one to achieve a four-level network setup. The energy consumption is estimated using Contiki’s software-based on-line energy estimation method [5]. We concentrate on the energy for radio listening as radio listening is the dominating factor for power consumption in WSNs [5]. We compare our protocol to X-MAC, a power-saving MAC protocol that is designed to run on top of the 802.15.4 physical layer [1]. X-MAC reduces the power consumption by switching the radio on and off at regular intervals. To send a packet, a node broadcasts a train of short strobe packets. The strobe packet train is long enough to allow all nearby devices to be switched on at least once. When receiving a unicast strobe, a receiver immediately sends a short acknowledgment packet allowing the sender to send the full packet.

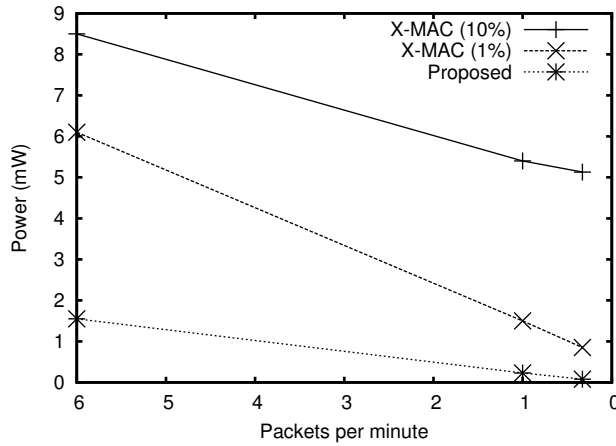
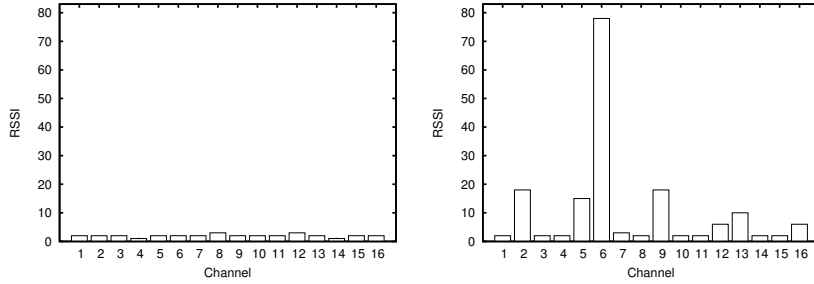


Fig. 6. Comparison of average radio listen power

Figure 6 shows the average power consumption for radio listening comparing X-MAC with the proposed protocol. Since our protocol has a constant radio listening time for each packet, the radio listening time decreases approximately linearly when less packets are sent. In all scenarios, the proposed protocol performs better than X-MAC.

Figure 6 also shows that with the same duty cycle, X-MAC consumes less energy when there is less traffic, i.e. a duty 10% duty cycle will not per se extend the lifetime of the network with a factor of 10 but that the lifetime extension depends on the traffic volume. The reason for this is that after the intended receiver of a packet has indicated that it is ready to receive a packet, the receiver must have its radio turned on until it has received the packet. This



**Fig. 7.** RSSI levels without an interfering node (left) and RSSI levels with a nearby node interfering on channel 6 (right).

task consumes much more energy than listening for the short X-MAC strobe packets.

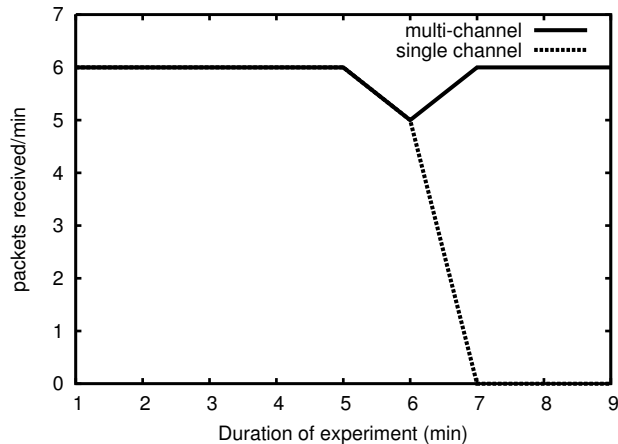
The results indicate that our proposed protocol is suitable for data collection applications with very low duty cycles, for example applications collecting temperature values in buildings.

## 4.2 Channel Switching

**Channel Switching Overhead** Using a microcontroller hardware timer, we measure the duration of our channel switching radio driver function call. The function waits until any pending transmission is finished, and then commands the radio chip to change operating frequency. Finally, the function activates the new frequency by resetting the radio to receive mode. Without any pending transmissions, our results show that the duration is approximately 131  $\mu$ seconds. These experiments demonstrate that the additional delay introduced by switching radio channels is not significant.

**Measuring Channel Quality** We have implemented a small procedure to measure channel quality based on the received signal strength indicator (RSSI). Figure 7 shows the RSSI levels of the 16 channels with and without a disturber node on channel 6. The increase of the RSSI on channel 6 when the disturber is turned on is clearly visible.

**Increased Robustness with Multiple Channels** In the next experiment, we use a sink node, a sender and a disturber node. The latter is programmed to cause interference by sending packets back-to-back on a predefined channel. The sink sends acknowledgements every 10 seconds. Therefore, the sender transmits 6 packets per minute. The algorithm is the same as described in Section 3. Five minutes and 25 seconds after the beginning of the experiment, the disturber interferes with the packet transmission on channel 6.



**Fig. 8.** Using multiple channels, the communication between sender and sink can be sustained despite interference on one of the channels.

Figure 8 shows that an application that uses a single channel only is not robust against channel interference. As expected, when multiple channels are used, our channel switching algorithm takes the loss of one packet as an indication to move future communication from the corresponding channel. Hence, after one packet is lost, all packets arrive reliably at the sink again. Our scheme expects that retransmissions of lost packets are performed during the node’s next TX slot. For example, the automatic retransmissions in 802.15.4 cannot be used, since our acknowledgements are not unicast packets.

We have not yet integrated the channel quality measurement procedure into our protocol. This would allow us to proactively stop using a channel when it is interfered instead of using packet loss as the indication for interference. Since the quality measurement procedure is very fast, its energy consumption is almost negligible.

## 5 Related Work

One of the reasons for reduced robustness and reliability in sensor networks are temporal disturbances/uncertainties in the radio medium. In 2003, experiments by Zhao et al. have demonstrated the existence of temporal disturbances, i.e. they have shown that packet reception rates of sensor nodes vary significantly over time even in quite static environments [17]. Petrova et al. have measured that different 802.11 channels interfere with a set of different 802.15.4 channels [13].

While modern low-power radios such as the IEEE 802.15.4-compliant CC2420 are available, so far there are only a few attempts to use the available channels. Many of these have leveraged multiple channels to increase throughput and

increase performance. Zhou et al. have focused in their simulation results on throughput, energy efficiency and channel access delay [18]. Durmaz Incel et al. have shown that a multi-channel version of LMAC increases performance proportional to the number of available frequencies compared to the single-channel version [6]. Liang et al. have reduced the dissemination time of large objects in wireless sensor networks by utilizing multiple channels [10]. In contrast to these efforts, we use multiple channels to increase robustness and reliability by switching to a different channel if interference makes it impossible to use the selected channels.

One of the most energy-efficient convergecast protocols is Dozer [2]. In contrast to our approach, Dozer does not build a collision-free delivery tree. The same is true for Twinkle [9], DMAC [11] and the approach proposed by Gandham et al. [8]. The latter tries to reduce latency by minimizing the number of required timeslots. Zhang et al. focus on bursty convergecast where large bursts of packets are transmitted [16]. None of these protocols uses multiple channels.

## 6 Conclusions

In this paper, we have presented a convergecast protocol that dynamically builds a collision-free tree based on information in the acknowledgements. The protocol also performs channel switching to reduce interference problems. Our simulations and experiments with real hardware have demonstrated the effectiveness of the proposed approach.

## Acknowledgments

This work was funded by the Swedish Energy Authority.

## References

1. M. Buettner, G. Yee, E. Anderson, and R. Han. X-mac: a short preamble mac protocol for duty-cycled wireless sensor networks. In *ACM SenSys*, Boulder, USA, November 2006.
2. N. Burri and R. Wattenhofer. Dozer: ultra-low power data gathering in sensor networks. In *6th international conference on Information processing in sensor networks (IPSN 2007)*, Cambridge, USA, April 2007.
3. A. Dunkels, B. Grönvall, and T. Voigt. Contiki - a lightweight and flexible operating system for tiny networked sensors. In *Proceedings of the First IEEE Workshop on Embedded Networked Sensors*, Tampa, Florida, USA, November 2004.
4. A. Dunkels, F. Österlind, and Z. He. An adaptive communication architecture for wireless sensor networks. In *Proceedings of the Fifth ACM Conference on Networked Embedded Sensor Systems (SenSys 2007)*, Sydney, Australia, November 2007.
5. A. Dunkels, F. Österlind, N. Tsiftes, and Z. He. Software-based on-line energy estimation for sensor nodes. In *4th workshop on Embedded networked sensors (EmNets'07)*, pages 28–32, 2007.

6. O. Durmaz Incel, S. Dulman, and P. Jansen. Multi-channel Support for Dense Wireless Sensor Networking. In *European Conference on Smart Sensing and Context (EuroSCC)*, October 2006.
7. Amre El-Hoiydi, Jean-Dominique Decotignie, Christian C. Enz, and E. Le Roux. wisemac, an ultra low power mac protocol for the wisenet wireless sensor network. In *ACM Conference on Networked Embedded Sensor Systems (SenSys 2003)*, pages 302–303, 2003.
8. S. Gandham, Y. Zhang, and Q. Huang. Distributed Minimal Time Convergecast Scheduling in Wireless Sensor Networks. *Proceedings of the 26th IEEE International Conference on Distributed Computing Systems (ICDCS)*, July 2006.
9. B. Hohlt and E. Brewer. Network Power Scheduling for TinyOS Applications. *IEEE Int. Conference on Distributed Computing in Sensor Systems (DCOSS)*, June 2006.
10. C.J.M. Liang, R. Musaloiu-E, and A. Terzis. Typhoon: A Reliable Data Dissemination Protocol for Wireless Sensor Networks. In *Proceedings of the Fifth European Conference on Wireless Sensor Networks (EWSN2008)*, Bologna, Italy, January 2008.
11. G. Lu, B. Krishnamachari, and C. S. Raghavendra. An adaptive energy-efficient and low-latency mac for data gathering in wireless sensor networks. In *International Parallel and Distributed Processing Symposium (IPDPS)*, April 2004.
12. F. Österlind, A. Dunkels, J. Eriksson, N. Finne, and T. Voigt. Cross-level sensor network simulation with cooja. In *Proceedings of Proceedings of the First IEEE International Workshop on Practical Issues in Building Sensor Network Applications (SenseApp 2006)*, Tampa, Florida, USA, 2006.
13. M. Petrova, J. Riihijarvi, P. Mahonen, and S. Labella. Performance study of IEEE 802.15. 4 using measurements and simulations. In *Proceedings of IEEE WCNC*, April 2006.
14. S.Y. Shin, S. Choi, H.S. Park, and W.H. Kwon. Packet Error Rate Analysis of IEEE 802.15. 4 under IEEE 802.11 b Interference. In *International Conference on Wired/Wireless Internet Communications (WWIC 2005)*, May 2005.
15. A. Tyamaloukas and J. J. Garcia-Luna-Aceves. Channel-hopping multiple access. In *IEEE International Conference on Communications (ICC)*, June 2000.
16. H. Zhang, A. Arora, Y. Choi, and M.G. Gouda. Reliable bursty convergecast in wireless sensor networks. *Computer Communications*, 30(13):2560–2576, 2007.
17. J. Zhao and R. Govindan. Understanding packet delivery performance in dense wireless sensor networks. In *The First ACM Conference on Embedded Networked Sensor Systems (SenSys 2003)*, Los Angeles, California, November 2003.
18. G. Zhou, C. Huang, T. Yan, T. He, J.A. Stankovic, and T.F. Abdelzaher. MMSN: Multi-Frequency Media Access Control for Wireless Sensor Networks. In *Proceedings of IEEE Infocom*, Barcelona, Spain, April 2006.

# Evaluation of Routing Overlay Solution for a Hybrid Mobile ad hoc Network

Juhani Latvakoski<sup>1</sup>, Tommi Aapaoja<sup>1</sup>, and Juha Kärnä<sup>1</sup>

<sup>1</sup> VTT Technical Research Centre of Finland Kaitoväylä 1, P.O.Box 1100. FIN-90571 Oulu, Finland

{Juhani.Latvakoski, Tommi.Aapaoja, Juha.Kärnä}@vtt.fi

**Abstract.** In this paper, evaluation of the routing overlay solution for Hybrid Mobile ad hoc Networks is provided. The motivation for the routing overlay solution arises from interoperability, network infrastructure cost and complexity problems related to integration of static Internet networks and mobile ad hoc networks with different types of devices. The evaluation has been carried out by comparing the overlay routing with the traditional ad hoc routing, and measuring the performance and overhead in laboratory test system. The measurements indicate that the overlay causes some more delay and overhead, however, it works quite well from the functional point of view when limitations are taken into concern.

**Keywords:** ad hoc mobile networks, mobility, hybrid networks, routing, overlay

## 1 Introduction

Today, the commercial wireless networks are usually quite static in nature, and only the last or first hop to the end user system is wireless. The ad hoc networks are different in the sense that wireless media is also applied between the devices, which establish the dynamic network. This means that communication between the devices, where a direct radio link does not exist, is supported over some other intermediate device(s) by means of the multihopping function. The hybrid mobile ad hoc network, discussed in this paper, is an ad hoc network which may be mobile and may be temporarily connected with static network such as Internet, and it can therefore be called as a hybrid mobile ad hoc network. The routing overlay refers to a virtual network created on top of the existing physical network. In such a routing overlay network the virtual links create the network. A link is logically a direct connection between the nodes, however, there still may exist several physical routing hops in the physical underlying network.

There are multiple of ad hoc routing protocols available. Each of these protocols, e.g. Topology Dissemination Based on Reverse-Path Forwarding (TBRPF), Ad hoc On-Demand Distance Vector (AODV), and Dynamic MANET On-demand (DYMO) has some benefits and drawbacks, however, it can be claimed that none of them have

reached de facto status in market. Or at least, none of the protocols is applicable for all of the use environments due to different delay requirements, reaction times for route changes, the power capabilities of the routing devices, and the limitations of the bandwidth usage, quality of service level and security. Therefore, it can be assumed that either multiple ad hoc routing solutions should be allowed, or at least the solution should be modular enough to enable smooth configuration. When multiple ad hoc routing solutions are applied, the *interoperability* will become one of the most critical requirements. Another challenge is related to the network infrastructure, and especially the cost of investments required when building it. When an ad hoc routing protocol is taken into use, it usually requires changes in the network infrastructure part of the existing system to allow interoperability with legacy systems. One possible way of overcoming the required infrastructure cost is the application of overlay routing. This limits the investments to overlay routers, the number of which is smaller than infrastructure routers. The mirror side of the same problem is visible in the terminal side; the commercial mobile terminals may support the novel developed features, such as mobile IP [2], ad hoc routing and NEMO extensions [3] for the mobile IP in future, but their proper functioning requires network infrastructure support. When mobility is concerned, the NEMO enhancements to Mobile IP provide one potential solution. However, the availability of a home agent for all the devices in the ad hoc network may be a problem. In addition, the overhead caused by bi-directional tunneling in the NEMO solution may be a problem, which is more serious in nested mobility cases. In addition, the complexity of the system is huge due to the heterogeneity of the configurations. It is still not known whether the currently ongoing specification work at IETF for Mobility EXTensions for IPv6 (MEXT) improve the situation. Based on the described *interoperability, network infrastructure cost and complexity reasons*, we have developed the overlay routing approach for the mobile ad hoc networking, the approach of which is initially described in our previous publication [1].

In the overlay routing approach [1], the system consists of different types of nodes, which can be normal IP network node, open network infrastructure (ONI, e.g. Linux) or closed network infrastructure (CLI, e.g. Symbian, Microsoft) nodes. The ONI nodes allow user changes to the network infrastructure part of the software, and, basically, the CLI does not allow it. The approach relies on the solutions such as *Routing Overlay Bridge, Overlay Router*, and a novel overlay routing control protocol (*ORCP*). The Routing Overlay Bridge routes packets destined for the CLI devices to the overlay network and overlay level packets coming from the CLI devices to the physical routing (IP) level. In addition, the Routing Overlay Bridge manages the pipes between the ONI device and the CLI device. The CLI device contains an overlay router, whose task is to manage the overlay routing in general. The overlay routing includes solutions for network mobility and ad hoc routing in the overlay level. First, the IP packets are routed between the routing overlay and the network layer. Then the routing and management of pipes towards the other nodes is carried out based on the overlay routing tables. Finally, an application programming interface (API) is provided for user applications.

The contribution of this paper is evaluation of the overlay routing approach. The evaluation has been carried out by comparing the overlay routing with the traditional



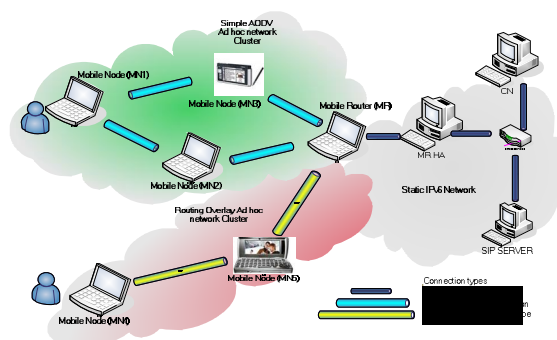
ad hoc routing [6], and measuring the performance and overhead in laboratory test system.

Rest of this paper is organized as follows. Chapter 2 describes the developed routing overlay and AODV based ad hoc routing solutions in the context of experimental system. Chapter 3 describes the measurements carried out in the experimental system and discusses about the key achieved results. Finally, conclusions are provided in chapter 4.

## 2 Ad hoc Routing Solution

### 2.1 Experimental System Overview

The experimental test platform for ad hoc routing solutions is visualized in Figure 1. The mobile network consists of two mobile clusters: AODV Ad hoc Network and Routing Overlay Ad hoc Network. Both ad hoc network clusters consist of a mobile router (MR) and set of mobile nodes (MN). The difference between the clusters is that the ad hoc routing in the overlay cluster applies the routing overlay approach [1, 5], and the routing in the AODV cluster is executed using simple AODV<sup>1</sup>. The other issue is that different kind of devices have been applied: the multihop node in the AODV cluster is a Laptop Linux (ONI) device (MN2) or Internet Tablet Linux (ONI) device (MN3), and the multihop node in the routing overlay cluster is Symbian (CLI) device (MN5). The role of MR is to connect both clusters into the static IPv6 network, which contain of home agents (HAs) for MNs and MR, correspondent node (CN) and SIP server. In the test platform, the mobility of the network represented by the MR has been implemented using the NEMO protocol, which actually is extension to Mobile IPv6 technology. It is transparent to the provided ad hoc routing solutions.



<sup>1</sup> The simplified version of the AODV protocol, which is designed and implemented by VTT.

Figure 1. Experimental test platform.

The physical realization of the mobile network connections has been implemented using 11/2 MBit WLAN cards, the laptops and desktop machines execute Linux, and the symbian device is Nokia 9500 communicator with Symbian operating system version 7.0. The operating system has a hybrid IPv4/IPv6 stack, however, it does not contain the Mobile IP functionality, and the API provided to the programmer does not allow modifications or additions to the IP stack. This means, that it doesn't allow the user to modify the routing tables. Therefore, the Symbian device is a CLI device, and all the needed functionality must be provided at the application level, as an application level routing overlay network. The *application layer* refers to the layer, which uses purely the APIs provided by the operating system. The Internet tablet is Linux based Nokia 770, which is an ONI device and it is possible to modify the physical routing capabilities of it.

## 2.2 Routing overlay Solution

The routing overlay solution is based on the pipes, and a novel overlay routing control protocol, called here as *ORCP*, has been developed for it. Thus the overlay works using pipes between nodes, **Figure 2** [1]. There are several different pipes in the ORCP: the pipes between the Symbian mobile nodes, pipes between Symbian and Linux mobile nodes, and the NEMO specified bi-directional pipe between the mobile router and the home agent. The NEMO bi-directional pipe works purely on the IP layer, so it is transparent from the routing overlay perspective. There is no pipe between the Linux mobile node and the mobile router, communication between them works purely on the IP layer. The pipes between Symbian mobile node and Linux node are different from the pipes between two Symbian mobile nodes. That's because there is the *IPBridge* software between on the Linux mobile node that enables the creation and usage of the pipe.

The pipes are designed to be *data transparent*, which means that any data can be transmitted over them. The routing overlay applies TCP/IP for transport, because of the need for reliable transmission of data, even if it adds some overhead to the communication. The routing overlay protocol packets contain starting and end markers, as well as the length of the message and the actual data payload. A special poll message has been created to see if the pipe is still alive. The poll message is an *empty* pipe message in the ORCP with *pipe message length* of zero.

A single pipe is identified by a link-local address of the node, the mobile prefix, and its global and site-local MANET addresses. When data is to be sent to the global address of a node, it is actually sent to the link-local address. The real network layer connection exists on this IP address. There is need for overlay nodes to inform others about their address, or for the Linux device to inform the Symbian mobile nodes about its sequence number. This is carried out by using special messages of the ORCP protocol. The message for sequence number exchange contains the sequence number identifier and the sequence number, and the message for address exchange contains the address type identifier, the address length and the address.

Neighbour discovery is a fundamental part of the dynamic discovery of network nodes. For this, multicast messages are employed, using the services of the underlying network directly. The multicast messages sent from the Symbian mobile node are sent to a multicast group, which the mobile nodes have joined. The Linux nodes also send these messages, advertising their IPBridge software. The single sent message is received by all the nodes listening to that address. The multicast group used is the *link-local*, not *global*.

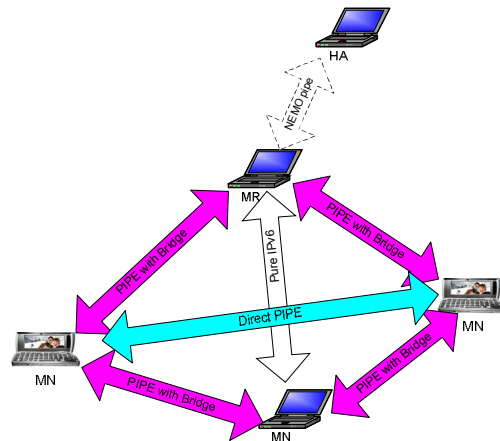


Figure 2. The Routing Overlay Pipes.

The *IPBridge* component is needed to transmit the IP packets from the overlay network level to the IP level in order to function with the normal Internet and nodes not implementing the overlay network functionality. This software works on the Linux platform and has to capture IP packets destined to the Symbian mobile nodes, as well as push the IP packets received from the Symbian pipes to the IP layer of the Linux device. In order to do that, the Netfilter/IPTables architecture has been used. It defines *hooks* in which the user can register and mangle the packets received. So, the software is implemented as two separate modules, one functioning on the *kernel* side with the netfilter hooks and one functioning on the *user-space*. The *IPBridge* also needs to be advertised and this is done with a multicast message just like with the Symbian mobile nodes.

The core of the ORCP protocol has been realized in this research. The solutions described earlier establish the basis for the *Overlay Router and mobility engine*, which implements overlay routing. When an IP packet is to be sent to a destination, the overlay routing table is consulted for the next hop to the destination. This next hop is a reachable neighbor, to which the mobile node has a pipe.

## 2.3 Discussion

The overlay routing is realized using pipes controlled by ORCP protocol. The pipes are created for the transmission of user data payload, and each pipe is realized using TCP/IP. This means naturally that some additional overhead is included for the user data traffic, because of additional TCP/IP headers in the messages. In this kind of an approach, a pipe is a logical link and direct connection between the nodes, however, there still may exist several physical routing hops in the physical underlying network. In this approach, the routing happens between pipes in the overlay network nodes. Ad hoc routing using e.g. AODV is different in the sense that the control protocol, AODV, works in the application level, however, the routes are established over the physical connections from a node to next physically adjacent node. When sending data from source to destination, the route goes through each of these physically adjacent nodes. Thus, each node in the physical route has a routing table related to the route. In the overlay routing approach, the routing tables are only in the overlay routing nodes.

When the system contains CLI nodes, they are very difficult to be taken as a part of the ad hoc network. One possible solution is application of overlay approach to enable connecting them to the ad hoc network. As a part of this research we have realized ORCP protocol, which makes it possible. However, the question is how much the overlay routing degrades the performance and how much overhead is caused. And especially, it is very interesting to know whether the overlay routing is feasible and possible to be applied.

## 3 Measurements

### 3.1 Arrangements

The objective of the measurements has been to execute the same tests with the ORCP based routing overlay and SAODV based ad hoc routing solutions, and then analyse the degradation of performance and additional overhead caused in the overlay approach, and its feasibility in general. The measurements have been carried out in the experimental test platform visualized in *Figure 1*. In the tests, the end node, middle node and mobile terminal are laptops with Pentium 3 700/800 MHz processors, 256 MB ram and executing Linux Fedora core 4. Mobile router is a laptop with Pentium-M 1.8 GHz processor, 512 MB ram and executing Linux Fedora core 3. Nokia 770 is Linux based internet tablet with Texas Instruments ARM 250 MHz processor and 64 MB ram. Nokia 9500 communicator executes Symbian S80 operating system.

In the first phase tests, *registration time, signaling overhead and data payload overhead* are measured and explained. The registration time refers to delay which is caused by initializing the system to be ready for operation. Signaling overhead refers

to the number of required control messages and transmission of their contents. Data payload overhead refers to the required control type of bits included for each of the user data packets. Comparison of route discovery wasn't feasible, because of limited realization of ad hoc routing in the overlay level.

In the second phase, the performance of the solutions is compared using real end-user mobile devices as intermediate nodes. The performance with the SAODV is tested by measuring transfer speed and ping delays with Linux laptops and Nokia 770 internet tablet. The performance of the Overlay is tested using the Nokia 9500 Communicator running Symbian S80 as an intermediate node.

In the tests, Linux laptops and Nokia 9500 are setup in 11 Mbps mode, but Nokia 770 is in 54 Mbps mode. This difference should be seen in throughput tests if 770's processing speed is high enough. Test system has been built in a laboratory room and the system consist of seven different devices in the test platform (*Figure 1*), but only devices required in each test had their WLAN cards enabled. There were also from one to six other WLAN networks in the office environment consuming the available WLAN band, but they were in other channels and their signals were relatively weak. Most of the measurement data were collected with Ethereal tool and Nethawk's WLAN analyzer was used for double-checking results. Throughput and delay tests were made in a passage of the office in a way that end nodes didn't hear each other, but middle node could hear both end nodes.

### 3.2 Measurement Results

The first phase measurement results dealing with registration delays and overhead are shown in *Table 1*. The registration delay with SAODV includes duration of the Hello procedure and route discovery and with overlay the duration of the neighbor discovery and pipe establishment. The route discovery is carried out to establish a route between MN1 and MR. The pipe establishment refers initialization of two pipes: MN4-MN5, MN5-MR enabling a connection between MN4 and MR. The procedures makes the same result, the connections between MN1-MR, and MN4-MR are ready for use. Therefore, the registration delays are estimated to be comparable with each other.

The SAODV hello procedure with laptops takes ca 5 ms, and the route finding ca. 12 ms (+-2 ms). When distance between SAODV nodes increases, increases also route finding time, which was in very short distances (<1 m) about 12 ms and in short distances (~10m) about 30 ms in office environment. With Nokia 770 the SAODV registration time is little longer than with laptops. Hello procedure takes ca. 10 ms and route finding ca. 22 ms.

The total registration time with overlay took 1698 ms to complete. With MR and MN5 registration took about 801 ms. First packet is a neighbor advertisement UDP broadcast and the last packet is an ACK-message of the TCP-connection creation. Registration with MN4 and MN5 took 897 ms. Last TCP-messages were pure ACK-messages, so if actual data transfer would begin immediately after connection

creation, those ACK-messages could have been sent along with data packets about 200 ms earlier.

*Table 1. Measurement results*

	registration time	number of signaling messages	signaling overhead (bytes)	data payload overhead (bytes)
SAODV laptop	5,3 ms (Hello) + 12-+2 ms (route discovery)	9 (hello), 9 (route disc.)	450 B (Hello) + 712 B (route discovery) = 1162 Bytes	40
SAODV Nokia 770	10,2 ms (Hello) + 22 ms (route discovery)	9 (hello), 9 (route disc.)	1162 Bytes	40
ORCP Nokia 9500	1698 ms (801 ms, 897 ms)	19	738 Bytes + 810 Bytes = 1548 Bytes	72, or 144 (if separate acks)

The measurements indicate that SAODV registration takes considerable less time (under 20ms) than overlay (1.6 sec). The reasons for this are estimated to be caused by the delays in the neighbor discovery and pipe establishment.

The number of required signalling messages in the SAODV hello procedure was 9, and another 9 for the route discovery, which means 18 included messages in the SAODV registration. The overlay start-up required 19 messages for MR and MN4 to make neighbour advertisements, neighbour solicitations and creation of TCP connections (pipes) with MN5 (Nokia 9500). The last TCP-messages are pure ACK-messages so if actual data transfer would begin immediately after connection creation, those ACK-messages could have been included into data messages. Thus there are not any big differences in the number of messages.

After the connection has been established, SAODV doesn't create any additional overhead to the user data payload. This means that the normal IP traffic headers are included, which means 40 bytes header for IPv6. The only additional overhead is caused by the hello messages, which are sent every now and then. But they do not add any data to the user data traffic messages. The overhead introduced by the overlay solution is caused by the use of TCP as the transmission protocol. TCP connection establishment is not included in the overhead analysis. The absolute overhead per user data byte have not been calculated since the amount of user data sent in a TCP segment varies. When the IP packet is pushed over the pipe with a TCP PSH+ACK message, the IPv6 header allocates 40 bytes. After that, the following TCP header allocates 32 bytes. So, the minimum overhead for every IP packet is 72 bytes. This is the case when the TCP ACK message [7] can be sent with the data. If the receiving

end does not have any data to send to the other end, it will send a TCP ACK message, which is another 72 bytes. It can be calculated that the same overheads occur every time an IP packet traverses across a pipe. If the packet traverses across two pipes – like in this scenario - the minimum occurring overhead will be 144 bytes.

The second phase measurements dealing with delays and throughput are visualized in **Figure 3** and **Figure 4**. In the measurements, data is sent with ping6 from MR and measurement results are collected with Nethawk network Analyzer and Ethereal executed in the MR. The pings are sent using various speed from MR, and the delay from the request to the response are measured. As shown in **Figure 3**, ping delay for the overlay solution is ca. 30 ms, and for the SAODV solution ca. 10 ms when the pings are sent relatively rare. In the latter case, the delays with the laptops are a little bit smaller than the ones with Nokia 770. When the system become more loaded, then at some point (when throughput is 17 kb/s) the delay with the overlay start to increase. When throughput is ca. 145 kB, the delay with the SAODV and laptop start to increase, while Nokia 770 continues to work as before. This is probably due to the reason that Nokia 770 was configured in 54 Mb mode.

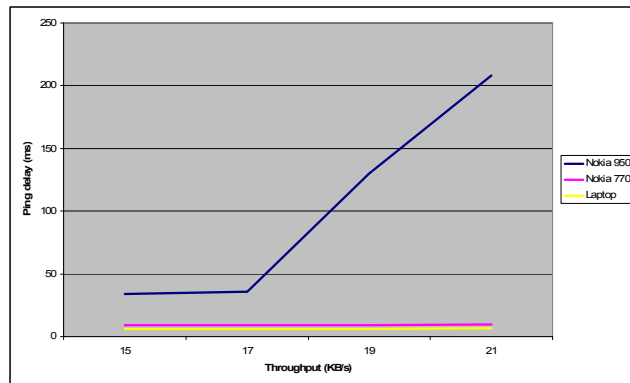


Figure 3. Ping delays

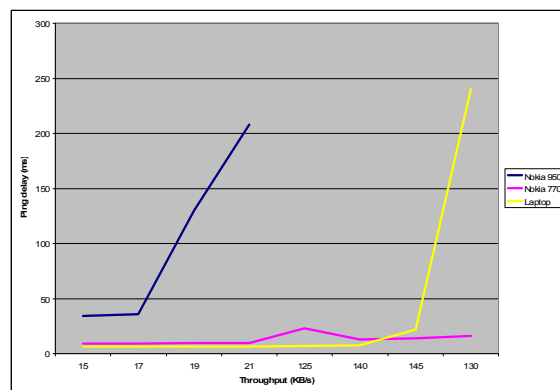


Figure 4. Maximum throughput.

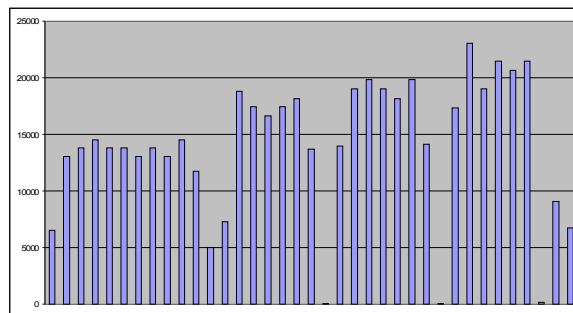
### 3.3 Detailed overlay performance measurements

One of the methods to measure overlays throughput was basic Ping. Pings were run with different intervals and packet sizes. However, it seemed that there were some inaccuracies with low interval times, so the following Ping-measurements are – in that part – only suggestive and not exact.

The overlay's maximum throughput was tested with simple program that sends UDP-packets with Netcat from MN7 to MR. Sending speed (packets/s) is slightly increased after each sent packet to see how much traffic overlay could handle. Traffic was measured with Nethawk WLAN Analyzer and results were calculated from trace files. In that test the overlay was able to reach about 19 kB/s throughput in one direction.

Overlay's maximum throughput was also tested with ping with different intervals and packet sizes. The Nokia 9500 throughput in one direction is shown in **Figure 5**, which describes the results of five tests with different interval and packet size settings. First test were made with 600 bytes packets and 0,05 s intervals and last successful test (fourth) were made with 700 bytes packets and 0,035 s intervals. In fourth test the overlay reached its maximum throughput, about 20kB/s. In fourth test the overlay was a bit unstable and ping varied pretty much, but the overlay was able to make through the test without crashing. In fifth test packet size was the same 700 bytes as in fourth test, but interval was changed to 0.03 s. As picture shows, the overlay crashed just after the start of the test.

After logging was turned off, it was possible to stream video and audio over the overlay network. VoIP sessions were tested and calls were made successfully with KPhone software. KPhone had 3 available codecs to be used: iLBC, G.711 and GSM. The VoIP sessions through the overlay were most successful with iLBC, because of the lowest amount of packets transmitted in a certain time interval. The low packet transmit rate through Nokia 9500 is important, because the Symbian device has limited capabilities and hangs if the packet transfer rate becomes too high.



*Figure 5. Nokia 9500 throughput in one direction.*

It was also possible to stream light video through the overlay with VLC media player. Due to the overlay's limits, only very small video clips could be streamed through the overlay and clips' quality had to be reduced manually with VLC.



### **3.4 Discussion**

The measurements indicate that the application of overlay causes the registration time to be considerable higher than with the traditional ad hoc routing (ca. 20ms vs. 1.6 sec). This is due to the ipv6 neighbor discovery and pipe establishment delays in the overlay case. From SAODV point of view the neighbor advertisement protocol is useless and only mixes SAODV control traffic. In addition the overhead of the overlay is higher (32 bytes, 72 bytes).

The measurements also indicates that the overlay solution and/or Nokia 9500 has some problems with performance, which is seen in the higher delays, and the fact that the delays start to increase when the system is more loaded. Nokia 770 has a bit higher delays than laptop which is caused probably due to limited system resources. One potential source of deviations to the measurements may be caused by the fact that Nokia 770 was configured in 54 Mb mode when executing the tests, while the other equipments were in the 11 Mb mode.

However, the overlay solution based on the ORCP seems to work quite well in the functional point of view. Thus the overlay solution can be applied, but its limitations related to registration time, overhead and performance should be taken into concern

## **4 Conclusions**

In this paper, evaluation of the routing overlay solution for Hybrid Mobile ad hoc Networks is provided. The motivation for the overlay solution arises from interoperability, network infrastructure cost and complexity problems related to integration of static Internet networks and mobile ad hoc networks with different types of devices. The evaluation has been carried out by comparing the overlay routing with the traditional ad hoc routing, and measuring the performance and overhead in laboratory test system.

The measurements indicate that the application of overlay causes the registration time to be considerable higher than with the traditional ad hoc routing (ca. 20ms vs. 1.6 sec). In addition the overhead of the overlay is higher (32 bytes, 72 bytes). Even if the delays and overhead may cause some concerns, the overlay approach seems to work quite well from the functional point of view. In addition, it provides a possible approach for the interoperability, network infrastructure cost and complexity reasons problems. This means that overlay approach is feasible, but its limitations related to registration time, overhead and performance should be taken into concern.

## References

1. Latvakoski, J. A., T. (2005). "Towards a Routing Overlay for a Mobile Ad hoc Network." First International Workshop on Convergence of Heterogeneous Wireless Networks (ConWiN) 10th Jul 2005 Budapest, Hungary: 6p
2. Johnson D., Perkins C., Arkko J. (2004) Mobility Support in IPv6. RFC 3775, 165 p.
3. Devarapalli V., Wakikawa R., Petrescu A., Thubert P. (2004) Network Mobility (NEMO) Basic Support Protocol Internet Draft, NEMO Working Group, 38 p.
4. Andersen D. (2001) "Resilient Overlay Networks." MS\_Thesis, Massachusetts Institute of Technology: 86p.
5. Aapaoja, T. 2005. Routing Overlay for an Ad hoc Network. Diploma Thesis. University of Oulu. 77p.
6. C. Perkins et al. 2003. Ad hoc On-Demand Distance Vector (AODV) Routing. IETF RFC 3561.
7. Stevens W. (1994) TCP/IP Illustrated, Volume 1, Addison-Wesley, 576 p

# Multi-hop Cellular Networks: Integrated IEEE 802.11 Ad hoc and Universal Mobile Telecommunications System (UMTS) Networks

Anthony Lo, Jinglong Zhou and Ignas Niemegeers

Delft University of Technology, The Netherlands,  
{A.Lo, J.Zhou, I.Niemegeers}@ewi.tudelft.nl

**Abstract.** Multi-hop cellular network is a new emerging wireless communications system, which leverages the advantages of ad hoc networking and those of cellular networks. This paper presents an architecture for a multi-hop cellular network, which is composed of two popular and complementary technologies, namely the Universal Mobile Telecommunications System (UMTS) and the IEEE 802.11. The two networks are seamlessly connected via a gateway. The gateway composition including self-configuration protocols for addressing, gateway discovery and routing in the multi-hop cellular network were proposed. Finally, we have examined the end-to-end performance of the Transmission Control Protocol (TCP) over the multi-hop cellular network.

**Keywords:** Multi-hop cellular, Ad hoc networks, IEEE 802.11, UMTS

## 1 Introduction

Multi-hop cellular network [1] is a new emerging wireless communications system, which incorporates the ad hoc characteristics into cellular networks. In ad hoc networks, nodes communicate with each other on a peer-to-peer basis without the need of infrastructure support. If direct communication is not feasible between a source node and a destination node, then intermediate nodes are used as routers, which results in multi-hop communication. Cellular networks, on the other hand, rely on the support of fixed infrastructure and require network planning. Since multi-hop ad hoc and cellular networks have different but complementary features, the co-operation between these two networks would leverage the advantage of each other.

In this paper, we discuss a multi-hop cellular network architecture based on the Universal Mobile Telecommunications System (UMTS) and the IEEE 802.11 ad hoc mode. UMTS [2] is a Third-Generation (3G) cellular network which is currently being deployed world-wide. UMTS enables ubiquitous mobile Internet connectivity in a seamless fashion at data rates up to 2 Mb/s. IEEE 802.11 [3] has recently emerged as an important technology for ad hoc networking. IEEE 802.11 operates in the unlicensed frequency band of 2.4 GHz and offers data rates up to 54 Mb/s for the enhanced version, i.e., IEEE 802.11g. The co-operation between UMTS and IEEE 802.11 ad hoc networks poses a new set of problems. Currently cellular networks

including UMTS only allow connectivity of single terminals. If a notebook PC accesses the Internet via UMTS, then the single terminal functions merely as a modem. In multi-hop cellular networks, we no longer have single terminals but a mobile ad hoc network wanting to establish co-operation with UMTS for Internet connectivity. Hence, the functionality of UMTS terminal should be enhanced to play the role of a gateway between the UMTS and IEEE 802.11 ad hoc networks.

The contribution of the paper is twofold. Firstly, it presents an architecture for the multi-hop cellular networks, which includes the design of the UMTS-802.11 ad hoc network gateway and self-configuration protocols for addressing, gateway discovery and routing. Secondly, it evaluates the end-to-end performance of Transmission Control Protocol (TCP) [4]. TCP is a transport protocol employed in the Internet to provide reliable end-to-end data transfer and congestion control. It is used by a large number of Internet applications such as Email, File Transfer Protocol (FTP) and web browsing. TCP is very likely to continue to be the dominant transport protocol in the multi-hop cellular network. To date, the performance of TCP over multi-hop cellular networks has not been investigated yet although the performance of TCP over each individual technology has been extensively studied. Hence, our work advances the knowledge and results published in prior research work by providing an insight into TCP performance in such a network. We have developed simulation modules that model the multi-hop cellular network in the widely used network simulator, *ns-2* [5].

## 2 Related Work

In this section, we review prior related work in the integration of ad hoc and cellular networks.

The authors of [6] described the main issues in the realization of an integrated ad hoc and cellular architecture. Specifically, their work focused on gateway discovery mechanism, mobility and routing. The ad hoc GSM cellular system [7] proposes a relay mechanism that enhances the coverage of GSM networks over dead spots where direct communication with the base station is not possible. Lin and Hsu [8] proposed a multi-hop cellular network where every mobile station participates in relaying traffic. Their work aimed at reducing the number of base stations and used relays to improve coverage. The work of Lin and Hsu was further extended by the authors in [9] to include an explicit control channel, neighbor discovery mechanisms, routing protocol and channel allocation scheme for both best-effort and real-time data.

In [10], the authors proposed a hybrid cellular and IEEE 802.11 ad hoc network architecture. In stead of multi-hop, their work focused on one-hop relay in the ad hoc network. The rationale behind their design was to reduce system complexity, avoid inefficient ad hoc routing, and the impact of inefficient medium access mechanism. The authors in [11] also proposed a hybrid cellular and ad hoc network architecture namely, UCAN, to increase cell throughput without sacrificing fairness. The proposed architecture required every mobile station to have both cellular and IEEE 802.11 ad hoc links. In [12], a similar integrated cellular and ad hoc network architecture as UCAN was proposed. However, the authors primarily focused on reducing connection blocking probability by diverting traffic from congested cells to

neighboring lightly-loaded cells. They used specialized stationary relays which were strategically placed between cells for this purpose. In [13], the channel pool is divided into a set of fixed channels and a set of forwarding channels so that traffic can be redirected to non-congested cells using the forwarding channels.

SOPRANO [14] advocated self-organization at the physical, data link and network layers for the purpose of optimizing the capacity of multi-hop cellular network. The authors of [15] described an integrated ad hoc and cellular network architecture whereby base stations were involved in coordinating peer-to-peer communications in order to increase cell throughput and coverage. The opportunity Driven Multiple Access (ODMA) [16] provides a relaying protocol to enhance cellular coverage and reduces radio transmission power and co-channel interference.

### **3 Application Scenarios**

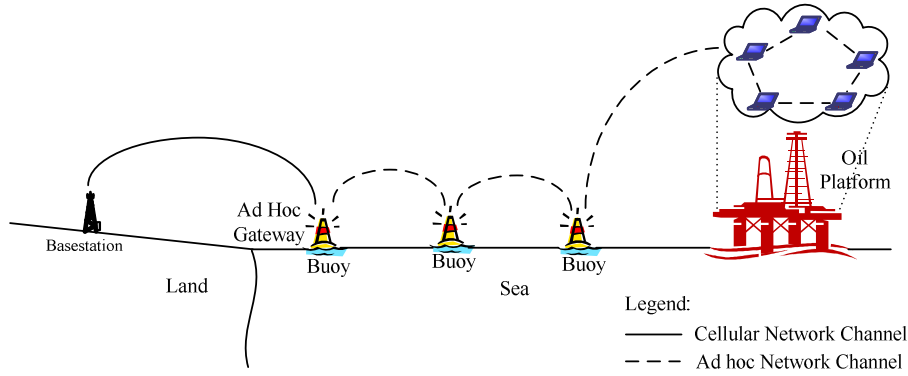
In this section, we present two scenarios to illustrate the practical application of the multi-hop cellular networks. The first scenario extends the reach of UMTS connectivity to mobile users on offshore oil and gas production fields. In the second scenario, we show how the multi-hop cellular network can provide communications in times of calamity – earthquakes, hurricanes, tsunamis, terrorist attacks, etc.

#### **3.1 Offshore Communications**

The coverage of terrestrial infrastructure cellular networks (e.g., UMTS) is limited by the radio transmission range of the basestations. In many remote areas (e.g., sea), terrestrial cellular networks are simply not available. Consequently, satellite technologies are usually used for communications, which are expensive. Figure 1 shows the multi-hop cellular network as an enabling technology for offshore communications, which is cheaper than satellite systems. The multi-hop ad hoc network is used to extend the reach of terrestrial cellular network to mobile users on an oil rig. In the figure, the ad hoc network can be realized using the IEEE 802.11 technology. The ad hoc network is connected to the terrestrial cellular such as UMTS via a gateway.

#### **3.2 Disaster Recovery Operations**

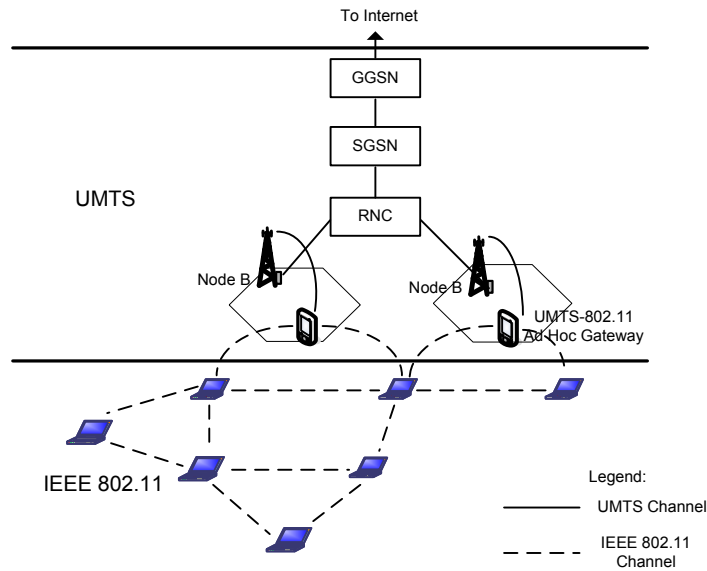
When a natural disaster or terrorist attack strikes, search and rescue operations are usually hampered by communication failure as the incumbent communications infrastructure is damaged or destroyed in the incident area. An ad hoc network, with the support of Voice over IP and video streaming, can be spontaneously set up to restore communication. In addition, the ad hoc network can provide global connectivity if it is connected to the nearest undamaged basestations of the cellular networks. One or more ad hoc nodes function as a gateway for interworking with the infrastructure basestation.



**Fig. 1.** Offshore Communications

### 3 Multi-hop Cellular Network Architecture

The multi-hop cellular network shown in Figure 2 is engineered using two complementary technologies, namely UMTS and the IEEE 802.11 technologies. The architecture of the multi-hop cellular network comprises three tiers: *tier 1* – the UMTS network, *tier 2* – the UMTS-802.11 gateway, *tier 3* – the IEEE 802.11 ad hoc network, which comprises a group of IEEE 802.11 mobile nodes. The protocol architecture for the multi-hop cellular network is shown in Figure 3.



**Fig 2.** Multi-hop Cellular Network Architecture

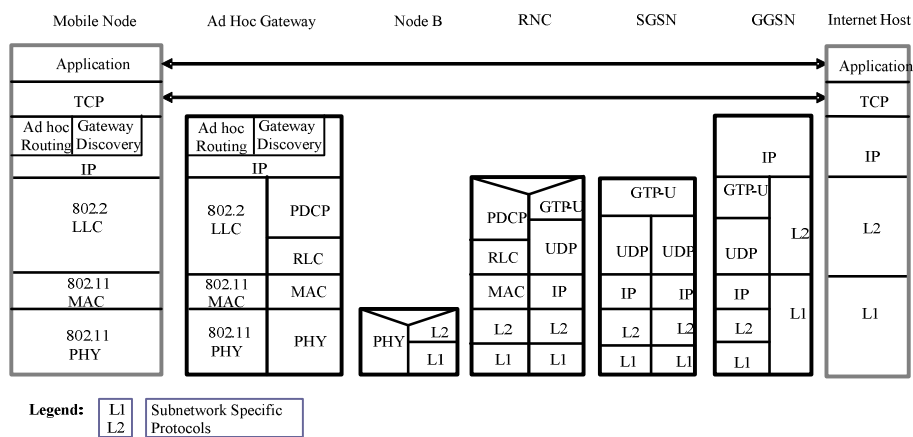
*Tier 1* – the UMTS network provides connectivity to the Internet. The main components of the UMTS network are Node B, Radio Network Controller (RNC),

Serving GPRS Support Node (SGSN), and Gateway GPRS Support Node (GGSN). The Node B sees each UMTS-802.11 ad hoc gateway as a UMTS mobile station. In other words, the IEEE 802.11 mobile nodes are transparent to the UMTS network. The Node B is connected to the RNC which is in turn connected to the SGSN. The SGSN is responsible for routing data packets to the correct RNC from GGSN and vice-versa.

*Tier 2* – the UMTS-802.11 ad hoc gateway is a hybrid device which has two different network interfaces located between the UMTS network and the IEEE 802.11 ad hoc network. On the UMTS side, it contains the UMTS radio access protocol stack, and on the other side, it is the IEEE 802.11 ad hoc mode protocol stack.

*Tier 3* – the IEEE 802.11 mobile nodes are end-user devices. They can either stationary or mobile, and can form a wireless ad hoc network among themselves.

For simplicity, we will refer to the UMTS-802.11 ad hoc gateway and the IEEE 802.11 mobile node as ad hoc gateway and Mobile Node (MN), respectively.



**Fig. 3.** Multi-hop Cellular Network - Protocol Architecture

### 3.1 UMTS-802.11 Ad Hoc Gateway Discovery

For Internet access, an MN must search for an ad hoc gateway that provides Internet connectivity. The ad hoc gateway can be more than one hop away from the MN. The ad hoc gateway discovery is a process by which an MN finds the ad hoc gateways. The ad hoc gateway discovery can be realized proactively or reactively. In the latter approach, the ad hoc gateway discovery is triggered by an MN, while the former is initiated by the ad hoc gateway. To leverage the advantages of both approaches, a hybrid approach is appropriate for the ad hoc gateway discovery.

Instead of defining a new ad hoc gateway discovery protocol, we chose to overlay the ad hoc gateway discovery mechanism on an existing ad hoc network routing protocol. We selected AODV [17] as the ad hoc network routing protocol and defined

two new ad hoc gateway discovery messages: Gateway Advertisement and Gateway Solicitation. Gateway Advertisements are periodically broadcast into the ad hoc network by the ad hoc gateway through its IEEE 802.11 radio interface. Note that, the Gateway Advertisements are not broadcast to the UMTS network. If an MN in an ad hoc network wants to learn about the ad hoc gateway immediately, it can broadcast a Gateway Solicitation which triggers immediate Gateway Advertisements. For instance, an MN will send a Gateway Solicitation message if the frequency of the Gateway Advertisement is low. The advantage of overlaying ad hoc gateway discovery protocol on an existing ad hoc network routing protocol is that routes between the ad hoc gateway and an MN are concurrently constructed during the ad hoc gateway discovery phase using the AODV mechanism rather than separating the gateway discovery from route construction, which leads to shorter route discovery time and lower overhead. The Gateway Advertisement message format includes the ad hoc gateway identification (i.e., IP address), sequence number, Time to Live (TTL), hop count and subnet prefix. Intermediate MNs may only forward the Gateway Advertisement if the number of hops has not been reached, which is defined by TTL.

### 3.2 IP Address Auto-configuration

For Internet connectivity, each MN is assigned a unique and globally routable IP address. We will assume IPv6 since it has much larger address space than IPv4. IPv6 defines two dynamic address allocation schemes, namely, stateful auto-configuration and stateless auto-configuration techniques. The former auto-configuration scheme relies on a Dynamic Host Configuration Protocol (DHCP) [18] server to allocate IPv6 addresses. The server maintains a database containing the necessary information and keeps tight control over the address assignment. In the stateless auto-configuration [19], the MN is more involved in the allocation of the address. The MN generates its own address by combining a subnet prefix with an interface identifier (i.e., the IEEE 802.11 MAC address). Both auto-configuration techniques cannot be used unchanged since they are not designed for multi-hop networks. The stateless auto-configuration is preferred over stateful because it does not rely on a dedicated server, which is natural in an ad hoc environment.

The signaling involved in the IPv6 address allocation is shown in Figure 4. When an MN joins the multi-hop cellular network, it generates a link-local address by adding the interface identifier (IEEE 802.11 MAC address) to the link-local unicast prefix (FE80::/64). The MN uses this link-local address to send a Gateway Solicitation message (step 1). If the ad hoc gateway does not have a subnet prefix and no Packet Data Protocol (PDP) context exists, it must establish a PDP context using the PDP context activation (steps 2-5). The PDP context is a logical connection between the ad hoc gateway and the GGSN. Once a PDP connection is established, the ad hoc gateway is visible to the UMTS network and it can send and receive data packets. Furthermore, the ad hoc gateway issues a Router Solicitation message to the GGSN (step 6), which in turn triggers a Router Advertisement message (step 7). After the ad hoc gateway receives the Router Advertisement message, it constructs a Gateway Advertisement message which includes the subnet prefix contained in the



Router Advertisement. The ad hoc gateway broadcasts the Gateway Advertisement message to the MN (step 8). The MN generates a globally routable IPv6 address by concatenating its interface identifier and the subnet prefix.

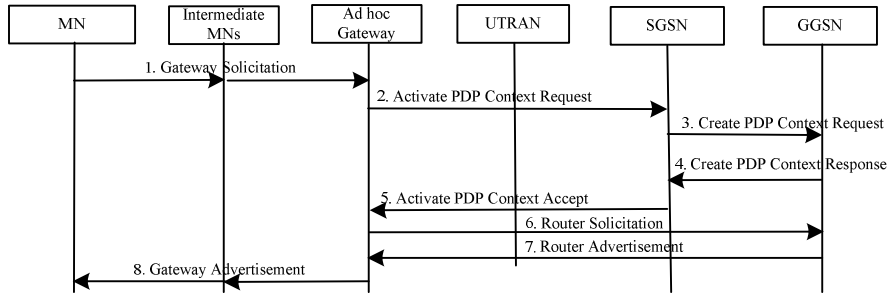


Fig. 4. Modified IPv6 Stateless Address Allocation

## 4 Performance Evaluation

In this section, we evaluate the performance of end-to-end TCP over the multi-hop cellular network.

### 4.1 Simulation Set-up

The end-to-end performance of TCP was evaluated using an event-driven simulator, *ns-2* [5]. *ns-2* supports the TCP/IP protocol suite, several ad hoc routing protocols and the basic IEEE 802.11 ad hoc and infrastructure functionality. The UMTS modules, which were developed in our previous work [20], were used since *ns-2* did not support UMTS. However, the UMTS modules lacked an ad hoc gateway component. Hence, extensions were made to the UMTS modules for modeling the ad hoc gateway. The AODV implementation of *ns-2* was extended to support the gateway discovery protocol. With the extensions in place, instances of the ad hoc gateway together with the IEEE 802.11 MNs, and the UMTS components can be instantiated and formed the simulation topology shown in Figure 5. For the ad hoc network, a chain topology was configured from 1 hop to a maximum of 4 hops. A chain topology was chosen because it is a good example of multi-hop scenario. Each MN was stationary and equally spaced with a distance of 130 m from its adjacent MNs. All the MNs were single-mode 802.11 nodes. The chain topology was connected to the UMTS network via a single ad hoc gateway. A full-duplex UMTS Dedicated radio Channel (DCH) was allocated to the ad hoc gateway. The downlink and uplink bit rate of DCH were set to 2 Mb/s and 384 kb/s, respectively. An independent and uniformly distributed error model was used to model the DCH channel characteristics. The transport block error rate in the range of 0% to 30% was considered. A transport block corresponds to a UMTS MAC data frame. The TCP version employed was New Reno. For ad hoc network, the IEEE 802.11b was used since *ns-2* only supports this version. The gross bit rate offered by the IEEE 802.11b is 11 Mb/s.

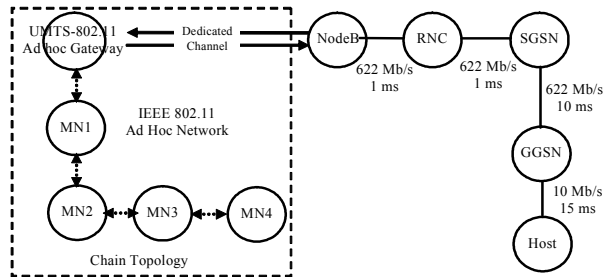


Fig. 5. Simulation Topology

## 4.2 Simulation Results

In the performance evaluation, we consider a single data flow and multiple data flows.

### 4.2.1 Single Data Flow

The performance metrics of primary interest is *throughput*. The throughput at the RLC and TCP layers was obtained using a single file transfer flow between an MN and a fixed host. Data were flowing in the downlink direction from the host to the MN. That means, the only data going in the reverse or uplink direction were the TCP acknowledgements. RLC throughput is defined as the amount of correctly received data (both RLC blocks and control messages) at the RLC layer in bits per second (b/s), excluding the RLC header. TCP throughput (b/s) is defined as the amount of successfully received TCP segments (including header) at the TCP layer. The TCP segment and RLC payload size was set to 512 bytes and 40 bytes, respectively.

In order to simulate long-lived TCP connection, the file transfer session was set to run for 1000 s, which is equivalent to 100,000 UMTS radio frames. Firstly, the simulation was run using the IEEE 802.11 basic CSMA/CA mechanism, and then, the same set of simulation was repeated for the CSMA/CA with Request to Send (RTS)/Clear to Send (CTS). Figures 6 and 7 depict the normalized throughput of RLC and TCP as a function of transport block error rate. Both the obtained RLC and TCP throughputs were normalized with respect to the UMTS downlink channel bit rate of 2 Mb/s. The throughput was evaluated from 1 to 4 hops in the IEEE 802.11b ad hoc network. The throughput for UMTS without the ad hoc part was also evaluated as a performance reference. In the figures, the throughput when the receiver is just the UMTS mobile station without the ad hoc part is denoted as *0-hop* while the other throughputs are labeled with the number of hops in the ad hoc part. For example, *1-hop* refers to an MN when it is directly reachable by the ad hoc gateway. With reference to Figure 5, MN1, MN2, MN3 and MN4 are *1-, 2-, 3-, 4-hop* away from the ad hoc gateway, respectively. The normalized RLC and TCP throughput plots with and without the RTS/CTS mechanism are shown in Figures 6 and 7, respectively.

The attained throughput for RTS/CTS is lower, which is due to the overhead incurred by transmitting the RTS/CTS frames. The normalized TCP throughput plot is non-linear. The non-linear behavior is due to the underutilization of the UMTS radio channel as a result of the large TCP bandwidth-delay product. [4]. In the simulation,

the TCP window size was set to 256 segments (or 131,072 bytes), which was bigger than the maximum allowable TCP window advertisement (65,535 bytes) [4]. Since the delay in the Internet and the UMTS network is fixed, the large bandwidth-delay product is attributed to the large transmission delay over the UMTS radio interface which is the result of RLC retransmission and in-sequence delivery. In the case of 30% transport block error rate, the mean TCP round-trip delay was 1.74 s, which gave a bandwidth-delay product of 435,000 bytes. This would require a TCP window size of approximately 850 segments. The underutilization problem can be overcome by increasing the TCP window size using the TCP Window Scale option, but this option can cause TCP timeouts due to excessive queuing delays at RLC and requires larger buffer size.

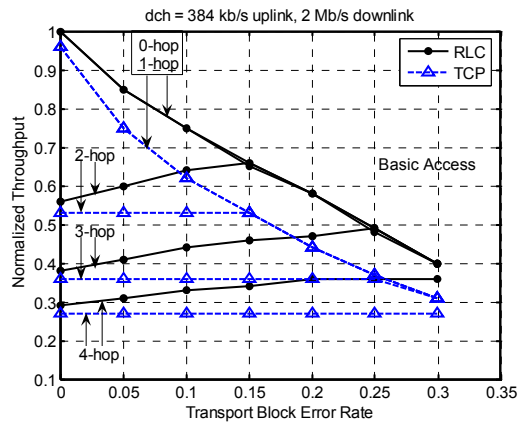


Fig. 6. Throughput versus Transport Block Error Rate for Basic Access

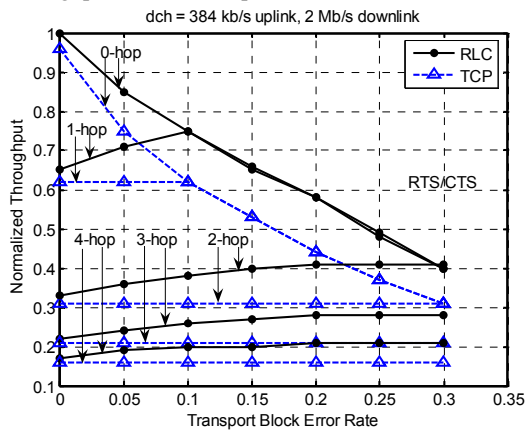


Fig. 7. Throughput versus Transport Block Error Rate for RTS/CTS

An interesting phenomenon was observed in Figures 6 and 7. The TCP throughputs obtained for 2- (only RTS/CTS), 3- and 4-hop scenarios saturated at a fixed level for all the different transport block error rates. For the 2-hop scenario of the basic access case, the TCP throughput saturated at transport block error less than 15%. The simulation traces were examined and did not reveal any timeout or abnormality in

TCP. However, the RLC plot in Figures 6 and 7 indicate that the UMTS radio channel was underutilized. This led us to study the performance of the IEEE 802.11b chain topology alone without the UMTS network. The relevant simulation parameters (e.g., TCP segment size, window size, etc.) for the chain topology were the same as those used in the integrated UMTS-802.11 network simulation. The saturation throughput of TCP obtained from the chain topology alone corresponds to the achievable TCP throughput in Figures 6 and 7. For instance, the TCP throughput for the basic access case for 4-hop scenario was 0.56 Mb/s, which was approximately 28% of the UMTS channel capacity of 2 Mb/s as indicated in Figure 6. For more than 1-hop, the RLC throughput initially increased with the transport block error rate because of redundant retransmissions of RLC data. The retransmission is controlled by a fixed value timer, which is non-optimal for higher transport block error rate.

#### 4.2.2 Multiple Data Flows

We now consider multiple competing data flows in the downlink direction with the same network topology as in the previous subsection. In this case, each MN was allocated an FTP flow. Therefore, we have a total of four FTP flows that are active concurrently. All of the FTP flows are originating from the same fixed host. The first to the fourth FTP flows were assigned to MN1 to MN4, respectively. The FTP flows were sequentially started with a 10 s delay between two successive FTP flows. The fourth FTP flow was set to commence the earliest at 0 s then followed by the third FTP flow which started 10 s later, and the first FTP flow begins at 30 s. All of the FTP flows ended at the same time 600 s. It is important to note that all the flows share a single DCH channel. Even though, the FTP flows commenced at different times, the throughput was only computed over the period at which all the FTP flows were concurrently active. We computed the TCP throughput attained for each FTP flow and also the aggregate throughput. In addition to computing the throughput, we also determined the *fairness index* ( $f$ ), which is defined as [21]

$$f = \frac{\left(\sum_{i=1}^n x_i\right)^2}{n \sum_{i=1}^n x_i^2}$$

where  $n$  is the number of concurrent FTP flows and  $x_i$  denotes the throughput achieved by the  $i$ th flow. In our case,  $n$  is equal to 4 and flows 1, 2, 3 and 4 correspond to the FTP flow at MN1, MN2, MN3 and MN4, respectively. The fairness index is used to investigate if the UMTS and the IEEE 802.11b radio channels are equally shared among all the FTP flows. When  $f$  equals to 1, it indicates that both the UMTS and the IEEE 802.11b radio channels are equally shared by all the competing flows. The TCP throughput plots and fairness indices with and without RTS/CTS are shown in Figures 8 and 9, respectively. The aggregate TCP throughput for the basic access mechanism is higher than the RTS/CTS, which is consistent with the throughput results obtained for the single flow case. The fairness indices were less than 1 for all the different transport block error rates. As observed in Figures 8 and 9, the TCP throughput of flow 1 was approximately two times larger than other flows, which leads to unfairness. Flow 1 gained a larger share of the radio channel because it had the shortest round-trip time, which encouraged the TCP window size to increase faster.

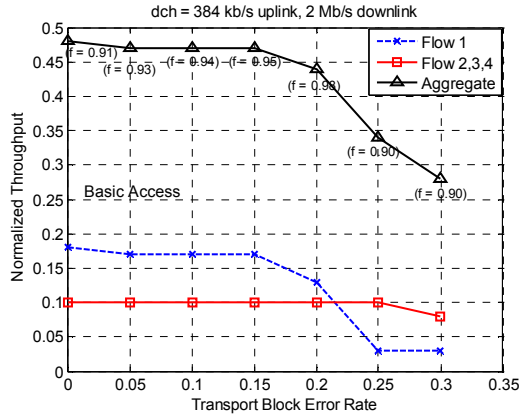


Fig. 8. TCP Throughput versus Transport Block Error Rate for Basic Access

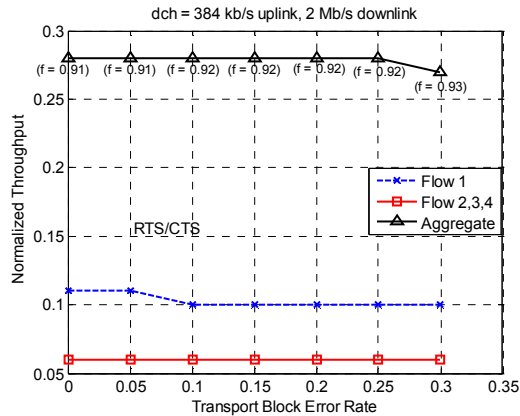


Fig. 9. TCP Throughput versus Transport Block Error Rate for RTC/CTS

## 5 Conclusion

The paper has proposed a multi-hop cellular network which is composed of the UMTS and IEEE 802.11 ad hoc networks. In such a network, the IEEE 802.11-enabled nodes can get ubiquitous Internet connectivity via UMTS. A UMTS-802.11 gateway is used to seamlessly interwork the UMTS and IEEE 802.11 ad hoc networks. The gateway composition and self-configuring protocols for addressing, gateway discovery and routing were discussed. We have evaluated the end-to-end performance of TCP over the multi-hop cellular network. Simulation results show that multi-hop cellular network using the IEEE 802.11 basic access mechanism outperforms the access mechanism with RTS/CTS. The IEEE 802.11 ad hoc network can be the performance bottleneck if the number of hops is more than one. When there are multiple data flows, the UMTS radio channel resource can be unfairly shared. The data flows, which span the least number of hops in the 802.11 ad hoc network, will gain a larger share of the UMTS radio resource.

**Acknowledgments.** This work was partially funded by the Freeband PNP2008 project.

## References

1. P. H. J. Chong, F. Adachi, S. Hamalainen and V. Leung, "Technologies in Multi-hop Cellular Network", *IEEE Communications Magazine*, vol. 45, no. 9, 2007.
2. H. Kaaranen, A. Ahtiainen, L. Laitinen, S. Naghian, and V. Niemi, *UMTS Networks: Architecture, Mobility and Services*, John Wiley & Sons, New York, NY, 2005.
3. IEEE Standard 802.11. *Wireless Medium Access Control and Physical Layer Specification*. 1999.
4. W. Stevens, *TCP/IP Illustrated, Vol. 1*. Addison-Wesley, Reading, MA, 1994.
5. K. Fall, and K. Varadhan, "The ns Manual", available at <http://wwwwi.isi.edu/nsnam/ns/ns-documentation.html>
6. N. Bayer et al. "Integration of Heterogeneous Ad Hoc Networks with the Internet", *Proceedings of the International Workshop on Wireless Ad Hoc Networks (IWWAN'05)*. London, UK, 2005.
7. G. N. Aggelou, and R. Tafazolli, "On the Relaying Capacity of Next-Generation GSM Cellular Networks", *IEEE Personal Communications Magazine*. 2001.
8. Y. Lin, and Y. Hsu, "Multihop Cellular: A New Architecture for Wireless Communications", *Proceedings of IEEE INFOCOM*, Tel Aviv, Israel, 2000.
9. B. S. Manoj, R. Ananthapadmanabha, and C. Siva Ram Murthy, "Multihop Cellular Networks: Architecture and Protocols for Best-effort and Real-time Communication", *Journal of Parallel and Distributed Computing*. 2005.
10. H. Wei, and R. D. Gitlin, "Two-Hop-Relay Architecture for Next-Generation WWAN/WLAN Integration", *IEEE Wireless Communications*. 2004.
11. H. Luo, R. Ramjee, P. Sinha, L. Li, and S. Lu, "UCAN: A Unified Cellular and Ad Hoc Network Architecture", *Proceedings of the Annual International Conference on Mobile Computing and Networking (MobiCom'03)*. San Diego, USA, 2003.
12. H. Wu, C. Qiao, S. De, and O. Tonguz, "Integrated Cellular and Ad hoc Relaying Systems: iCAR" *IEEE Journal on Selected Areas in Communications (JSAC)*. 2001.
13. X. Wu, S. Chan, B. Mukherjee, and B. Bhargava, "MADF: Mobile-Assisted Data Forwarding for Wireless Data Networks", *Journal of Communications and Networks*, 2000.
14. A. Zadeh, B. Jabbari, R. Pickholtz, and B. Vojcic, "Self-organizing Packet Radio Ad Hoc Networks with Overlay (SOPRANO)", *IEEE Communications Magazine*. 2002.
15. H. Hsieh, and R. Sivakumar, "On Using Peer-to-Peer Communication in Cellular Wireless Data Networks", *IEEE Transaction on Mobile Computing*. 2004.
16. T. Rouse, I. Band, and S. McLaughlin, "Capacity and Power Investigation of Opportunity Driven Multiple Access (ODMA) Networks in TDD-CDMA Based Systems", *Proceedings of the IEEE International Conference on Communications (ICC)*. New York, USA, 2002.
17. C. E. Perkins, E. Belding-Royer and S. Das, "The Ad hoc On-demand Distance Vector Protocol". In *Ad hoc Networking*, C. E. Perkins (editor), Addison-Wesley, 2001.
18. R. Droms et al., "Dynamic Host Configuration Protocol for IPv6", RFC 3315, 2003.
19. S. Thompson and T. Narten, "IPv6 Stateless Address Autoconfiguration", RFC 2462, 1998.
20. A. Lo, G. Heijnen and I. Niemegeers, "Performance Evaluation of MPEG-4 Video Streaming over UMTS Networks using an Integrated Tool Environment", *Proceedings of SPECTS*, 2005.
21. R. Jain, *The Art of Computer Systems Performance Analysis*, John Wiley & Sons, NY, 1991.

# Speech-based user interaction for mobile devices

Angel M. Gómez, Victoria Sánchez, José A. González, Antonio M. Peinado \*

Dpto. de Teoría de la Señal, Telemática y Comunicaciones, University of Granada  
{amgg,victoria,joseangl,amp}@ugr.es

**Abstract.** In this paper we review the technologies devoted to provide a faster, more comfortable and flexible user interaction with mobile devices through voice. We focus on the concept of speech enabled services (SES) and their implementation by means of remote speech recognition (RSR). In particular, we describe the available architectures to access these services, namely, the ready-to-go network-based approach, on one hand, and the promising distributed speech recognition architecture, on the other. In addition, we explore the use of context information (user context, network context, ...), a recurrent issue in mobile services, to improve the RSR user experience. Finally, some of latest contributions of our research work are briefly described.

**Keywords:** Mobile interaction, new interfaces, speech enabled services, remote speech recognition, distributed speech recognition, context-aware speech recognition.

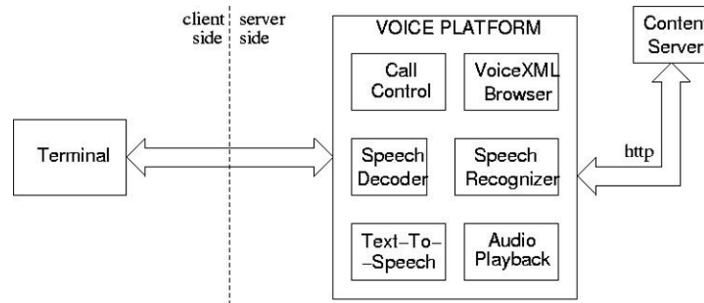
## 1 Introduction

The ubiquitous and pervasive access to information services has become not only desirable but almost necessary. However, mobile and portable devices devoted to provide such service access, make it difficult or even frustrating due to their required small size and inherent weight restrictions. Speech technologies can not only solve these hurdles but also extend this access to special circumstances, for example, driving or in a emergency, as well as help people with visual impairments. This has emerged the concept of speech-enabled services (SES) [1], i.e. services which it is possible to interact with by means of voice.

It is evident that automatic speech recognition (ASR) is an essential component in SES systems. However, there exist serious limitations to include an ASR subsystem in a portable device. Although possible, the embedded recognition system should be very limited or its memory and computational requirements would go beyond current mobile device capabilities and those foreseen in an immediate future. Thus, instead of installing an ASR subsystem in the client terminal which sends information requests to a remote server, an approach based on the remote recognition of speech is proposed. Under this approach, the client device just sends the speech signal or a parameterization of it to the server, while

---

\* Work supported by project MEC/FEDER TEC2007-66600.



**Fig. 1.** Speech Enables Service (SES) conceptual scheme.

the recognition itself takes place on the remote server. This approach is usually known as remote speech recognition (RSR). Figure 1 depicts a diagram of a SES system using this approach.

The advantages of the RSR approach are clear. First, there is no need to install a complex ASR subsystem in the terminal user, so thin portable clients can be used. Second, since speech recognition takes place at the server, this can be much more powerful and flexible. In addition, the client is not limited to any language, as it only transmits speech data.

Depending on how and what speech data is transmitted, two different architectures can be considered [2]:

- In Network-based Speech Recognition (NSR) architectures [3], speech is directly transmitted by a speech coder, for example, ITU-T G729 [4] or G723.1 [5], ETSI Adaptive Multi-Rate (AMR) [6] or iLBC [7]. Prior to recognition, speech must be decoded and parametrized, so all the main ASR tasks (feature extraction and pattern recognition) are performed, from the point of view of the client, by the network side.
- On the contrary, in Distributed Speech Recognition (DSR) architectures [8], speech signal is analyzed and recognition-oriented features extracted at the client side. In such a way the portable device transmits only the parameters required by the recognition system at the server. Currently there are four DSR standards developed by ETSI [9–12], which reflects the potential of this approach.

The choice between these architectures on digital channels is not easy since it depends on the terminal intended for the application (cell phones without DSR capability, DSR-enabled mobile phones, personal computers, laptops or PDAs), the network type (GSM / GPRS / UMTS network or WLAN) and the robustness against degradation sources.

In remote speech recognition two key sources of degradation can be considered. One of these sources is acoustic noise, consequence of the highly variable environment in which the user will interact (conference halls, airports, stations, ...). To combat this kind of degradation a number of techniques have been proposed to provide speech extraction parameters more robust against acoustic



noise, such as Wiener filters, voice activity detectors and feature normalization [2].

The second kind of degradation is due to the distortion introduced by the transmission channel. It is well known that any wireless communication, as in mobile devices, involves a hostile channel which causes a distortion in the speech transmission. This distortion can cause from bit level errors to complete packet losses (depending on the transmission technology) but, in any case, it results in a performance reduction on RSR systems. Speech recognition systems are, to some extent, used to dealing with acoustic noise. However, performance reduction due to channel errors is novel and particular of RSR systems. Thus, in recent years, a number of research groups have worked to improve RSR robustness against this distortion. Our group has been one of them, providing different techniques oriented to improve speech recognition performance over digital channels, in particular, in IP and packet-switched networks.

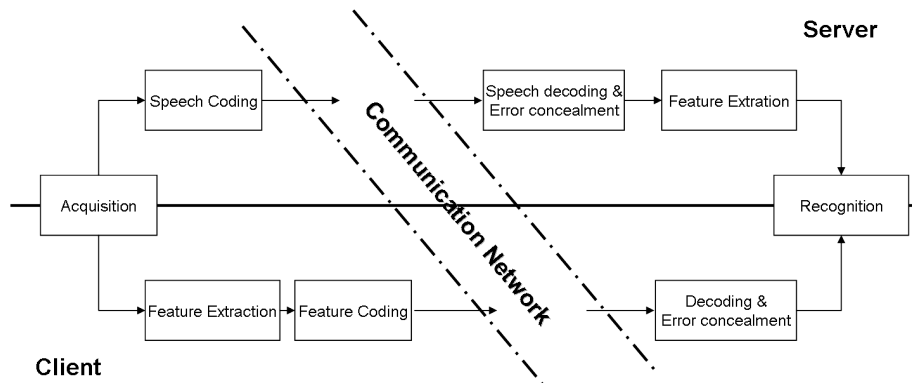
In this paper we briefly describe the RSR related technologies which will allow speech-based interaction with mobile devices. We first review the aforementioned architectures available for remote speech recognition, namely, Network-based and Distributed speech recognition (section 2). Then, we will focus on a very promising approach derived from the anywhere-anytime capability provided by RSR, the context-awareness in speech recognition (section 3). This recent approach proposes the exploitation of all the available context information (computation context, user context and physical context) in order to enhance the quality of speech recognition and the user experience. In addition, we will briefly describe some of the recent developments of our research group, in particular, our proposal for robust channel coding in DSR, our transcoding approach for NSR systems and an example application which is being used as common framework to test our techniques (section 4). Finally, section 5 is devoted to summarize our conclusions.

## 2 Architectures for Remote Speech Recognition

Speech recognition can be seen as a pattern recognition problem in which we can distinguish three different tasks:

1. Speech acquisition. The speech signal obtained from a microphone is sampled and quantized so that a digital signal  $x(n)$  is available.
2. Speech parametrization. The speech signal is segmented into frames of 15-30 ms. Every frame is processed in order to obtain a corresponding vector of parameters (feature vector) with representative speech information useful for recognition (spectral envelope, energy, etc.). Thus, the original signal is transformed into a sequence  $X = (\mathbf{x}_1, \mathbf{x}_2 \dots, \mathbf{x}_T)$  of feature vectors.
3. Pattern matching. The final recognition is usually carried out as a MAP estimation as follows:

$$W^* = \arg \max_W P(W|X) = \arg \max_W P(X|W)P(W) \quad (1)$$



**Fig. 2.** Architectures for Remote speech recognition: Network-based speech recognition (above) and Distributed speech recognition (down).

where  $W^*$  is the recognizer text,  $W$  represents any possible text and  $X$  is the feature vector sequence. The required probabilities  $P(X|W)$  and  $P(W)$  are obtained from statistical models known as acoustic model and language model, respectively. Hidden Markov Modeling (HMM) is the predominant approach for acoustic modeling, while a number of different approaches are used for language modeling [2].

These three tasks do not need to be implemented at the same place, being possible to distribute them in different locations as long as there is a communication network between them.

Depending on where we place the communication network in the processing chain, we can distinguish between two alternative architectures for remote speech recognition. In the NSR architecture, the client only has to acquire and transmit coded speech, while all the processing is performed at the server. On the other hand, the DSR architecture distributes the extraction the speech features to the client whilst the classification process, or recognition itself, is performed at the server.

## 2.1 Network-based Speech Recognition

The NSR approach is based on the use of a speech codec to transmit speech information to the remote recognition server. Figure 2 (above) depicts this architecture. Three different types of speech codecs can be used in this architecture: waveform, parametric and hybrids ones. We will briefly consider the basis underlying these three types.

In waveform codecs the goal is to obtain a decoded signal that reproduces the original signal sample by sample. These codecs usually operate at medium bit-rates (2 bits per sample approx.) and show an acceptable robustness against acoustic and transmission channel distortions. The basic technologies include

time domain codecs based on PCM (G.711 standard) [13], DPCM and ADPCM (standards such as G.726 [14]). Waveform codecs can be also implemented in the frequency domain [15], by means of transform and subband coding, mainly applied to general audio encoding (MPEG standard).

Parametric codecs, traditionally known as vocoders [16, 17], employ a model of voice production. They work on a frame basis so that, for each frame, a set of model parameters are transmitted. These parameters (vocal tract information, voiced or unvoiced classification and fundamental frequency or pitch) are used to reconstruct or synthesize the original speech. This approach is not intended to provide a decoded signal identical sample by sample. Instead the goal is to achieve a signal perceptually equivalent to the original one.

Finally, hybrid codecs can be considered a mixture of waveform and parametric ones [18, 19]: They use a parametric model but also preserve the signal waveform. To do so, a technique known as analysis by synthesis (AbS) is used. By inserting the decoder into the encoding procedure, it is possible to know in advance the waveform of the decoded signal. Then, original and decoded signals can be compared and an optimal encoding (in terms of the minimum mean square error) obtained. An special mention deserves some hybrid codecs designed specifically for Voice over IP (VoIP). Here we include G.723.1 [5], G.729 [4], AMR [6], and the more recently proposed iLBC [7] (internet low bit-rate codec). This codec, very popular in several VoIP systems, avoids any inter-frame dependencies, allowing a very efficient packet loss concealment.

Despite the differences among codecs, it is clear that NSR systems must rebuild the speech signal (through the corresponding decoder) and parametrize it in a convenient way for recognition. This decoding is usually damaging for the speech recognition process [20, 21], reducing its accuracy. Therefore, a variant of the NSR architecture, which uses a direct transcoding from speech transmission parameters to speech recognition features, has been extensively researched in the last years [22–24].

## 2.2 Distributed Speech Recognition

In the DSR architecture the client device encodes and transmits only the required parameters for recognition (see figure 2) so that, the use of a speech codec as seen in the previous subsection is not required. In this way, the inconveniences of using a codec (i.e. performance reduction) are avoided, achieving a high performance recognition which is robust against several types of degradations, requiring very low bit-rates (2-5 kbps in comparison with 5-16 kbps of traditional codecs).

DSR technology is now well-established through the standardization efforts of ETSI [9–12]. The key elements of this architecture are the speech parametrization and the encoding and decoding (including the mitigation of transmission errors) of the speech parameters, as well as the client robustness against environmental noise [10].

The most common parameterization in DSR is that based on Mel frequency cepstrum coefficients (MFCCs). This is due to the robustness of these parameters

against changes in the acoustic environment. The procedure to obtain a vector of MFCC parameters is rather simple and can be briefly described as follows. First, a fast Fourier transform (FFT) is applied over each frame of speech signal (20-30 ms). Then, FFT is decimated by means of a filter bank and the cosine transform (DCT) applied to the logarithmic outputs of the filter bank. The filter bank responds to a non-uniform scale, known as Mel scale, which tries to replicate the non-uniform frequency resolution of human hearing. In addition, energy related parameters are usually added to feature vectors and, in the server side, temporal derivatives of these parameters are computed and also included as dynamic speech information.

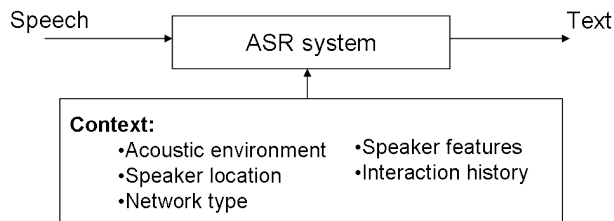
The parameter coding can be performed in several ways. A first approach is to use a simple scalar quantization, although more efficient encoding schemes that use vector quantization (VQ) and prediction have been tested successfully [25, 26]. The standard proposed by the ETSI [9] is based on VQ quantization, in particular on the split VQ (SVQ) where groups of speech features (feature pairs using the notation in the ETSI standard) are independently VQ quantized.

### 2.3 Architecture selection

As has been shown, there exist two possible architectures for RSR, namely, Network-based and Distributed speech recognition. Both architectures are equally valid, however, both present advantages and disadvantages which decide positively or negatively on their feasibility and desirability in one or other context. Thus, the choice between these two architectures should be done taking into account the following factors:

- Bandwidth restrictions: DSR typically requires a lower bandwidth.
- Terminal class which the application is oriented to (WiFi phones, mobile phones without DSR capabilities, DSR-enabled mobile phones, laptops or PDAs) and network type (GSM / GPRS / UMTS or WLAN).
- Transmitted speech availability: some secure applications require the complete speech signal at the receiver (for example, for identity authentication). Although some ETSI DSR standards [11, 12] allow speech reconstruction from feature vectors, NSR provides a better signal quality.
- Robustness against acoustical noise and channel distortions. In general, DSR is more robust, but the NSR architecture can be improved in these areas.

Remote speech recognition based on DSR turns out to be very promising in the two most extended digital channels today, that is, IP and mobile wireless networks. In the first case, the client-server architecture proposed in DSR naturally fits in IP-based networks, since such architecture has been extensively used by many other services already available. On the other hand, in mobile networks (as in GPRS and UMTS ones), the higher protection applied in data traffic channels (those used for DSR) make feature transmission more robust to channel distortions. Additionally, the bandwidth necessary for the speech transmission is significantly lower.



**Fig. 3.** Context-aware speech recognition.

On the contrary, in the short term, NSR is the only architecture available to enable speech recognition on mobile devices. NSR systems have the advantage of allowing the use of the current mobile terminals, such as mobile or WiFi phones. However, the largest difficulty of this architecture lies in the reduction of recognition performance caused by transmission errors and the coding/decoding process. Due to this, research on NSR has not been abandoned.

### 3 Context-awareness in speech recognition

The importance of context-awareness in the future services over wireless networks has been recognized by the technological platform eMobility in its Strategic Research Agenda [27]. In the case of speech recognition for user interaction in mobile devices, context-awareness has the meaning of adaptation to a number of environmental parameters. Thus, in addition to the speech signal, the context information is exploited in order to improve the system performance and, in general, the dialog between user and service. This idea is depicted in figure 3.

As can be seen in the figure, there are different types of information to be taken into account in order to carry out context adaptation, but they can be classified as follows:

1. Computation context (network type, user device, ...). The network type determines aspects like the format of the data transmitted from the client to the server or the type of error concealment that must be applied (bit error mitigation for wireless transmission, packet loss concealment for IP networks, ...). The type of user device may determine the RSR architecture employed (DSR for PDAs and smart phones or NSR for normal cellular phones).
2. User context (user profile, position, dialog records, ...). This context is mainly related with the dialog that the user maintains with the server. Therefore, regarding speech recognition, this context can be employed to adapt the language models. In particular, the application example presented in the next section uses the GPS user position to make the dialog more flexible.
3. Physical context (acoustic environment). This topic has been widely treated in the literature as robustness against acoustic noise. In this case, the techniques that have been proposed include speech enhancement, feature compensation and model adaptation.

As can be seen, the above issues have already been treated in some manner in the field of user interaction, although under an independent vision. The concept of context-awareness opens a new different and unitary vision to the environment adaptation problem. The use of this concept in the field of ASR is relatively new and appears as the next step in the development and deployment of RSR systems [28, 29].

Regarding the adaptation to the physical context (acoustic environment), the predominant context-aware approach is that of recognizing the acoustic environment, that is, the type of noise which envelops the user. Once the acoustic environment and its corresponding average SNR is identified, it is possible to employ a set of acoustic models specifically trained for it [30, 29] so that the mismatch between training and user acoustic conditions can be minimized. It is important to observe that under this approach, we are not considering the instantaneous noise which corrupts the speech signal at every specific instant, but the average acoustic environment (airport, train station, restaurant, street, exhibition, ...). Therefore, this approach can be used in combination with classical noise reduction techniques. We must also note that it is impossible to store a set of acoustic models for every possible acoustic ambient, but a finite set of *selected* ambients. Thus, we can expect that in a real situation, the system will identify the closest ambient amongst the set of considered ambient, so that a residual mismatch will remain. In order to reduce it, Jacobian model adaptation and feature compensation based on MMSE estimation and Vector Taylor Series have been proposed [31]. The environmental noise has been classified in different ways, using Hidden Markov Models [32], Gaussian Mixture Models [30], or Gaussian CDFs [31].

## 4 Recent developments in our research group

In recent years, a number of research groups have worked to improve the RSR approach. In particular, several techniques have been developed to provide speech extraction parameters which are robust against acoustic noise (by means of Wiener filters, spectral attenuation, introduction of voice activity detectors and feature normalization). In addition, coding (predictive codes, transform coding, scalable encoding [2]) and decoding algorithms (interpolation, estimation techniques, missing data and soft data [2]) have been proposed in order to make speech parameters robust against channel distortions.

In the last years, our research group has developed different techniques oriented to improve remote speech recognition performance over digital channels. In particular, we have been focused on IP and packet-switched networks, since currently there is a notable stream of research to what it is known as all-IP convergence. In this section, we describe several novel techniques that have been recently published.

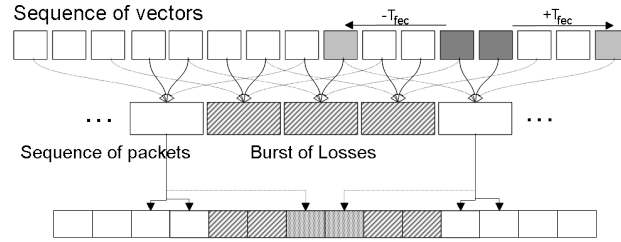
#### 4.1 Robust channel coding for DSR

As we have mentioned, DSR is very suitable for IP networks. However, when transmitting real-time data over a packet switched network (such as IP), one of the most common problems encountered is that of packet loss. Since IP networks were designed to offer a best effort service, they are unable to offer a reliable and quality packet delivery. Thus, on congested IP networks, routers will discard packets if their input flow exceeds their output flow for a given data route. Furthermore, irrecoverable errors in the wireless link commonly utilized by handheld devices will result in packet elimination at the back-end. In these scenarios, packet losses tend to appear consecutively and, in speech recognition, this burst-like nature causes the most negative impact. In fact, DSR has shown to be tolerant to very high loss ratios as long as the average burst length is reasonably short (one or two frames) [33].

Novel receiver-based techniques have recently been proposed to mitigate the effect of packet losses on the performance of DSR systems. These techniques exploit the redundant information present in the speech signal to achieve a better recognition performance than the standard reconstruction method based on replacing the lost vectors by copies of the nearest received ones. Reconstruction based on data-source modeling and the maximum a posteriori (MAP) estimation method [34] are examples of this. However, receiver-based concealment is usually limited in the sense that it implicitly relies on the assumption that the signal segment to be recovered is in steady state.

Our research group has explored the application of the so called sender-driven recovery techniques, which assume the participation of the sender in order to recover from an error. In particular, we have considered the application of media-specific FEC schemes in which a replica, a strongly quantized version of the current feature vector, is repeated in another packet. Thus, we start from a very simple but effective FEC strategy based on the replication of the feature vectors in packets greatly separated in time [35]. Let us suppose that, along with the feature vectors corresponding to the current frame pair, we also include in the packet replicas of the feature vectors corresponding to the frames located  $T_{fec}$  time units before and after the current frame pair. Each packet would then be composed of four frames (figure 4, further details are found on [36]). In this way replicas can be used not only to recover some lost frames, but also to break bursts of losses into shorter ones. For example, as can be observed in figure 4, two replicas (marked in gray) are introduced in the middle of the burst, breaking it into two halves.

Replicas can be further exploited in different ways. One of them is to enhance them using a Forward-backward Minimum mean square estimation (FB-MMSE) as it is proposed in [35–37]. The other approach is to treat them, along with the lost frames, at the recognition stage itself. In order to do so, Weighted Viterbi Recognition (WVR) [38, 39], a modification of the Viterbi Algorithm (VA) whereby the confidence in the decoded feature is taken into account, can be applied. As advantage, the replicas degradation (due to the strong quantization) and losses are treated at the recognition stage, where the rich speech



**Fig. 4.** FEC scheme: each frame pair is sent along with a media-specific FEC containing information about distant frames.

source model utilized by the recognizer can be exploited. Further details of this technique can be found in [36].

Additionally, in order to achieve better results, this FEC scheme based on VQ replicas can be combined with frame interleaving strategies. A frame interleaver permutes the order in which complete frames are transmitted. As a consequence, when frames are restored into their original order at the receiver, consecutive frame erasures are perceived as shorter bursts. The main hurdle is that a direct composition of FEC and interleaving schemes results in a sum of their corresponding delays.

To solve this problem, our group recently proposed a double stream (DS) strategy whereby this sum is avoided. In this scheme, two independent sequences of feature vectors are considered: a primary stream which consists of feature vectors SVQ quantized, as in the DSR standard, and a secondary stream composed of the same vectors but VQ quantized. This second flow is just a redundant stream inserted within the packets by means of VQ replicas. Initially, vectors from both streams are ordered according to their time instant. Then, different interleaving functions are applied over each one. Thus, since both streams are independently interleaved, the total latency caused in the transmission is equal to the maximum of both interleavers.

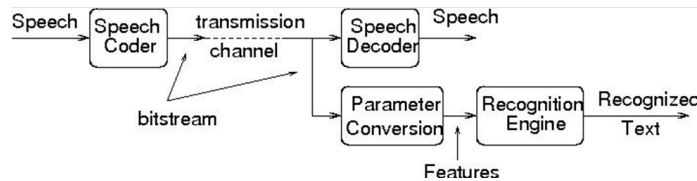
This strategy has shown a considerable improvement on recognition accuracy. Assuming a short delay of 120 ms, it is possible to achieve an acceptable recognition accuracy (up to 97%) in a very hostile channel with a 30% of packet loss and average bursts lengths of 6 packets (12 DSR frames) [40].

## 4.2 Transcoding Solution for NSR

The most obvious way to recognize speech under an NSR approach is by using the decoded signal. However, it is also possible to perform speech recognition from parameters directly extracted from the bit-stream, that is, without rebuilding the speech signal (see figure 5). This approach, which is commonly known as transcoding or Bitstream-based NSR(B-NSR), has some advantages over basic NSR:

- One of the reasons that NSR gives a lower performance than DSR is due to the post-processing performed by speech decoders. Although this post-





**Fig. 5.** Transcoding operation.

processing improves the perceptual quality of speech, it negatively affects the recognition. By means of transcoding, we can avoid this post-processing.

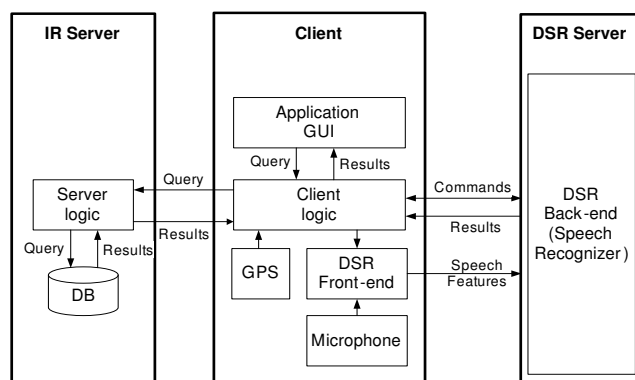
- The spectral parameters and the excitation information can be treated independently in a transcoding approach. With appropriate mitigation algorithms, this independence can be exploited to improve the robustness to transmission channel degradations.
- As it is not strictly necessary the complete reconstruction of the speech signal, transcoding saves calculation.

The objective of this transcoding approach is to combine the advantages of NSR and DSR. On one hand, the user terminal remains unmodified, that is, specific DSR hardware is unnecessary. On the other, the same speech recognition performance as the DSR approach is achieved. Of course, the main problem is to achieve a precise transformation of parameters from speech coding to speech features for recognition. Speech recognition from speech codec parameters and transcoding has been researched by a number of authors [41–44]. Our research group has already proposed and published two different transcoders. The first is designed to work over the EFR bit-stream and it is oriented for mobile networks such as GSM [24]. The second one works on the iLBC codec and is designed for packet-switched networks such as IP [45]. Both transcoders have shown significant benefits in terms on recognition accuracy in adverse transmission channel conditions.

### 4.3 Application example: An speech-enabled yellow pages application with GPS information

In our research group we have developed a yellow pages based application as a fast and simple framework to show the previous technologies. In this application, the users can request information about the location of an service. For example, requests such as *Where can I find a restaurant in New York?*, *Where is the nearest hotel?*, ...; can be used. As can be inferred from the previous requests, the user can specify the place where he is looking for the service or not. In the later case, the information provided by a global positioning system (GPS) device is used. Finally, the system response consists of a map showing the position of the searched establishments and information about them (name, street name, telephone number, ...).

The application has been implemented in a personal digital assistant (PDA), thus, the users can use it anywhere, anytime. However, a PDA can be very limited in both interfaces (e.g., small keyboards, small displays, ...) and performance (reduced CPU performance and limited battery life). The first limitation justifies the use of an oral interface in the application. The second limitation prevents that a traditional oral interface can be implemented in these kind of mobile devices. Thus, a fully remote speech recognition approach is used. The main components of this architecture are the *client* implemented in the user terminal (PDA), the speech recognizer (*DSR server*) implemented in a remote server, and the information retrieval (IR) server (*IR server*). The overall architecture of the system is depicted in figure 6.



**Fig. 6.** Architecture of the speech enabled yellow pages system.

As a first step, the speech recognition capabilities have been implemented using a DSR architecture where the recognition processing is split into a DSR client based on the ETSI standard advanced front-end and a recognition server. Speech features are transmitted from the client to the server over a packet network (GPRS, UMTS, HSDPA, Wi-Fi, ...) using the corresponding DSR recommendation for IP networks [46]. For generality shake, the recognized text is sent from the remote server back to the client which makes the corresponding request to an IR server (this request could be directly made by the server). Finally, the IR server receives the user's requests about the services that the user is looking for and returns a map with their position.

## 5 Conclusions

In this paper we review the latest techniques and architectures devoted to provide a user interaction with mobile device services based on speech and so, make it faster, more comfortable and flexible. We have describe the two available architectures to access speech enabled services, NSR and DSR. While in NSR a

usual speech coder is used, a specific feature extractor is used in DSR. As we have shown, both architectures are equally valid but, since they present different advantages and disadvantages, their choice should be done taking into account some considerations as user device, network type and robustness against noise and channel distortion, among others.

Another relevant issue is the exploitation of the context in the mobile interaction. This well known topic in mobile services is also a key aspect in SES services, where context information can be exploited to improve the dialog between user and service. In this paper we expose and briefly describe three different types of contextual information: computation, user and physical context and present how these are already being used to improve the RSR user experience.

In addition, we have described some of the recent contributions of our research group to this area. Thus, by means of an integrated scheme that involves FEC codes, interleaving and weighted viterbi recognition, we provide a robust coding for lossy packet channels such as IP networks. On the other hand, ASR performance is improved in NSR architectures by means of transcoders. Our group has developed two of them (one for EFR/AMR and the other for iLBC) which directly transform speech codec parameters to features vectors, avoiding undesirable post-processing damaging effects. Finally, a simple but interesting application, which is currently being used as integrating platform that combines ASR capabilities and context information, is shown as example of the potential of SES services.

## References

1. D.Pearce, "Enabling new speech driven services for mobile devices: An overview of the ETSI standards activities for Distributed Speech Recognition front-ends," *AVIOS 2000*, 2000.
2. A.M. Peinado and J.C. Segura, *Speech recognition over digital channels. Robustness and standards*, Wiley, 2006.
3. T.Fingscheidt, S.Aalburg, S.Stan, and C.Beaugeant, "Network-based vs. distributed speech recognition in adaptative multi-rate wireless systems," *ICSLP-2002*, 2000.
4. *Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear prediction (CS-ACELP)*, ITU-T G.729 Recommendation, 1993.
5. *Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s*, ITU-T G.723.1 Recommendation, 1996.
6. *Adaptive Multi-Rate (AMR) speech transcoding*, ETSI EN 301 705 v7.2.1, 2004.
7. S.V. Andersen, W.B. Kleijn, R. Hagen, J. Linden, M.N. Murthi, and J. Skoglund, "iLBC - a linear predictive coder with robustness to packet losses," *IEEE Workshop Proceedings. on Speech Coding*, pp. 23–25, October 2002.
8. WeiQi Zhang, Liang He, Yen-Lu Chow, RongZhen Yang, and YePing Su, "The study on distributed speech recognition system," *IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 3, pp. 1431–1434, 2000.
9. *Distributed speech recognition; Front-end feature extraction algorithm; Compression algorithms*, ETSI ES 201 108, 2000.
10. *Advanced front-end feature extraction algorithm*, ETSI ES 202 050 v1.1.1, 2002.

11. *Distributed Speech Recognition; Extended Front-end Feature Extraction Algorithm; Compression Algorithm, Back-end Speech Reconstruction Algorithm*, ETSI ES 202 211, 2003.
12. *Distributed Speech Recognition; Extended Advanced Front-end Feature Extraction Algorithm; Compression Algorithm*, ETSI ES 202 212, 2005.
13. *Pulse code modulation (PCM) of voice frequencies*, ITU-T Recommendation G.711, 1988.
14. *40, 32, 24, 16 kbit/s adaptive differential pulse code modulation (ADPCM)*, ITU-T Recommendation G.726, 1990.
15. J.Trebolet and R.E.Crochiere, "Frequency domain coding of speech," *IEEE Trans.on Information Theory*, pp. 55–67, 1981.
16. B.Gold and C.Rader, "The channel vocoder," *IEEE Trans.Audio Electroacoustics*, pp. 148–160, 1967.
17. *Analog to digital conversion of radio voice by 2400 bit/second linear predictive coding*, Federal Standard 1015, National Communication System, Office of Technology and Standards, 1984.
18. P.Kroon, E.F.Deprettere, and R.J.Sluyter, "Regular-pulse excitation: A novel approach to effective and efficient multipulse coding of speech," *IEEE Trans.On Acoustics, Speech and Signal Processing*, vol. 34, pp. 1054–1063, 1986.
19. M.R.Schroeder and B.S.Atal, "Code-excited linear prediction (CELP): High quality speech at very low bit rates," *In Proceedings of ICASSP*, pp. 937–940, 1985.
20. H-G.Hirsch, "The influence of speech coding on recognition performance in telecommunication networks," *ICSLP 2002*, pp. 1877–1880, 2002.
21. J.M.Huerta and R.M.Stern, "Distortion-class modeling for robust speech recognition under GSM RPE-LTP coding," *Speech Communication*, vol. 34, 2001.
22. A. Gallardo-Antolin, F. Diaz de Maria, and F.Valverde-Albacete, "Avoiding distortion due to speech coding and transmission errors in GSM ASR tasks," *Proc. ICASSP*, vol. 1, pp. 277–280, 1999.
23. A. Gallardo-Antolin, F. Diaz de Maria, and F.Valverde-Albacete, "Recognition from GSM digital speech," *Proc. ICSLP*, 1998.
24. A.M. Gómez, A.M. Peinado, V. Sánchez, and A.J. Rubio, "Recognition of coded speech transmitted over wireless channels," *IEEE Trans. on Wireless Communications*, 2006.
25. G.N.Ramaswamy and P.S.Gopalakrishnan, "Compression of acoustic features for speech recognition in network environments," *in proc.IEEE Int.Conf.on Acoustics, Speech and Signal Processing, ICASSP*, 1998.
26. A.Bernard and A.Alwan, "Low-bitrate distributed speech recognition for packet-based and wireless communication," *IEEE Transactions on Speech and Audio Processing*, vol. 10, pp. 570–579, 2002.
27. "Strategic research agenda," *eMobility; European Mobile and Wireless Communications Technology Platform*, 2006.
28. Z. Tan, P. Dalsgaard, B. Lindberg, and H. Xu, "Robust speech recognition in ubiquitous networking and context-aware computing," *Proc. of Interspeech*, September 2005.
29. H. Xu, Z. Tan, P. Dalsgaard, and B. Lindberg, "Robust speech recognition based on noise and snr classification - a multiple-model framework," *Proc. of Interspeech*, September 2005.
30. M. Akbacak and J. Hansen, "Environmental sniffing: Noise knowledge estimation for robust speech systems," *Proc. of ICASSP*, 2003.

31. H. Xu, Z. Tan, P. Dalsgaard, and B. Lindberg, "Robust-speech recognition from noise-type based feature compensation and model interpolation in a multiple model framework," *Proc. of ICASSP*, 2006.
32. L. Ma, D. Smith, and B. Milner, "Context-awareness using environmental noise classification," *Proc. of Eurospeech*, 2003.
33. B. Milner and A. James, "Robust speech recognition over mobile and IP networks in burst-like packet loss," *IEEE Trans. on Audio, Speech and Lang. Proc.*, vol. 14, pp. 223–231, 2006.
34. A.M. Gómez, A.M. Peinado, V. Sánchez, B.P. Milner, and A.J. Rubio, "Statistical-based reconstruction methods for speech recognition in IP networks," *COST 278, Robust2004: Robustness Issues in Conversational Interaction*, 2004.
35. A.M. Peinado, A.M. Gómez, V. Sánchez, and A.J. Rubio, "Packet loss concealment based on VQ replicas and MMSE estimation applied to Distributed Speech Recognition," in *proceedings of ICASSP*, vol. 1, pp. 329–332, 2005.
36. A.M. Gómez, A.M. Peinado, V. Sánchez, and A.J. Rubio, "Combining media-specific FEC and error concealment for robust distributed speech recognition over loss-prone packet channels," *IEEE Trans. on Multimedia, In Press*, 2006.
37. A.M. Gómez, A.M. Peinado, V. Sánchez, J.L. Carmona, and A.J. Rubio, "Inter-leaving and MMSE estimation with VQ replicas for distributed speech recognition over lossy packet networks," in *proceedings of INTERSPEECH-ICSLP*, 2006.
38. A. Bernard and A. Alwan, "Joint channel decoding - viterbi recognition for wireless applications," *Proc. Eurospeech*, vol. 4, pp. 2703–2706, 2001.
39. A. Cardenal-Lopez, L. Docio-Fernandez, and C. Garcia-Mateo, "Soft decoding strategies for distributed speech recognition over IP networks," *Procs. of ICASSP04*, vol. 1, pp. 49–52, 2004.
40. A.M. Gómez, A.M. Peinado, V. Sánchez, and A.J. Rubio, "On the ramsey class of interleavers for robust speech recognition in burst-like packet loss," *IEEE Trans. on Audio Speech and Language Processing*, vol. 8, pp. 1496–1499, 2007.
41. J.M. Huerta and R.M. Stern, "Speech recognition from GSM codec parameters," *Proc. of International Conference on Spoken Language Processing (ICSLP)*, vol. 4, pp. 1463–1466, 1998.
42. C. Pelaez-Moreno, A. Gallardo-Antolin, and F. Diaz de Maria, "Recognizing voice over IP: a robust front-end for speech recognition on the world wide web," *IEEE Transactions on Multimedia*, vol. 3, 2001.
43. H.K. Kim and V. Cox, "A bitstream-based front-end for wireless speech recognition on IS-136 communications system," *IEEE Transactions on Speech and Audio Processing*, vol. 9, no. 5, 2001.
44. B. Raj, J. Migdal, and R. Singh, "Distributed speech recognition with codec parameters," *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pp. 127–130, 2001.
45. J.L. Carmona, A.M. Peinado, J.L. Pérez-Córdoba, A.M. Gómez, and V. Sánchez, "iLBC-based transparametrization: a real alternative to DSR for speech recognition over packet networks," in *proceedings of ICASSP*, 2007.
46. Q. Xie, D. Pearce, S. Balasuriya, Y. Kim, S.H. Maes, and H. Garudari, "RTP payload format for DSR ES 201 108," *IETF Audio Video Transport WG, Internet RFC 3557*, 2002.



# Hovering Information : Infrastructure-Free Self-Organising Location-Aware Information Dissemination Service

Alfredo A. Villalba Castro<sup>1</sup>, Giovanna Di Marzo Serugendo<sup>2</sup>, and  
Dimitri Konstantas<sup>1</sup>

<sup>1</sup> Centre Universitaire d'Informatique, University of Geneva,  
24 rue Général Dufour, 1211 Geneva 4, Switzerland,

{alfredo.villalba,dimitri.konstantas}@cui.unige.ch

<sup>2</sup> School of Computer Science and Information Systems,

Birkbeck, University of London,

Malet Street, London WC1E 7HX, UK,

dimarzo@dcs.bbk.ac.uk

**Abstract.** This paper proposes a location-based service for disseminating geo-localised information generated by and aimed at mobile users. The service itself works in a self-organising manner. A piece of hovering information is attached to a geographical point, called the anchor location, and to its vicinity area, called the anchor area. It is responsible for keeping itself alive, available and accessible to other devices within its anchor area. Hovering information uses mechanisms such as active hopping, replication and dissemination among mobile nodes to satisfy the above requirements. It does not rely on any central server. Previous results involving a single piece of hovering information have shown the interest of the concept. This paper reports on a series of simulations involving multiple pieces of hovering information. Our goal is to investigate the scalability of the technique up to 200 pieces in a small geographic area. Two main replication algorithms for pieces of hovering information are compared, an Attraction Point algorithm and a Broadcast-based one. These replication algorithms are combined with two different caching policies, Location-based and Generation-based, for discarding hovering information pieces from mobile nodes buffer when memory is not enough.

## 1 Introduction

The last two decades were marked by a rapid evolution of computer and communication related technologies available to the end-users. From the 300MHz processors available in the late 80s, today we are above 4GHz, and from 56Kbps networks (modems), we have today home networks of more than 20Mbps. In a similar way mobile storage capacities available to end-users have gone up from 1MB diskettes to 8GB memory sticks. The average end user today has more wireless network, processing power and storage capacity available to a mobile device, like PDA or smart-phone, than the semi-professional user of the late 80s.

It is thus safe to assume that at the end of the 2010s the average user will have a mobile device with more than 4GHz processing power, 100Mbps wireless network connectivity and more than 1TB of local storage capacity, all supported by powerful power supply, allowing him to have a continuous high bandwidth network connection for periods longer than 24 hours. We can thus also anticipate that the mobile available storage capacity of the end-users will be an important percentage of the fixed storage capacity on the planet. Furthermore we can expect that, mobile devices like phones and PDAs will be equipped with high quality geo-localisation hardware (like GPS and Galileo chips).

Besides these technological advances, we have witnessed during the last few years a new direction taken by the end-users in the creation of information. With the available communication means, end-users are changing their behaviour from information consumers to information producers. More and more information created by end-users is becoming available on the internet, as it is observed by the big success of sites like YouTube and FaceBook. We can thus anticipate that for the next decade end-users will keep on producing even more content, making it available to other users in different forms and under different means.

Considering the above predictions for the next decade, we can sketch a daily scenario for the average user of the late 2010s. The user is equipped with a mobile device through which he accesses location related information that was created by him, friends, or even strangers. All this information is stored primarily into his mobile device and may become available to other users without passing by a mobile operator or a centralized server, instead setting up a mobile ad hoc network and taking advantage of the vast amounts of mobile storage capacities available. In this direction, we have defined the concept of Hovering Information, which provides a promising solution by creating the basis and models for large-scale, location related information management. Instead of accessing location-based information via a wireless network operator, the Hovering Information concept will allow mobile devices to disseminate the information among them, thus relieving the load of the wireless networks. Applications such as stigmergy-based systems, traffic management over vehicle networks or mounting distributed information systems on disaster areas could be implemented using hovering information.

This paper presents the hovering information concept, a replication algorithm allowing single pieces of hovering information to get attracted to their respective geographical related locations, called anchor locations; a broadcast-based algorithm for comparing network and memory usage performances; and two mobile nodes caching policies, Location-based and Generation-based, for discarding pieces of hovering information when memory space is full. A complete formal description of the hovering information model is described in [14].

## 2 Hovering Information Concept

This section describes the main concepts of a hovering information system: mobile nodes, hovering information; and three main dependability requirements of hovering information: survivability, availability and accessibility.



## 2.1 Mobile Nodes and Hovering Information

Mobile nodes represent the storage and motion media exploited by pieces of hovering information. A *mobile node*  $n$  is defined as a tuple:

$$n = (id, loc, speed, dir, r_{comm}, buff),$$

where  $id$  is its mobile node identifier,  $loc$  is its current location (a geographic location),  $speed$  is its current speed in  $m/s$ ,  $dir$  is its current direction of movement (a geographic vector),  $r_{comm}$  is its wireless communication range in meters and  $buff$  is its buffer (having a limited size) aimed to store replicas of the pieces of hovering information.

A piece of hovering information is a piece of data whose main goal is to remain stored in an area centred at a specific location called the *anchor location*, and having a radius called the *anchor radius*. A *piece of hovering information*  $h$  is defined as a tuple:

$$h = (id, a, r, n, data, policies, size),$$

where  $id$  is its hovering information identifier,  $a$  is its anchor location (geographic coordinate),  $r$  is its anchor radius in meters,  $n$  is the mobile node where  $h$  is currently hosted (hosting node),  $data$  is the data carried by  $h$ ,  $policies$  are the hovering policies of  $h$  and  $size$  is the size of  $h$  in bytes. Policies stand for hovering policies stating how and when a piece of hovering information has to hover.

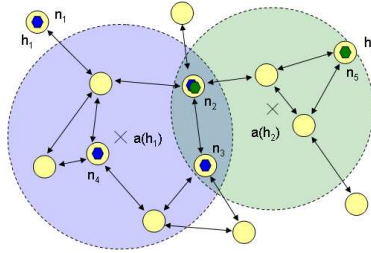
We consider that identifiers of pieces of hovering information are unique, but replicas (carrying same data and anchor information) are allowed on *different* mobile nodes. We also consider that there is only one instance of a hovering information in a given node  $n$ , any other replica resides in another node.

A hovering information system is composed of mobile nodes and pieces of hovering information. A hovering information system at time  $t$  is a snapshot (at time  $t$ ) of the status of the system. We denote by  $\mathcal{N}_t$  the set of mobile nodes at time  $t$ . Mobile nodes can change location, new mobile nodes can join the system and others can leave. New pieces of hovering information can appear (with new identifiers), replicas may appear or disappear (same identifiers but located on other nodes), hovering information may disappear or change node.

Figure 1 shows two different pieces of hovering information  $h_1$  (blue) and  $h_2$  (green), having each a different anchor location and area. Three replicas of  $h_1$  are currently located in the anchor area (in three different mobile nodes  $n_2$ ,  $n_3$  and  $n_4$ ), while two replicas of  $h_2$  are present in the anchor area of  $h_2$  (in nodes  $n_2$  and  $n_5$ ). It may happen that a mobile node hosts replicas of different pieces of hovering information, as it is the case in the figure for the mobile node  $n_2$  that is at the intersection of the two anchor areas. The arrows here also represent the communication range possibilities among the nodes.

## 2.2 Properties - Requirements

**Survivability.** A hovering information  $h$  is alive at some time  $t$  if there is at least one node hosting a replica of this information. The survivability along a period



**Fig. 1.** Hovering Information System at time  $t$

of time is defined as the ratio between the amount of time during which the hovering information has been alive and the overall duration of the observation.

**Availability.** A hovering information  $h$  is available at some time  $t$  if there is at least a node in its anchor area hosting a replica of this information. The availability of a piece of hovering information along a period of time is defined as the rate between the amount of time along which this information has been available during this period and the overall time.

**Accessibility.** A hovering information is accessible by a node  $n$  at some time  $t$  if the node is able to get this information. In other words, if it exists a node  $m$  being in the communication range of the interested node  $n$  and which contains a replica of the piece of hovering information. The accessibility of a piece of hovering information  $h$  is the rate between the area covered by the hovering information's replicas and its anchor area.

The interested reader can refer to [14] for a full set of definitions.

### 3 Algorithms for Hovering Information

In [15] we studied the performances in terms of availability of a hovering information system containing many replicas of *only one* piece of hovering information. We assumed that each node had a buffer with an unlimited amount of memory for storing replicas. Therefore, the proposed replication algorithms should not have had to cope with buffer overflows problems when a new incoming replica arrived. In this paper, we drop the assumption of unlimited memory and we study a hovering information system containing *multiple* (distinct) hovering information and their respective replicas. Instead of keeping an unlimited buffer size for storing replicas, we limit the size of the buffer. The need for caching policies becomes then important when it is time to insert a new incoming replica.

In this paper, we propose and study two different caching policies: Location-Based Caching (LBC) and Generation-Based Caching (GBC). The LBC policy decision to erase a replica is based on the proximity of that replica to its anchor location and on the portion of the surface of the anchor area covered by the communication area of the node hosting it. The GBC policy takes the decision of removing a replica based on the generation of the replica, removing replicas having the oldest generations.

**Assumptions.** We make the following assumptions in order to keep the problem simple while focusing on measuring availability and resource consumption. *Uniform size:* All pieces of hovering information have the same size and the caching algorithms do not take in consideration the size as a criteria when removing a replica. *Unlimited energy:* All mobile nodes have an unlimited amount of energy. The proposed algorithms do not consider failure of nodes or impossibility of sending messages because of low level of energy. *In-built geo-localization service:* Mobile nodes have an in-built geo-localization service such as GPS which provides the current position. We assume that this information is available to pieces of hovering information. *Neighbours discovering service:* Mobile nodes are able to get a list of their current neighbouring nodes at any time. This list contains the position, speed, and direction of the nodes. As for the other two services, this information is available to pieces of hovering information.

### 3.1 Safe, Risk and Relevant Areas

In this paper we consider that all pieces of hovering information have the same hovering policies: active replication and hovering in order to stay in the anchor area (for survivability, availability and accessibility reasons), caching when there is no free space to store a replica, and cleaning when too far from the anchor area to be meaningful (i.e. disappearance). The decision on whether to replicate itself or to hover depends on the current position of the mobile node in which the hovering information is currently stored. Therefore, we distinguish three different areas: safe area, risk area and relevant area.

A piece of hovering information located in the *safe area* can safely stay in the current mobile node, provided the conditions on the node permit this: power, memory, etc. This area is defined as the disc having as centre the anchor location and as radius the safe radius ( $r_{safe}$ ).

A piece of hovering information located in the *risk area* should actively seek a new location on a mobile node going into the direction of the safe area. It is in this area that the hovering information actively replicates itself in order to survive and stay available in the vicinity of the anchor location. This area is defined as the ring having as centre the anchor location and bound by the safe and risk radii ( $r_{risk}$ ).

The *relevant area* limits the scope of survivability of a piece of hovering information. This area is defined as the disc whose centre is the anchor location and whose radius is the relevant radius ( $r_{rele}$ ). The *irrelevant area* is all the area outside the relevant area. A piece of hovering information located in the irrelevant area can disappear; it is relieved from survivability goals.

All these radii cope with the following inequality (where  $r$  is the anchor radius):

$$r_{safe} < r < r_{risk} < r_{rele}$$

The values of these different radii are different for each piece of hovering information and are typically stored in the Policies field of the hovering information. In the following algorithms we consider that all pieces of hovering information have the same relevant, risk and safe radius.

### 3.2 Replication

A piece of hovering information  $h$  has to replicate itself onto other nodes in order to stay alive, available and accessible. We describe two replication algorithms simulating two variants of the replication policies: the Attractor Point and Broadcast-based algorithms. Both algorithms are triggered periodically - each  $T_R$  (replication time) seconds - and only replicas of  $h$  being in the risk area are replicated onto some neighbouring nodes (nodes in communication range) which are selected according to the replication algorithm.

When replicas consider themselves too far from their anchor area and not able to come back anymore, the cleaning mechanism periodically - each  $T_C$  (cleaning time) seconds - and for each node, removes the replicas that are too far from their anchor location, i.e. those replicas that are in the irrelevant area. This avoids as well the situation where all nodes have a replica.

**Attractor Point Algorithm (AP)** The anchor location of a piece of hovering information acts constantly as an attractor point to that piece of hovering information and to all its replicas. Replicas tend to stay as close as possible to their anchor area by jumping from one mobile node to other. The number of target nodes composing the multicast group that will receive a replica is defined by the constant  $k_R$  (replication factor).

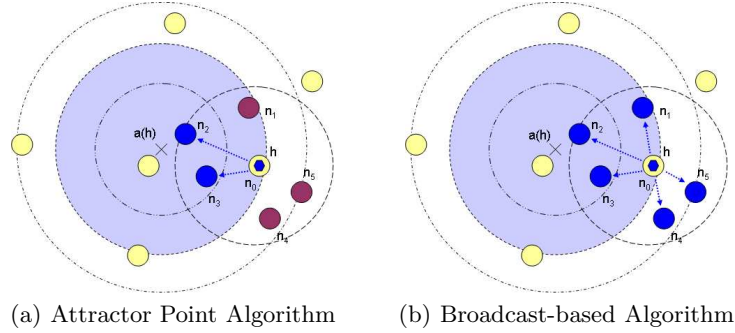
Figure 2(a) illustrates the behaviour of the Attractor Point algorithm. Consider a piece of hovering information  $h$  in the risk area. It replicates itself onto the nodes in communication range that are the closest to its anchor location. For a replication factor  $k_R = 2$ , nodes  $n_2$  and  $n_3$  receive a replica, while all the other nodes in range do not receive any replica.

**Broadcast-based Algorithm (BB)** The Broadcast-based algorithm is triggered periodically (each  $T_R$ ) for each mobile node. After checking the position of the mobile node pieces of hovering information located in the risk area are replicated and broadcasted onto *all* the nodes in communication range. We expect this algorithm to have the best performance in terms of availability but the worst in terms of network and memory resource consumption.

Figure 2(b) illustrates the behaviour of the Broadcast-based algorithm. Consider the piece of hovering information  $h$  in the risk area, it replicates itself onto all the nodes in communication range, nodes  $n_1$  to  $n_5$  (blue nodes).

### 3.3 Caching

In this paper we assume that nodes have a limited amount of memory to store the pieces of hovering information (replicas). As the number of distinct hovering information increases, so will be the total number of replicas. The buffer of nodes will get full at some point and some replicas should have to be removed in order to store new ones. We present two different caching policies: Location-Based and Generation-Based Caching. We compare these caching techniques with a simpler



**Fig. 2.** Replication Algorithms

one which only ignores the incoming replicas as soon as there is no free space in the mobile device buffer.

Besides these caching algorithms, it is important to notice that we only consider the position and the generation of replicas. We do not take into consideration caching policies such as the priority, the time-to-live or the replicas size (since all replicas considered in this paper have the same size).

**Location-Based Caching (LBC)** At each node, this caching policy decides to remove a previously stored replica from the node’s full buffer, and to replace it by the new incoming replica based on their respective location relevance value. We define the location relevance value of a replica, being this replica already stored in the node’s buffer or being a new incoming replica, to its anchor location and area as it follows:

$$relevance = \alpha * area + \beta * proximity,$$

where *area* is the normalised estimation of the overlapping area of the nodes’ communication range area and the replica’s anchor area, *proximity* is the normalised proximity value between the current position of the node and the anchor location of the replica,  $\alpha$  and  $\beta$  are real coefficients having values between 0 and 1 and  $\alpha + \beta = 1$ .

Each time a new incoming replica arrives, the least location relevant replica is chosen from all the replicas stored in buffer of the node. The location relevance of the incoming replica is computed and compared to that of the least location relevant replica, whatever the original hovering information they refer to. The least location relevant replicas is removed from the buffer and replaced by the incoming replica if the latter has a greater location relevance value. Otherwise, the incoming replica is just discarded. In this way, the location-based caching algorithm will tend to remove replicas being too far from their anchor location or being hosted in a node covering only a small part of their anchor area.

**Generation-Based Caching (GBC)** We define the generation of a replica in the following way: the first replica created (normally by the user or user application) of a piece of hovering information has a generation 0, when this replica replicates itself then it creates new replicas having generation 1, and so on. The generation of a replica gives us an idea of the number of replicas existing as the process of replication follows an exponential growth. The generation-based caching algorithm tends to remove replicas having a high generation number as they are likely more replicas leaving around than a replica having a lower generation number.

Each time a new incoming replica arrives, the oldest replica (the one having the highest generation value) is chosen from all the replicas stored in the buffer of the node. The generation of the incoming replica is retrieved and compared to that of the oldest replica, whatever the original hovering information they refer to. The oldest replica is removed from the buffer and replaced by the incoming replica if the latter has a smaller generation value. Otherwise, the incoming replica is just discarded.

## 4 Evaluation

We evaluated the behaviour of the above described replication and caching algorithms under different scenarios by varying the number of pieces of hovering information, each having many replicas of its. We also considered nodes having a limited buffer size to store the different replicas. We have measured the average availability, message complexity, replication complexity, overflows and erased replicas over the total number of pieces of hovering information existing in the system.

We performed simulations using the OMNet++ network simulator (distribution 3.3) and its Mobility Framework 2.0p2 (mobility module) to simulate nodes having a simplified WiFi-enabled communication interfaces (not dealing with channel interferences) with a communication range of 121m.

### 4.1 Simulation Settings and Scenarios

The generic scenario consists of a surface of 500m x 500m with mobile nodes moving around following a Random Way Point mobility model with a speed varying from 1m/s to 10m/s without pause time. In this kind of mobility model, a node moves along a straight line with speed and direction changing randomly at some random time intervals.

In the generic scenario, pieces of hovering information have an anchor radius ( $r$ ) of 50m, a safe radius ( $r_{safe}$ ) of 30m, a risk radius ( $r_{risk}$ ) of 70m, a relevance radius ( $r_{rele}$ ) of 200m, and a replication factor of 4 ( $k_R$ ).

Each node triggers the replication algorithm every 10 seconds ( $T_R$ ) and the cleaning algorithm every 60 seconds ( $T_C$ ). Each node has a buffer having a capacity to store 20 different replicas. The caching algorithm is constantly listening for the arrival of new replicas.

Based on this generic scenario, we defined 5 specific scenarios with varying number of pieces of hovering information: from 40 to 200 nodes, increasing the number of pieces by 40. Each of this scenarios has been investigated with different replication and caching algorithms, and with a different number of nodes.

We have performed 20 runs for each of the above scenarios. One run lasts 3'600 simulated seconds. All the results presented here are the average of the 20 runs for each scenario, and the errors bars represent a 95% confidence interval. All the simulations ran on a Linux cluster of 32 computation nodes (Sun V60x dual Intel Xeon 2.8GHz, 2GB RAM).

## 4.2 Results

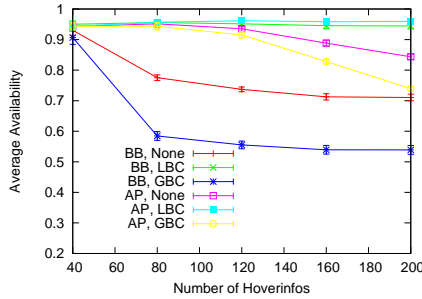


Fig. 3. Availability - 200 Nodes

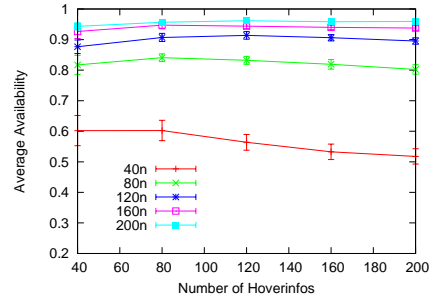


Fig. 4. Availability - AP with LBC

Figure 3 shows the average availability for the AP and BB algorithms using LBC or GBC caching policies, or without using any (None). For this experiment we used 200 nodes. We observe that both algorithms using LBC outperform the cases using GBC or no caching policy (None). We can also notice that the BB algorithm gets worse availability performances compared to AP. This is explained by the nature of the BB algorithm which tends to overload the system with an exponential growing number of replicas. As the buffer size at each node has a limited size of 20 replicas, the overloading causes the buffer resources to become badly shared by the pieces of hovering information, in particular regarding their individual target anchor location. Consequently the availability becomes lower. On the other hand, the AP algorithm controls the replication process by limiting the number of replicas and by focusing on the anchor location. When it is combined with the LBC caching policy, the system becomes scalable keeping high availability rates (around 95%) as the number of pieces of hovering information increases.

Figure 4 depicts the average availability of the AP replication algorithm using the LBC caching policy, under several number of nodes. For a number of nodes above 120, we notice that the availability is high enough (above 85%) and it keeps quite stable as the number of pieces of hovering information increases.

We confirm from this that the AP with LBC algorithms are scalable in terms of absorption of hovering information (number of distinct pieces of hovering information), since during the experiments with 120 nodes and more, up to 200 distinct hovering information pieces have been accommodated into the system with an availability above 85%. In the case when the number of nodes is 40, we can observe that the absorption limit, for this configuration, has been reached as the availability starts decreasing after 80 pieces of hovering information.

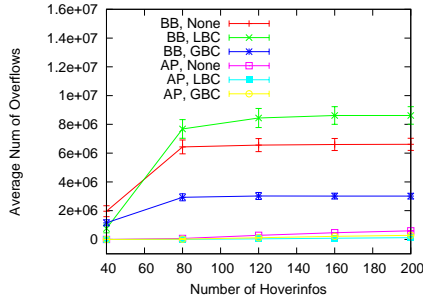


Fig. 5. Overflows - 200 Nodes

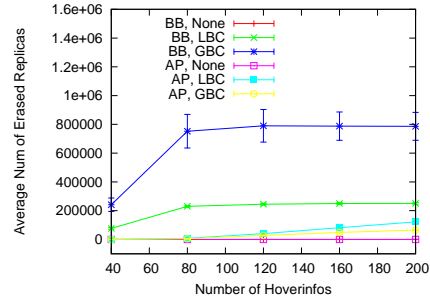


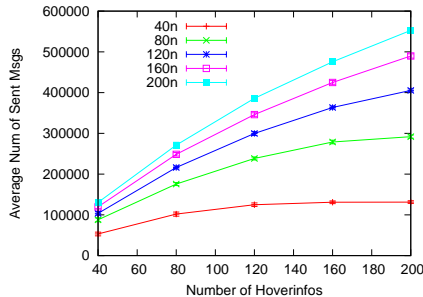
Fig. 6. Erased Replicas - 200 Nodes

Figure 5 shows the average number of overflows generated by the different replication and caching algorithms each time a new incoming replica arrives and there is no free space to store it. We observe that the AP algorithm produces at least ten times less overflows than the BB algorithm. We also observe that the number of overflows of the BB algorithm starts growing fast and then it gets stable after for 80 pieces of hovering information or more, this is again a consequence of the overloading of the system which tends to lose pieces of hovering information by not distributing the buffer resources in a fair way.

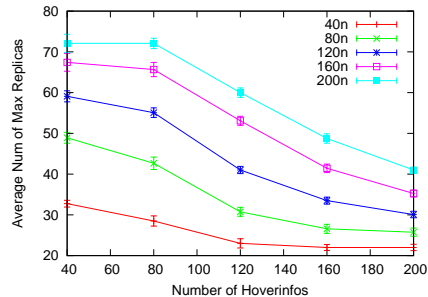
Figure 6 illustrates the average number of erased replicas from the buffer of the nodes after an overflow. Again, the BB erases around 10 times much more replicas than the AP algorithm to store a new incoming replica. We also notice that the BB with GBC tends to erase too many replicas compared to the other ones; this means that the generation-based caching policy combined with the exponential replication behaviour of BB is not a good differentiation factor for caching replicas since this combination of algorithms tends to insert and erase replicas permanently.

Figure 7 shows the average number of sent messages for the AP algorithm using the LBC policy under several numbers of nodes. We notice that the message complexity does not grow exponentially. Instead, it grows linearly or even logarithmically with the number of pieces of hovering information. This is a very important issue as the feasibility of these algorithms depends strongly on the messages complexity, especially when we will need to deal with the interference channel for even more realistic network interfaces.





**Fig. 7.** Messages Complexity - Attractor Point with Location-Based Caching



**Fig. 8.** Replication Complexity - Attractor Point with Location-Based Caching

Finally, figure 8 shows the average of the maximal number of replicas having existed in the system for the AP using the LBC. It is interesting to see that it decreases as the number of pieces of hovering information increases. It means that the buffer resources are evenly shared among the different pieces of hovering information, while the availability still remains at high levels (see Figure 4). We conclude from this, that the AP with LBC succeeds to distribute the network resource in a fair way among all the pieces of hovering information, and that we probably observe an emergent (not coded) load-balancing of the memory allocated to the different pieces of hovering information.

## 5 Related Works

To our knowledge, while there exist other information annotation/dissemination works closely related to the hovering information concept, they do not take the approach of offering a generic infrastructure-free location-aware information dissemination service as hovering information does. The Hovering Data Clouds (HDC) concept [16,8], which is part of the AutoNomos project, is applied to the specific design of a distributed infrastructure-free car traffic congestion information system. Although HDCs are defined as information entities having properties similar to hovering information, the described algorithms do not consider them as an independent service but as part of the traffic congestion algorithms. The hovering information dissemination service is thought as a service independent from the applications using it. The Ad-Loc system [1] is an infrastructure-free location-aware annotation system and shares similarities to hovering information. However, this approach does not focus on: studying properties such as the critical number of nodes or the absorption limits; or dealing with self-organizing algorithms allowing the information to adapt its behaviour according to the network saturation, the buffers' size, the mobility pattern of nodes or the number of replicas.

The opportunistic spatio-temporal dissemination service over MANETs [11] is a car traffic centred application and does not encompass ideas such as re-

combination of information. Similarly, works like Epcast [13] and Gossip [3] aim to disseminate information based on epidemic and gossiping spreading models. These works provide an interesting starting point for replication algorithms, but do not offer a solution for ensuring the persistency of the information.

In the domain of location-driven routing over MANETs, we can mention works such as GeoOpss [10], search and query propagation over social networks like PeopleNet [12] and collaborative services such as collaborative backup of the MoSAIC project [9, 2].

Finally, the virtual infrastructure project [4–7] aims to set up a set of virtual nodes having a well-know structure and trajectory over a mobile ad hoc network. These virtual nodes are equipped with a clocked automaton machine which will permit to implement distributed algorithms such as leader election, routing, atomic memory, motion coordination, etc. This approach works on offering a structured abstraction layer of virtual nodes. Hovering information takes a different approach where each piece of hovering information is an autonomous entity responsible for its own survivability exploiting the dynamics of the overlay network to this aim.

## 6 Conclusion

In this paper we discussed the notion of hovering information, we defined and simulated the Attractor Point algorithm which intends to keep the information alive and available in its anchor area. This algorithm multicasts hovering information replicas to the nodes that are closer to the anchor location. The performances of this algorithm have been compared to those of a broadcast version.

We have also defined and simulated two different caching polices, the Location-Based Caching and the Generation-Based Caching. Their performances have been compared under a scenario containing multiple pieces of hovering information and nodes having a limited amount of memory.

Results show that the Attractor Point algorithm with the Location-Based Caching policy is scalable in terms of the number of pieces of hovering information that the system can support (absorption limits). They also show the emergence of a load-balancing property of the buffer usage which stores replicas in an optimal way as the number of pieces of hovering information increases.

Concerning future work, we have tested the algorithms under a Random Way Point mobility model and under ideal wireless conditions. This is not characteristic of real world behaviour. We will apply the different algorithms to scenarios following real mobility patterns (e.g. crowd mobility patterns in a shopping mall or traffic mobility patterns in a city) with real wireless conditions (e.g. channel interferences or physical obstacles).

## References

1. D. J. Corbet and D. Cutting. Ad loc: Location-based infrastructure-free annotation. In *ICMU 2006*, London, England, Oct. 2006.

2. L. Courtès, M.-O. Killijian, D. Powell, and M. Roy. Sauvegarde coopérative entre pairs pour dispositifs mobiles. In *UbiMob '05: Proceedings of the 2nd French-speaking conference on Mobility and ubiquity computing*, pages 97–104, New York, NY, USA, 2005. ACM Press.
3. A. Datta, S. Quarteroni, and K. Aberer. Autonomous gossiping: A self-organizing epidemic algorithm for selective information dissemination in mobile ad-hoc networks. In *IC-SNW'04, International Conference on Semantics of a Networked World*, LNCS, pages 126–143, 2004.
4. S. Dolev, S. Gilbert, L. Lahiani, N. A. Lynch, and T. Nolte. Timed virtual stationary automata for mobile networks. In *OPODIS*, pages 130–145, 2005.
5. S. Dolev, S. Gilbert, N. A. Lynch, E. Schiller, A. A. Shvartsman, and J. L. Welch. Virtual mobile nodes for mobile ad hoc networks. In *DISC*, 2004.
6. S. Dolev, S. Gilbert, N. A. Lynch, A. A. Shvartsman, and J. Welch. Geoquorums: Implementing atomic memory in mobile ad hoc networks. In *DISC*, 2003.
7. S. Dolev, S. Gilbert, E. Schiller, A. A. Shvartsman, and J. Welch. Autonomous virtual mobile nodes. In *DIALM-POMC '05: Proceedings of the 2005 joint workshop on Foundations of mobile computing*, pages 62–69, New York, NY, USA, 2005. ACM Press.
8. S. P. Fekete, C. Schmidt, A. Wegener, and S. Fischer. Hovering data clouds for recognizing traffic jams. In *Proceedings 2nd International Symposium on Leveraging Applications of Formal Methods, Verification and Validation (IEEE-ISOLA)*, pages 213–218, 2006.
9. M.-O. Killijian, D. Powell, M. Banâtre, P. Couderc, and Y. Roudier. Collaborative backup for dependable mobile applications. In *MPAC '04: Proceedings of the 2nd workshop on Middleware for pervasive and ad-hoc computing*, pages 146–149, New York, NY, USA, 2004. ACM Press.
10. I. Leontiadis and C. Mascolo. Geopps: Opportunistic geographical routing for vehicular networks. In *Proceedings of the IEEE Workshop on Autonomic and Opportunistic Communications. (Colocated with WOWMOM07)*, Helsinki, Finland, June 2007. IEEE Press.
11. I. Leontiadis and C. Mascolo. Opportunistic spatio-temporal dissemination system for vehicular networks. In *MobiOpp '07: Proceedings of the 1st international MobiSys workshop on Mobile opportunistic networking*, pages 39–46, New York, NY, USA, 2007. ACM Press.
12. M. Motani, V. Srinivasan, and P. S. Nuggehalli. Peoplenet: engineering a wireless virtual social network. In *MobiCom '05: Proceedings of the 11th annual international conference on Mobile computing and networking*, pages 243–257, New York, NY, USA, 2005. ACM Press.
13. S. Scellato, C. Mascolo, M. Musolesi, and V. Latora. Epcast: Controlled dissemination in human-based wireless networks by means of epidemic spreading models. *CoRR*, abs/0711.2780, 2007.
14. A. Villalba Castro, G. Di Marzo Serugendo, and D. Konstantas. Hovering information - self-organising information that finds its own storage. Technical Report BBKCS-07-07, School of Computer Science and Information Systems, Birkbeck, University of London, Nov 2007.
15. A. Villalba Castro, G. Di Marzo Serugendo, and D. Konstantas. Hovering information - self-organising information that finds its own storage. In *IEEE International Conference on Sensors, Ubiquitous and Trust Computing (SUTC'08)*, 2008.
16. A. Wegener, E. M. Schiller, H. Hellbrück, S. P. Fekete, and S. Fischer. Hovering data clouds: A decentralized and self-organizing information system. In *International Workshop on Self-Organizing Systems*, pages 243 – 247, 2006.



# Reducing Handover Latency in Future IP-based Wireless Networks: Fast Proxy Mobile IPv6

Geert Heijenk<sup>1</sup>, Mortaza S. Bargh<sup>2</sup>, Julien Laganier<sup>3</sup>, and Anand R. Prasad<sup>3</sup>

<sup>1</sup> University of Twente, The Netherlands  
geert.heijenk@utwente.nl

<sup>2</sup> Telematica Instituut, The Netherlands  
mortaza.bargh@telin.nl

<sup>3</sup> DoCoMo Communications Labs Europe GmbH, Germany  
laganier@docomolab-euro.com

**Abstract.** Current IP-level mobility protocols have difficulties meeting the stringent handover delay requirements of future wireless networks. At the same time they do not give sufficient control to the network to control the handover process. This paper presents an extension to Proxy Mobile IP, which is the favorite IP level mobility protocol for the 3GPP System Architecture Evolution / Long Term Evolution (SAE/LTE). The extension, Fast Proxy Mobile IPv6 (FPMIPv6), aims at solving or reducing the control and handover problem. An elementary analysis shows that FPMIPv6 can significantly reduce the handover latency and the loss of packets during handover, especially if handovers can be predicted a few tens of milliseconds before they occur.

**Keywords:** Mobile IP, Handover

## 1 Introduction

Future wireless networks are assumed to fully integrate different wireless access technologies, in order to enable their users to exploit the advantages of the various technologies, depending on the momentary requirements and circumstances. Such access technologies are e.g., the 3GPP Long Term Evolution (LTE) [1] for wide area coverage at moderate data rates and (future) members of the IEEE 802.11 family of Wireless Local Area Networks (WLANs) [2] for high data rates at hotspots. To integrate the various access technologies, future wireless networks are assumed to be fully Internet Protocol (IP) -based. For instance, 3GPP is currently standardizing the System Architecture Evolution (SAE) [3], which will be IP-based. To enable the users of these future wireless networks to freely roam between various access networks, an IP-level mobility protocol is needed.

One of the problems with current IP-level mobility protocols is that they do sometimes not meet the stringent requirements for the handover latency. Ref. [4] states that in order to support real-time services such as Voice over IP (VoIP), "... the packet

transmission delay fluctuations are desired to be 30 ms or less, and interruption caused by packet losses should be 100 ms or less.” For handovers between different radio technologies, the first requirement is stretched to 50 ms.

An important aspect when designing mobility mechanism is the location of control. Traditional Mobile-IP-based approaches tend to locate the control of the handover in the mobile node, whereas current cellular networks use so-called mobile-assisted handovers, where the control of the handover is in the network, and the mobile node provides additional (measurement) information. Network controlled handovers (also when using (Mobile) IP-based protocols), enable the network operator to control the service and its quality provided to its customers, and to optimally utilize network resources. Therefore, a promising solution direction for handovers in future wireless networks is to give control of the handover process to the network. In this way the network can coordinate the entire handover process, making the most optimal decisions at the most optimal time, by using all information available. Thus, the network can improve the link quality for users, manage its radio resources, reduce interference, and enable fast and seamless handovers.

A promising proposal for an IP-level mobility protocol is Proxy Mobile IPv6 (PMIPv6) [5]. In PMIPv6, IP-level mobility is hidden from the mobile node. A special access router, called Mobile Access Gateway (MAG), enables a Mobile Node (MN) to continue to use its home address when attached. It is the MAG that detects and signals movement of the MN to the Local Mobility Anchor (LMA), which is the topological anchor point of the mobile node’s address. Typically, the handover latency of PMIPv6 equals the time needed to perform the handover at layer 2 (including authentication) plus a round-trip time between MAG and LMA. Packets sent during this period are typically lost.

In this paper we try to solve the control and the handover delay issue for PMIPv6. Our contribution extends PMIPv6 with ideas from Fast Handovers for Mobile IPv6 (FMIPv6) [6]. It will be referred to from now on as Fast Proxy Mobile IPv6 (FPMIPv6). Our extension to PMIPv6 aims to enable the network to control handovers with assistance from the mobile, while at the same time significantly reducing handover delays.

The outline of the paper is as follows. Section 2 describes related work. In Section 3, we describe the proposed extension to PMIPv6. The performance of the extension, FPMIPv6, is analyzed in Section 4. Section 5 presents conclusions and future work.

## 2 Related Work

The main problem with node mobility in IP networks is that the IP address functions both as the identifier as well as the locator of a MN in the routing hierarchy. In its role as identifier, the IP address has to be fixed in order to allow the MN to be identified by Correspondent Nodes (CNs) that want to initiate communication with it, and also because it is used by MNs to identify their ongoing Transmission Control Protocol (TCP) connections. In its role as locator, the IP address of a MN has to change when it moves to a different sub-network, in order not to disrupt the route aggregation in the Internet. IP-level mobility solutions address this problem by assigning a new IP ad-

dress to the MN, called Care-of Address (CoA). The CoA locates the MN, or the sub-network it is attached to, in the network. Typically tunneling is used to deliver packets carrying the identifier (e.g. home address) to the CoA, and hence to the MN.

In Mobile IPv6 (MIPv6) [7] a Home Agent (HA), located on the home (sub)network of the MN, tunnels packets addressed to the home address (i.e., identifier) of the MN to the CoA (i.e., current locator) of the MN. Note that in its role as an identifier for the MN, the home address also acts as a locator for the MN's sub-network (i.e. the MN's HA, or the MN itself when being at home). Additionally, CNs may be informed of the current CoA of the MN in order to enable route optimization, which allows packets to be exchanged directly between a MN and its CN without routing via the HA.

An extension to MIPv6 is Hierarchical Mobile IPv6 (HMIPv6) [8], which adds an indirection for locating a MN. The HA tunnels packets to a Mobility Anchor Point (MAP), which is addressed by a regional CoA. The MAP in turn tunnels these packets to the MN, addressed by a local CoA. The main advantage of this approach is that local handovers of the MN only have to be signaled to the MAP thus avoiding high latencies and overhead for the so-called local binding updates.

Another extension to MIPv6 is Fast Handovers for MIPv6 (FMIPv6) [6], which aims at reducing handover latencies by proactively executing the configuration of the MNs interface for the link to the target Access Router (AR) while the MN is still connected to the link to the serving AR. Further, FMIPv6 exploits forwarding of packets by the serving AR to the target AR during the critical phase of the handover and buffering of these packets at the target AR until the MN attachment is completed.

Recently, Proxy MIPv6 (PMIPv6) [5] has been proposed as a Network-based Localized Mobility Management (NetLMM) protocol to hide IP layer mobility from the MN. A special AR, called MAG, enables a MN to continue to use its home address when attached. The MAG emulates the MN's home link on the access link by advertising the MN's home network prefix to the MN. Upon handover, it is the new MAG that signals an LMA (being similar to a HA) regarding the MN movement. Packets are tunneled from the LMA to the MAG, using a proxy CoA.

In PMIPv6, which is the favorite IP level mobility protocol for SAE/LTE, control is loosely coordinated. There is limited coordination in the LMA, which replaces existing bindings with new ones, based on time-stamps provided in the proxy binding updates. In our extension, we propose to let the old MAG play a key role in coordinating the handover. In the proposal, the old MAG (or current MAG) will collect information regarding the current radio link (from MN and current Access Point (AP)), resource use (from local APs and other MAGs), and regarding potential new radio links (from MN, local APs, and other MAGs). Based on the collected info, it will trigger the handover. It does so both on the network side, by instructing the MAG aimed at to send a proxy binding update to the LMA, and on the radio side, by instructing the MN to connect to a new AP.

Recently, initial proposals have been issued to the IETF, to extend PMIPv6 with communication between old MAG and new MAG [9, 10]. However, as opposed to this paper, the proposals do not deal with handovers that are predicted wrongly or too late, and neither do [9, 10] present any analysis of the proposals.

### 3 Fast Proxy Mobile IPv6

In our description of FPMIPv6, we use the same terminology as used in PMIPv6 (see Fig. 1). Nodes in the FPMIPv6 domain that are involved in the handover process are the Mobile Node (MN), the current or old and the new Access Points (APold and APnew respectively), which provide link level wireless access to the MN, the old and the new Mobile Access Gateway (MAGold and MAGnew respectively), which are the first IP-level node as seen from the MN, and the Local Mobility Anchor (LMA), where the IP address that the MN uses resides.

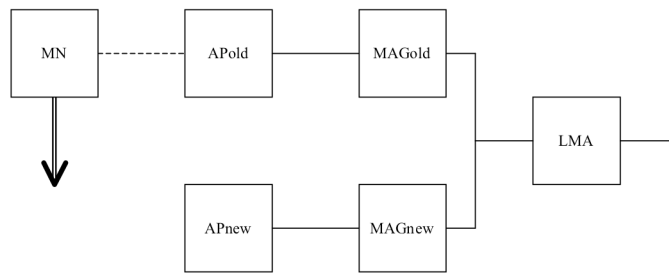


Fig. 1. Nodes in the FPMIPv6 domain.

#### 3.1 Protocol overview

The aim of FPMIPv6 is to (1) minimize the handover latency at the network layer and (2) minimize loss of packets due to the handover latency. The proposal is based on two main ideas:

- 1 A handover can be predicted.  
As a result of the prediction, the new MAG can be triggered to send a proxy binding update to the LMA, so that downlink traffic from LMA to MN can already be switched to the new MAG. The new MAG will buffer the traffic until the MN is connected.
- 2 Downlink traffic arriving at the old MAG after a MN disconnects will be buffered and redirected to the new MAG.  
To instruct the old MAG to redirect traffic, the new MAG will inform the old MAG of the handover by means of a Fast Binding Update (FBU) message.

The main coordination point of the handover is the old MAG. Based on information from MN, old AP (e.g., to predict that a handover will be needed), other APs, and other MAGs, it will take the decision to initiate a handover. The targeted new MAG is instructed to send a proxy binding update message to the LMA. As soon as this binding update is acknowledged, the new MAG will inform the old MAG. The old MAG will now instruct the AP and/or MN to release the radio link. Depending on the wireless interface capabilities of the MN, the MN will establish a radio link to the new AP before or after releasing the old one. A MN with multiple wireless interfaces can for instance already establish a new radio link as soon as the MAG initiates the handover.



To deal with mismatches in timing, the new MAG will buffer downlink packets if the MN has not yet attached to its AP. Similarly, the old MAG will buffer downlink packets if the MN has already disconnected and it still receives packets from the LMA. The new MAG will instruct the old MAG to forward buffered packets by means of a fast binding update message.

A critical issue in the operation of FPMIPv6 is the timely and correct prediction of a handover. In [11], simulation experiments using a simple handover prediction technique for 3GPP LTE indicate that it is indeed possible to predict handovers. In that paper, a timely prediction of approximately 70% of the handovers is obtained. Only a few percent of the predicted handovers had to be reversed, because the prediction was not accurate.

In the remainder of this section, we first present the basic operation of the protocol, in Section 3.2, followed by the case where the MN has two interfaces, in Section 3.3. We discuss the case that the prediction is not timely, i.e., the handover is premature, in Section 3.4. Finally, in Section 3.5, we discuss the case that the prediction is not correct.

### 3.2 Basic Operation

The basic operation of the FPMIPv6 proposal is illustrated in the Time Sequence Diagram (TSD) depicted in Fig. 2. Ideally, the old MAG receives a timely notification from L2 (from the old AP and indirectly the MN) that a handover is imminent, including the expected new Access Point to which the MN will connect (HOInfo). From that information the old MAG can easily derive the address of the new MAG. How exactly the address is derived is outside the scope of this paper. Each MAG can be provided with a table relating the APs of its neighboring MAGs to the relevant MAG address. Alternatively, the address of the MAG can be broadcasted by the connected APs on their pilot channel, so that the MN can inform the old MAG (HOInfo).

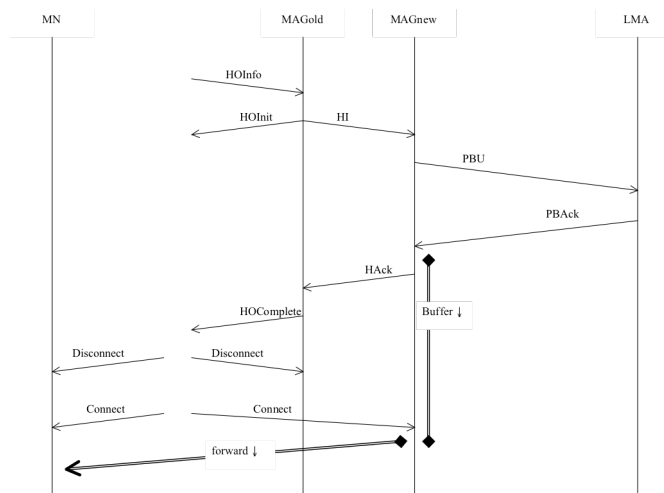


Fig. 2. Basic operation of FPMIPv6

If the old MAG decides to initiate a handover, it will inform the old AP (L2 message HOInit). Further, the old MAG will send a Handover Initiation (HI) message to the new MAG. This HI message includes a.o. a MN identifier and a timestamp.

Upon receiving the HI message, the new MAG will send a Proxy Binding Update (PBU) message to the LMA, using the timestamp received in the HI message. The LMA will install the new binding, return a Proxy Binding Acknowledgement (PBAck) message to the new MAG, and from now on forward all traffic for the MN to the new MAG.

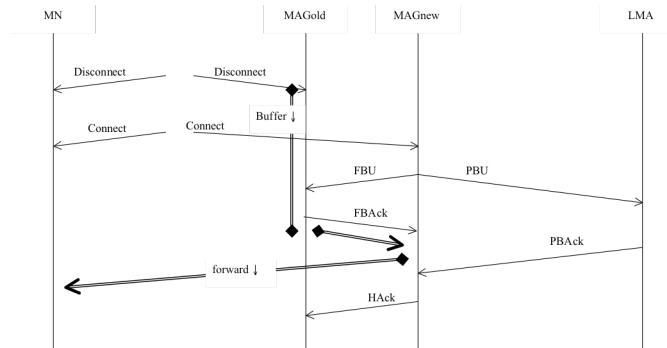
Upon receipt of the PBAck message, the new MAG will send a Handover Acknowledgement (HAck) message to the old MAG and start buffering traffic for the mobile node, until the MN connects to the new MAG (via the new AP), after which it will forward all traffic to the MN. The old MAG, upon receipt of the HAck message will send a L2 HOComplete message to the old AP, to instruct the MN to disconnect from the old AP, and connect to the new one, if it has not already done so. As we will see in Section D, some messages that are sent after the MN connects to the new MAG are omitted from the TSD in Fig. 2 in order to concentrate the description on the basic operation.

### 3.3 Operation with two interfaces

Depending on its implementation, a MN may be able to use two interfaces simultaneously. This may be because it uses two separate physical interfaces, e.g., to get access using different wireless technologies. The MN could also be using multiple virtual interfaces mapped onto a single physical interface [12]. Being able to connect to a new AP whereas the link to the old one will be maintained can significantly improve the handover performance. It also introduces new challenges, as the *network* should be able to (1) distinguish the interfaces, and (2) realize that the interfaces belong to the same node. We choose to introduce a shim layer in the MN that hides the presence of multiple interfaces from the IP layer. The MN only has a single globally routable IP address, and the shim layer should ensure that the MN is only using it on one interface at a time. Challenge (1) above is met because the different interfaces will use different hardware addresses. To meet challenge (2), the MN initially uses the MN identifier, and later the globally routable IP address. In case of multiple interfaces, the MN will use the HOInit that it gets forwarded from the AP to instruct its second interface to connect to the new AP. This way, the MN will already have a link to the new MAG available, at the moment it gets disconnected from the old MAG.

### 3.4 Premature Handover

Let us return to the single interface scenario. A handover cannot always be timely predicted. In order to avoid packet loss during an unexpected handover, the following reactive handover operation is proposed, illustrated by the TSD in Fig. 3.

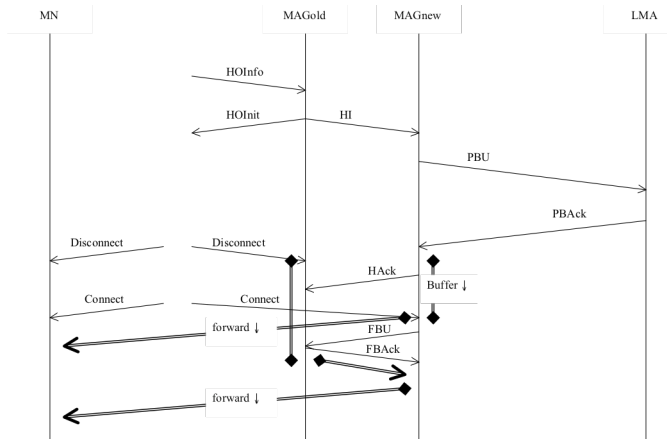


**Fig. 3.** Reactive handover operation

If the old MAG learns from L2 (APold) that the connection to the MN is lost, it starts buffering downlink packets that it receives for the MN. As soon as the new MAG learns from its L2 (APnew) that the MN has connected, it will send a PBU message to the LMA, and a Fast Binding Update (FBU) message to the old MAG. From this moment on, the new MAG will forward all downlink traffic it receives for the MN to the MN. The LMA will install the new binding, return a PBAck message to the new MAG, and from now on forward all traffic for the MN to the new MAG.

The old MAG upon receipt of the FBU message, acknowledges the message with a Fast Binding Acknowledge (FBAck) message, and redirects the traffic for the MN that it has buffered, or still receives to the new MAG.

The TSD depicted in Fig. 4 illustrates the case where the handover is predicted, but the MN is disconnected prematurely. The behavior of the nodes is the same as described above.



**Fig. 4.** Late prediction operation

### 3.5 Wrongly predicted handover

It can occur that a handover is predicted, but later on it appears that the MN has connected to a different AP and a different MAG. This requires that the old MAG corrects its HI message to the wrong MAG.

If the old MAG receives a FBU message for a MN from a MAG other than the new MAG where it has sent a HI message to, it will send the latter node also a FBU, so that it will start redirecting traffic for the MN back to the old MAG. The old MAG will redirect all traffic for the MN to the MAG it received the FBU from.

If a new MAG has initiated the handover, but before the MN connects, it receives an FBU message for that MN, it will start redirecting traffic for the MN to the sender of the FBU.

It can also occur that the old MAG has initiated a handover, but the MN is not able to perform the handover, and reconnects with the old AP / MAG. In that case, the old MAG will send a PBU message to the LMA, and also send an FBU message to the MAG the HI was sent to. In this situation, the old MAG carries out functions of a new MAG for the same MN simultaneously.

The detailed operation of the FPMIPv6 protocol is described in the finite state machines depicted in Fig. 5 and Fig. 6 for the old MAG and the new MAG respectively.

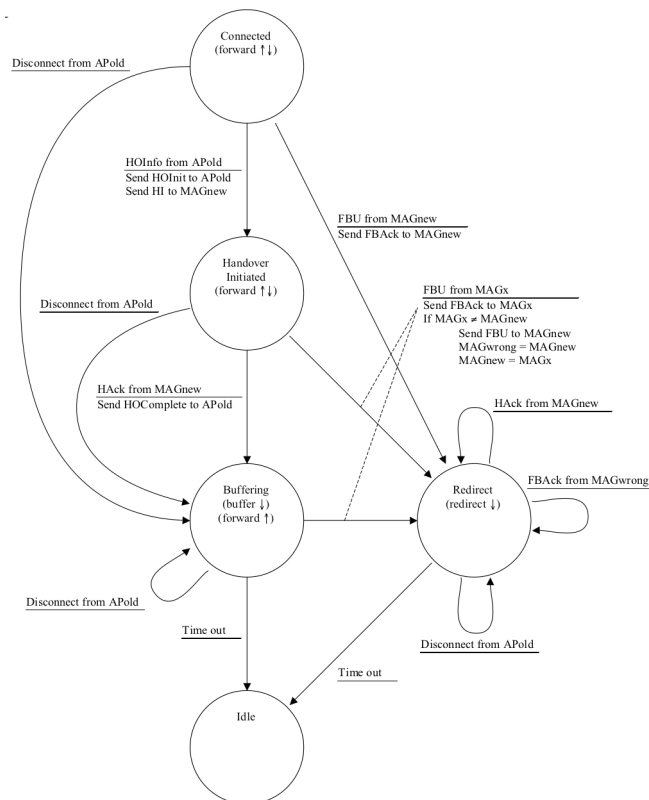


Fig. 5. Finite state machine of MAGold operation

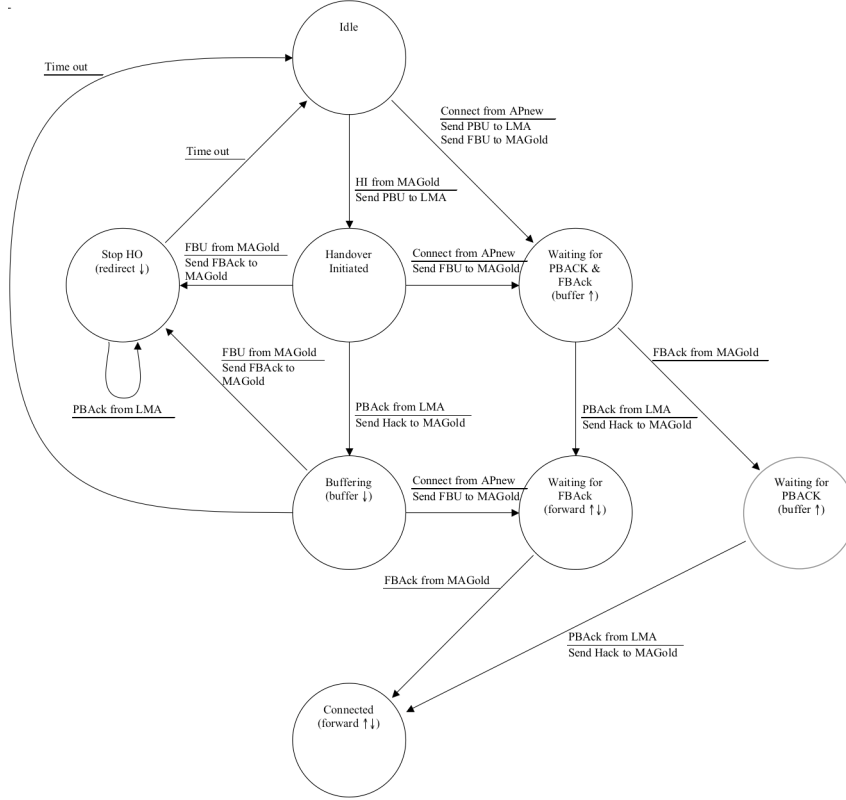


Fig. 6. Finite state machine of MAGnew operation

## 4 Analysis

In order to get insight into the handover latency ( $d_{HO}$ ) and packet loss due to handover ( $l_{HO}$ ), we construct a basic model to express these performance measures in terms of the round trip times (RTTs) between the various nodes.

### 4.1 Notation

We define  $d_{HO}$ , the handover latency, as the time during which no packets are received by the IP layer in the MN due to a handover. Further,  $l_{HO}$ , the handover loss period, denotes the number of packets that are lost because of the handover, measured as a time interval during which generated packets are lost. Finally,  $d_{add}$  denotes the maximum additional packet delay, i.e., the maximum delay that is experienced by packets during the handover due to buffering and redirection.

These performance measures will be expressed as a function of a number of network parameters. The first one is the prediction time,  $T_{pred}$ , which is the time that elapses between prediction of a handover and the possibly forced disconnection of the MN from the old AP. The layer 2 (L2) handover delay,  $D_{HO,L2}$ , denotes the time elapsing between the moment a MN is disconnected from the old AP until it has connected to the new AP, authenticated itself to the new AP, and notified the new MAG of its attachment. Finally we denote the RTT between MAG and LMA as  $RTT_{MAG-LMA}$ , the RTT between a MAG and an AP connected to it as  $RTT_{MAG-AP}$ , and the RTT between two neighboring MAGs as  $RTT_{MAG-MAG}$ .

## 4.2 Model

The model we are constructing to evaluate the handover latency and packet loss is an elementary one. We basically sum the delay components involved in the handover, and assume the individual components to be fixed. Our assumption is that processing delays can be neglected, and that the dominant delay components are those for the transfer of messages between the nodes involved. Which components have to be taken into account depends on the moment the handover is predicted, i.e. it depends on  $T_{pred}$ .

Let us first determine the packet loss during a handover. If the handover is predicted timely, no packets will be lost during handover, since all packets arriving at the old MAG can be delivered to the MN and packets arriving at the new MAG will be buffered until the MN has connected. If there is no timely prediction of the handover, packets that are already sent by the old MAG to the old AP before the old MAG gets notified of the disconnection get lost. This is at most the amount of data generated during one RTT between MAG and AP. It can be less, if the disconnection was predicted a little too late, but the LMA has already started redirecting the traffic from some point during the disconnection. The handover packet loss period can be summarized as

$$l_{HO} = \min(RTT_{MAG-AP}, \max(0, RTT_{MAG-AP} + RTT_{MAG-MAG} / 2 + RTT_{MAG-MAG} - T_{pred})) \quad (1)$$

The handover latency is a bit more difficult to determine. If the handover can be predicted timely, the handover latency is limited to the layer 2 handover latency plus the time needed for the new MAG to inform the old MAG that the binding in the LMA has been changed, i.e.,

$$d_{HO} = D_{HO,L2} + RTT_{MAG-MAG} / 2 \quad (2)$$

(for  $T_{pred} \geq RTT_{MAG-MAG} + RTT_{MAG-LMA}$ ).

The latter term may be reduced if the disconnection is just a bit earlier than predicted. This way, less time is wasted, where the LMA is already routing packets to the new MAG, and the new MAG is still informing the old MAG that it can disconnect the link,

$$d_{HO} = D_{HO L2} + T_{pred} - RTT_{MAG-LMA} - RTT_{MAG-MAG} / 2 \quad (3)$$

$$\text{for } \begin{cases} T_{pred} \leq RTT_{MAG-MAG} + RTT_{MAG-LMA} \\ T_{pred} > RTT_{MAG-MAG} / 2 + RTT_{MAG-LMA} \end{cases}$$

The handover latency is at its minimum if the old link disconnects prematurely, but the handover has been predicted a bit earlier, so that the new MAG has already sent a binding update to the LMA, and has already received packets from the LMA to forward to the MN, once it connects to one of its APs:

$$d_{HO} = D_{HO L2} \quad (4)$$

$$\text{for } \begin{cases} T_{pred} \leq RTT_{MAG-MAG} / 2 + RTT_{MAG-LMA} \\ T_{pred} > RTT_{MAG-LMA} + RTT_{MAG-MAG} / 2 \\ \quad -(D_{HO L2} - RTT_{MAG-AP}) \end{cases}$$

If the time between prediction and disconnection is even shorter, the handover latency will be increased, because after completion of the L2 handover, the MN has to wait until the new MAG has finished its binding update to the LMA and is receiving data from the LMA:

$$d_{HO} = RTT_{MAG-LMA} + RTT_{MAG-MAG} / 2 + RTT_{MAG-AP} - T_{pred} \quad (5)$$

$$\text{for } \begin{cases} T_{pred} \leq RTT_{MAG-LMA} + RTT_{MAG-MAG} / 2 \\ \quad -(D_{HO L2} - RTT_{MAG-AP}) \\ T_{pred} > RTT_{MAG-LMA} + RTT_{MAG-AP} \\ \quad -RTT_{MAG-MAG} / 2 - D_{HO L2} \end{cases}$$

The latter condition is included in Eq. 5 because if the disconnection occurs really early, just after, or even without prediction, the handover latency is determined by the time that is needed to perform the handover at L2 plus the time needed to request the old MAG to forward data to the new MAG. This results in the following handover latency:

$$d_{HO} = D_{HO L2} + RTT_{MAG-MAG} \quad (6)$$

$$\text{for } \begin{cases} T_{pred} \leq RTT_{MAG-LMA} + RTT_{MAG-AP} \\ \quad -RTT_{MAG-MAG} / 2 - D_{HO L2} \\ T_{pred} \geq 0 \end{cases}$$

Note that the handover latency will result in an additional packet delay. The maximum additional packet delay,  $d_{add}$ , is the difference between the handover latency and the packet loss (period), i.e.,

$$d_{add} = d_{HO} - l_{HO} . \quad (7)$$

If the mobile node has 2 interfaces, packets routed by the LMA towards the new MAG can be forwarded to the MN via the connected second interface right away. Therefore, for the 2 interface case, Eq. (2) – (4) can be replaced by

$$d'_{HO} = \max(0, RTT_{MAG-LMA} + RTT_{MAG-MAG} / 2 + RTT_{MAG-AP} - T_{pred}) \quad (8)$$

for  $T_{pred} > RTT_{MAG-LMA} + RTT_{MAG-AP} - RTT_{MAG-MAG} / 2 - D_{HO L2}$  .

The packet loss due to handover remains the same for the 2 interface case.

Finally, let us determine the handover latency in case a handover is wrongly predicted. If, after installing the binding with the new MAG in the LMA, the MN is not able to connect to an AP of the new MAG, and connects to a different MAG, the handover delay includes the L2 handover delay plus the time needed to either receive data from the LMA, or from the wrongly predicted MAG:

$$d''_{HO} = RTT_{MAG-MAG} / 2 + D_{HO L2} \quad (9)$$

+  $\min(RTT_{MAG-LMA}, 2RTT_{MAG-MAG}) + RTT_{MAG-AP} / 2$  .

Wrong prediction does not lead to additional packet loss. However, packets may experience an additional delay of up to the handover latency.

### 4.3 Numerical results

We will now provide some numerical results for the FPMIPv6 proposal. Values for network parameters are as much as possible taken from [13]. This implies that we take for the L2 handover delay  $D_{HO,L2} = 11$  ms, which was measured in [13]. For the various RTTs, we assume  $RTT_{MAG-LMA} = 30$  ms,  $RTT_{MAG-MAG} = 10$  ms, and  $RTT_{MAG-AP} = 5$  ms.

Fig. 7 shows the results obtained when varying  $T_{pred}$ . Apart from the handover delay and loss for the FPMIPv6 proposal, the same measures for the original PMIPv6 proposal are shown. For PMIPv6, the handover latency is simply the sum of the L2 handover delay and a round trip time between MAG and LMA. The PMIPv6 handover loss period lasts a full RTT between AP and LMA plus the L2 handover delay.

It can be observed from Fig. 7 that the performance of FPMIPv6 depends on the timely prediction of the handover. If the loss of the link to the old AP can be predicted at least 40 ms before the actual loss, the handover latency is limited to 16 ms, whereas packet loss can be avoided. If the handover comes totally unpredicted, the handover latency will be 21 ms, and 5 ms of packets will be lost. These figures are a significant improvement when compared to the 41 ms handover latency and 46 ms of packets lost for handovers in PMIPv6. It should be noted that FPMIPv6 might introduce an additional packet delay of up to 16 ms during handover. This could be compensated for by a play out buffer at the receiver.



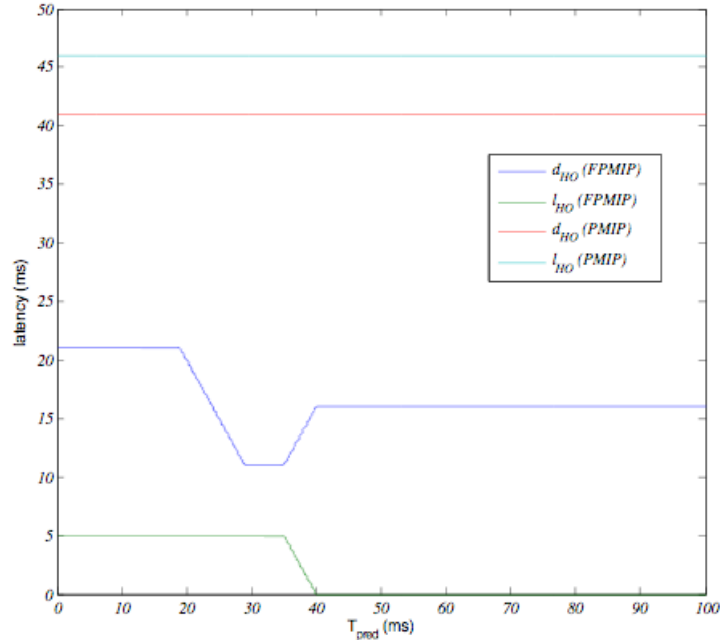


Fig. 7. Handover latency and packet loss

If 2 interfaces are used, performance is even better. If the loss of connection to the old AP is predicted at least 40 ms in advance, no noticeable handover latency and packet loss will be observed. In case a handover is wrongly predicted, the handover latency can be up to 38.5 ms. In order to deal with the additional packet delay, the receiver needs a considerable play-out buffer. Finally, it should be noted that in case of a premature handover, a series of packets might be delivered to the MN out of sequence. This will trigger TCP’s fast recovery, but usually only once, as the delayed packets will usually arrive before a retransmitted packet.

## 5 Conclusions and future work

Two important issues when dealing with an IP-level mobility protocol for future wireless networks are (1) to give the network more control over the handover process, and (2) to reduce the handover latency and packet loss, compared to traditional approaches. In this paper we have proposed an extension to Proxy Mobile IPv6, which we have called Fast Proxy Mobile IPv6 (FPMIPv6).

Performance analysis reveals that the handover latency can be reduced to the time needed to perform the handover at layer 2 plus half a round trip time between two neighboring MAGs. Furthermore, if the handover is predicted correctly and timely, the loss of data can be avoided. To that end, handovers should be predicted a few tens of milliseconds in advance. Packets may experience an additional delay of up to the

L2 handover latency plus 1 or 2 round trip times between neighboring MAGs. The latter occurs in case the old MAG wrongly predicts a handover. However, this additional delay may be compensated for with a play-out buffer at the receiver. In case a mobile node is able to establish a new radio link before releasing the old one through the use of multiple (virtual) interfaces, performance can be further improved.

We are currently measuring the performance of FPMIPv6 using a prototype implementation. Future work includes the detailed specification and full implementation of the protocol, a study of layer 2 requirements from the protocol, especially in the presence of multiple interfaces. Finally, the end-to-end performance of the protocol for e.g., VoIP and TCP traffic needs to be investigated.

## References

- 1 3GPP TR 25.912, "Feasibility Study for Evolved UTRA and UTRAN", v7.2.0, June 2007.
- 2 IEEE 802.11, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications", 1999.
- 3 3GPP TR 23.882, "3GPP System Architecture Evolution: Report on Technical Options and Conclusions", release 7, v1.11.0, July 2007.
- 4 Mobile IT Forum 4G mobile system requirements document (ver. 1.1), retrieved on 28/09/2007 from [http://www.mitf.org/public\\_e/archives/4G\\_req\\_v110E.pdf](http://www.mitf.org/public_e/archives/4G_req_v110E.pdf).
- 5 S. Gundavelli, K. Leung, V. Devarapalli, K. Chowdhury, B. Patil, Proxy Mobile IPv6, IETF NETLMM WG Internet-draft, draft-ietf-netlmm-proxymip6-06, September 23, 2007.
- 6 IETF RFC 4068, "Fast Handovers for Mobile IPv6", July 2005.
- 7 IETF RFC 3775, "Mobility Support in IPv6", June 2004.
- 8 IETF RFC 4140, "Hierarchical Mobile IPv6 Mobility Management (HMIPv6)", August 2005.
- 9 S. Park, J.E. Lee, J. Choi, Y. Kim, "Fast Localized Proxy Mobile IPv6 (FLPMIPv6)", IETF Internet-draft draft-park-netlmm-fastpmip-00, February 26, 2007.
- 10 F. Xia, B. Sarikaya, "Mobile Node Agnostic Fast Handovers for Proxy Mobile IPv6", IETF Internet-draft draft-xia-netlmm-fmip-mnagno-01, July 5, 2007.
- 11 T.-H. Kim, Q. Yang, J.-H. Lee, S.-G. Park, Y.-S. Shin, "A Mobility Management Technique with Simple Handover Prediction for 3G LTE Systems", in Proceedings of the 66<sup>th</sup> IEEE Vehicular Technology Conference, VTC-2007 Fall, September 30 – October 3, 2007.
- 12 R. Chandra, P. Bahl, and P. Bahl, "MultiNet: Connecting to multiple IEEE 802.11 networks using a single wireless card", in Proceedings of IEEE INFOCOM, 2004.
- 13 J. Laganier, M. Flege, A. Zugenmaier, A. Prasad, J., Kempf, J. Wood, "Travelling without Moving: 802.11 Access Points backed by Secure NETLMM", in Proceedings of 16th International Conference on Computer Communications and Networks 2007, ICCCN 2007, August 13-16, 2007.

## Part II

# Short papers and abstracts



# TCP Performance and Packet Aggregation in Wireless Mesh Networks under Low Link Quality

Jonas Karlsson, and Andreas Kassler

Karlstads Universitet, Universitetsgatan, 65188 Karlstad, Sweden  
{Jonas.Karlsson, Andreas.Kassler}@kau.se

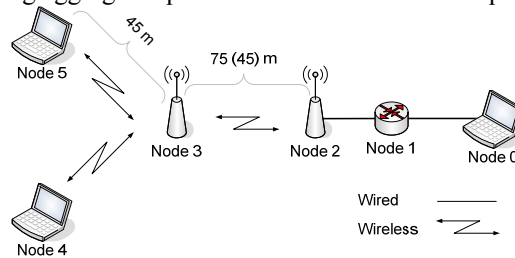
**Keywords:** TCP, MSS, Packet Aggregation, Wireless Mesh Networks

## 1 Extended Abstract

Recently, Wireless Mesh Networks (WMNs) have attracted attention as means to provide alternative internet connectivity to rural areas or communities. In WMNs, wireless access points communicate with each other wirelessly forming a true wireless, mesh based access network of mesh relay nodes (MRN). Mesh gateways (MG) provide internet connectivity and standard mobile clients attach to MRNs, which forward packets via other MRNs to other meshed clients or through MGs to the internet. Therefore, the wireless backbone comprised of MRNs and MGs is similar to static, internet connected Ad Hoc networks. A major problem is however scalability of WMNs as well as MAC and PHY layer overhead for packet transmission. Capacity of WMNs can be increased significantly by aggregating (combining) several smaller packets into larger ones [1][2]. This is in particular beneficial to VoIP flows where packet sizes are small. The overall number of packets is reduced, average packet size increased and contention will take place only once for the larger packets. A relay node will then process fewer large packets instead of many small ones. While such aggregation mechanisms have been proposed for single-hop infrastructure WLAN, designing an aggregation strategy for multi-hop WMNs is a hard problem because in this multi-hop environment, signal quality and congestion for each link is different. When mesh relay nodes aggregate small packets, there is an inherent trade-off regarding packet size. Aggregating more packets leads to larger packets, which reduces the overall number of packets in the mesh and thus reduces multi-hop contention and packet loss due to collisions. However, such larger (aggregated) packets can lead to higher packet loss for a link that operates at low signal quality [3]. To find the optimum frame size for packet aggregation is therefore not trivial and depends on traffic and link quality which can vary over time and might be different for each link in a multi hop path.

In this paper, we will evaluate if packet aggregation is beneficial when applied to standard TCP in a WMN environment with low quality links. For integration of Mesh Networks into 4G environment, the transport layer needs to be compatible and interoperable with already deployed transport protocols. We will therefore compare end-to-end performance of standard TCP (with Selective and Delayed ACKs) using different MSS sizes in a mesh environment, both with and without deploying packet aggregation. Delayed ACKs and Selective ACKs has shown to be beneficial in

multihop environments as the number of packets is reduced leading to less contention [4]. As we will see, packet aggregation is, with proper tuning of parameters, also beneficial for TCP flows as it even more reduces the number of packets and collision probabilities. When using 802.11 MAC layer in WMNs, frame size on MAC layer is typically directly proportional to the chosen TCP MSS size and varying MSS size directly impacts MAC layer frame sizes. To maximize TCP throughput for an end-to-end path across a WMN, packet loss due to bit errors should be minimized to avoid coarse grained timeouts at the TCP sender. As different links in a path might have different qualities, the MSS should thus be sufficiently low to accommodate the weakest link on the path which will lead to underutilization on those links with good channel quality. However, when hop-by-hop packet aggregation is deployed in a mesh, MAC frame size within the path on each link can be optimized by adaptively aggregating [2] IP packets in mesh relay nodes together to form larger MAC layer frames by optimizing aggregated packet size to individual link qualities.



**Fig. 1.** Arrow Topology

To investigate the impact of adaptive packet aggregation and selected MSS size on TCP performance, we conducted several simulations using a MSS of 536 and 1460 bytes in a simple arrow topology with 2 wireless hops and 4 TCP flows with infinite backlogs. Two flows go from node 4 and 5 towards the wired host (node 0), one flow from node 0 to node 4 and one flow from node 0 to node 5, see Fig. 1. We used TCP with selective and delayed ACKs in NS 2.26 [5] with a modified adaptive packet aggregation scheme from [2] to also aggregate TCP and ACK packets. We have extended ns-2 to model a more realistic PHY layer implementation of 802.11a by taking into account packet loss due to bit errors as a result of low link quality, including effects of modulation scheme and convolution coding according to [3]. The distance between node 3 and node 2 was set to 75m, there where no hidden nodes in the scenario. As a reference we also simulated a distance of 45m (referred to as “good channel” in Table 1), where no packet loss due to bit-errors occurred. In our simulation a distance of 75 m between the nodes represents a link that showed 0.05 % packet loss at a MAC frame size of 1064 bytes. Shadowing propagation model was used with path loss exponent of 3. The wireless bandwidth was fixed to 24Mb/s data rate and 6 Mb/s basic rate, delay on the wired link between node 0 and node 1 was set to 1 ms as to simulate a local download to/from the mesh. The results from Table 1 show that using a small MSS where one link shows low signal quality, 75 m case, improves end-to-end performance both with and without packet aggregation. Performance improvement to use a MSS of 1460 instead of 536 is around 26% when aggregation is deployed and around 20 % without packet aggregation, for low link

quality. From Table 1 we can also see that the improvement of enabling packet aggregation in a good channel with an MSS of 536 is about 44%, obviously the larger MSS size is not punished by higher packet drop rates as in the case of a bad channel but even so the performance with packet aggregation and a MSS of 536 performs equally with a MSS of 1460 with no aggregation. This is mainly due to the fact that TCP ACKs can be aggregated together which leads to reduced contention on the medium. In good channel setup, MSS of 536 with aggregation performs slightly worse than using MSS of 1460 without aggregation. In bad channel environment, using small MSS with aggregation outperforms all other settings. The simulations also showed that the fairness is very good, where scores using Jain's fairness index [6] varied between 0.84 and 1.00, with aggregation. Worst fairness showed MSS 1460 under bad channel conditions. Without packet aggregation the fairness varied between 0.92 and 1.00, once again MSS 1460 with a bad channel had worst fairness. We will however remind about the fact that there were no hidden nodes in this simulation setup.

**Table 1.** Aggregated Throughput with bottleneck link, Kbytes/s.

MSS	Without Aggregation	With Aggregation	Improvement of aggregation
536	352 (574 good channel)	371 (827 good channel)	5% (44%)
1460	294 (882 good channel)	294 (938 good channel)	0% (6 %)
	20% improvement using smaller packet size for bad channel	26% improvement using smaller packet size for bad channel	

**Table 2.** Average packet loss ratio (in brackets results 45 m distance)

MSS	Without Aggregation	With Aggregation	Improvement of aggregation
536	$4.7 \cdot 10^{-3}$ ( $5.8 \cdot 10^{-3}$ good channel)	$2.6 \cdot 10^{-3}$ ( $10^{-5}$ good channel)	80%
1460	$14.6 \cdot 10^{-3}$ ( $6.2 \cdot 10^{-3}$ good channel)	$17.1 \cdot 10^{-3}$ ( $10^{-4}$ good channel)	-15%

Table 2 shows average packet loss ratio (PLR). As can be seen for the good channel, aggregation significantly reduces PLR both for large and small MSS, as it reduces the number of packet transmissions and the collision probabilities. This is the main reason for the performance improvement of deploying aggregation in the mesh. In bad channel case adaptive aggregation also leads to improvements in PLR when small MSS is used. However, when large MSS is used, aggregation leads to slightly higher PLR. The reason for this could be that the equation used to estimate the optimal size for an aggregated packet was fine-tuned for VOIP but further investigation is required for TCP; which is left for future work.

We conclude that using standardized MSS size of 1460 bytes will yield an unnecessary large packet drop rate under presence of links with high bit error rate in an end-to-end path through Wireless Meshed Networks, compared to using a smaller MSS. We also show that when packet aggregation is used, performance can be increased in both good and bad channel conditions. As a summary, due to the

performance gains that could be exploited in both bad and good channel conditions with the combination of packet aggregation and proper tuning of TCP MSS, packet aggregation is an interesting alternative for improving capacity in wireless mesh networks for TCP flows. Our future work will concentrate on fine tuning the adaptive packet size equation to suit TCP traffic better. We will also study the impact of congestion losses in fixed networks on E2E performance where some hops traverse a wireless mesh network, as well as studying different traffic scenarios in the presence of hidden nodes, including more hops as well as different hop lengths in the mesh.

## References

1. Castro, M.C., Dely, P., Karlsson, J., Kessler, A.: Capacity Increase for Voice over IP Traffic through Packet Aggregation in Wireless Multihop Mesh Networks. In: Future Generation Communication and Networking, vol.2, no., pp. 350--355, (2007)
2. Castro M.C., Dely P., Kessler A.: VoIP Packet Aggregation based on Link Quality Metric for Multihop Wireless Mesh Networks. In: Proceedings of the Future Telecommunication Conference, Beijing, China, (2007)
3. Mangold, S., Choi, S., Esseling, N.: An Error Model for Radio Transmissions of Wireless LANs at 5GHz. In: Proc. Aachen Symposium'2001, Aachen, Germany, pp. 209--214, (2001)
4. S. Xu, T. Saadawi: Performance evaluation of TCP algorithms in multi-hop wireless packet networks. In: Journal of Wireless Communications and Mobile Computing, vol. 2, no. 1, pp. 85--100, (2002)
5. UCB/LBNL/VINT. Network simulator - (version 2). 1999, <http://nslam.isi.edu/nslam/>
6. R. K. Jain, D.-M. W. Chiu, W. R. Hawe.: A quantitative measure of fairness and discrimination for resource allocation in shared computer systems. In: Technical report, DEC Research Report TR-301, Digital Equipment Corporation, Hudson MA, (1984)



# TCP Performance over Wireless Channels with Unreliable ARQ

Dmitri Moltchanov, Roman Dunaytsev, Yevgeni Koucheryavy

Department of Communications Engineering, Tampere University of Technology  
P.O. Box 553, FIN-33101, Tampere, Finland  
{moltchan, dunaytse, yk}@cs.tut.fi

**Abstract.** Providing reliable data communications over unreliable wireless channels is a challenging task due to the time-varying wireless channel characteristics that often lead to bit errors. These errors propagate to higher layers causing loss of protocol data units. ARQ and FEC try to prevent this by providing a reliable service to the higher layers. Most analytical models proposed to date assumed that the ARQ protocol is fully reliable meaning that a frame is always successfully transmitted irrespective of the amount of time it takes. In this project, we study the effect of a semi-reliable data-link layer on performance experienced by TCP. We develop an analytical model for a TCP connection running over a wireless channel with a semi-reliable ARQ, where the number of retransmission attempts is limited by some number. The model allows to evaluate the effect of many parameters of wireless channels on TCP performance making it suitable for performance optimization studies. These parameters include stochastic properties of the wireless channel characteristics, the size of protocol data units at different layers, the ARQ/FEC settings, the buffer size at the IP layer, and the service rate of the wireless channel.

## 1 Background

Most Internet applications and services, such as WWW, FTP, e-mail, instant messaging, and peer-to-peer file sharing, require guaranteed complete and error-free end-to-end data delivery. Since these applications and services are predominant in the Internet, error control has become an important issue both in wired and wireless networks. In wireless networks, error control requires even more attention than in wired networks due to the time-varying wireless channel characteristics that often lead to bit errors. These errors propagate to higher layers causing loss of protocol data units.

Automatic repeat request (ARQ) and forward error correction (FEC) are very popular error control techniques used in wireless networks. ARQ involves using redundant information embedded in the transmitted data to detect errors at the receiver and then returning a message to the sender requesting retransmission of the erroneously frame. In contrast to ARQ, FEC involves the addition of redundant information embedded in the transmitted data so the receiver can detect errors and correct them without requiring a retransmission. Adding more check bits reduces the amount of available bandwidth, but also enables the receiver to correct more errors. Thus, FEC imposes a greater bandwidth overhead than ARQ, but is able to recover from errors

more quickly. However, even very powerful error-correcting code may not be able to correct all errors that arise in a wireless channel. Hence a further error-control mechanism is required, in which errors in data are detected and the data are retransmitted. Due to complementary advantages, ARQ and FEC are often used in combination.

The maximum number of retransmission attempts is an important configurable parameter of an ARQ protocol that affects performance of data transmission over wireless channels. Setting this parameter to infinity results in completely reliable operation of the data-link layer. However, when the channel conditions are “bad” for relatively long period the delay of end-to-end data delivery increases significantly. This may lead to a number of undesirable effects for delay-sensitive applications as well as buffer overflow at the source (e.g., wireless access point) and increase of end-to-end delay for other packets in the buffer. In practice, the number of retransmission attempts is usually limited allowing some frames to be lost. In this case, the service provided by the data-link layer is considered to be semi-reliable.

Lost frames propagate to higher layers resulting in loss of IP packets and, consequently, TCP segments encapsulated into these packets. TCP was initially developed to operate over wired, rather wireless networks, where the primary cause for packet losses is buffer overflow at the IP layer due to network congestion. Since TCP cannot distinguish packet losses due to bit corruption in wireless channels from those due to network congestion, loss or excessive delay of frames at the data-link layer triggers TCP congestion control procedures. Therefore, the choice of ARQ/FEC parameters may severely affect TCP performance.

## 2 The Model

Using the earlier obtained results [1], [2], we develop an analytical model for data transmission over a wireless channel that explicitly takes into account the effect of a semi-reliable data-link layer. Data loss is allowed to occur due to both excessive number of retransmission attempts at the data-link layer and buffer overflow at the IP layer. The performance metric of interest is long-term steady state goodput of a TCP connection running over a wireless channel. Note that within the proposed framework we also allow the wireless channel model to follow a quite arbitrary Markov model. The developed model is a general framework rather than a model for a particular wireless technology. However, it can be easily extended by adding specific details of state-of-the-art wireless systems.

## References

1. Moltchanov, D., Koucheryavy, Y., Harju, J.: Simple, Accurate and Computationally Efficient Wireless Channel Modeling Algorithm. In: Braun, T., Carle, G., Koucheryavy, Y., Tsaoussidis, V. (eds.) WWIC 2005. LNCS, vol. 3510, pp. 234–245. Springer, Heidelberg (2005)
2. Moltchanov, D., Dunaytsev, R., Koucheryavy, Y.: Cross-Layer Modeling of TCP SACK Performance over Wireless Channels with Completely Reliable ARQ/FEC. In press

# ATOM: Adaptive Transport over Multipaths in Wireless Mesh Networks<sup>\*</sup>

Thomas Staub and Torsten Braun

Institute of Computer Science and Applied Mathematics, University of Bern,  
Neubrückestrasse 10, 3012 Bern, Switzerland  
`staub|braun@iam.unibe.ch`

**Abstract.** In this paper we propose the ATOM (Adaptive Transport over Multipaths) architecture to enable real-time communications in Wireless Mesh Networks (WMNs). WMNs provide robust communication facilities which are even available when certain other systems fail. On the other hand, real-time applications (e.g., IP telephony, video conferencing) in WMNs suffer from transmission errors, high delays, etc. caused by the unreliable wireless medium. ATOM reduces these effects by using path diversity and multi-stream coding. At the session establishment, ATOM decides on the used parameter set (encodings, paths etc.) considering current network conditions and collected historic data. Then, after the session establishment, the effect of this decision is continuously monitored. If necessary, the decision is adapted.

## 1 Motivation

Wireless Mesh Networks (WMNs) emerge as an important technology for increased wireless network coverage [1] and are becoming more and more popular (MIT Roofnet [2], deployments in cities [3] etc.). Wireless mesh nodes are interconnected by wireless links and therefore WMNs do without a wired backbone. A mesh network is automatically established over the wireless links among the mesh nodes. This multi-hop wireless ad-hoc network is dynamically adapted by its self-organization capability. As all communication inside a WMN is using wireless links, there is no need for a wired backbone like in conventional wireless access networks. This makes the deployment of WMNs relatively easy and cost-efficient.

WMNs provide a robust and redundant communication infrastructure even in situations where existing systems, e.g. GSM (Global System for Mobile Communications), might be overloaded or fail. WMNs are robust against partial failures. Unlike the GSM network, where the failure of one base station disconnects a whole region, WMNs can tolerate node and link failures due to redundancy and self-organization in the network. Therefore, WMNs provide increased network resilience. This makes them an interesting candidate for real-time emergency communications, e.g. IP telephony or video conferencing.

---

<sup>\*</sup> The work presented in this paper was supported by the Swiss National Science Foundation under grant number 200020-113677/1.

Unfortunately, the performance of real-time communication suffers from the quality variations of the wireless links in WMNs. The quality of real-time communications is degenerated. The noisy communication channel can cause bit errors in the data transmission. This requires either additional redundancy or retransmissions and therefore reduces the bandwidth. Moreover, fluctuations in the received signal strength due to interference or other changes in the environment can cause link failures. High delays or packet loss rates for the transmission are the consequence. The end user then gets bad quality in his received transmission, e.g., high rate of artefacts, stumbles, or even interruptions. This makes the deployment of real-time applications a challenging task.

Path diversity and multi-stream coding offer support for real-time applications in WMNs. Usually, multiple paths in the network are not affected by the same errors, delays, jitter and loss rates at the same time. Their error characteristics are mostly uncorrelated. Therefore a transmission over multiple paths (path diversity) can compensate the effects of the wireless medium by adding redundancy to the transmission. Instead of simply sending the stream multiple times, advanced multi-stream coding schemes such as layered (LC) or multi-description coding (MDC) can be used [4–6]. Layered coding requires the error free reception of at least the base layer stream to reconstruct the input stream. Whereas in multi-description coding, any combination of received streams can be used for reconstruction. The appropriate selection of encoding and the number of paths depends on the situation [7]. Therefore, the coding selection, the number of used paths, and the mapping of the streams to the individual paths have to be dynamically adapted according to current network conditions.

We propose a new architecture for adaptive transport over multipaths (ATOM) to solve this optimisation problem. ATOM enables real-time communications in WMNs by using path diversity and multi-stream coding. The architecture is described in the next section.

## 2 Architecture of Adaptive Transport over Multipaths (ATOM)

The ATOM architecture enables real-time communication in WMNs. For this purpose, it combines path diversity and multi-stream coding with dynamic adaptation to the current network conditions. Multi-path routing delivers multiple independent paths, the application provides several multi-stream encodings, and the ATOM architecture selects the appropriate set of paths, encoding, and mapping of streams to paths. The decision is supported by network measurements, monitoring, and statistical evaluation. It is adapted if the network conditions have changed.

The ATOM architecture at the end system is shown in Fig. 1. It consists of the following components which are then described in more detail: ATOM controller, ATOM aware application, history and statistical analyzer (HSA), monitoring and measurement system (MMS), multipath routing, path allocator, and end-to-end signalling. Besides these components at the end system, ATOM requires

at least the installation of multi-path routing at all mesh nodes. Furthermore, it makes sense to include HSA and MMS in the mesh nodes. The component placement is discussed later.

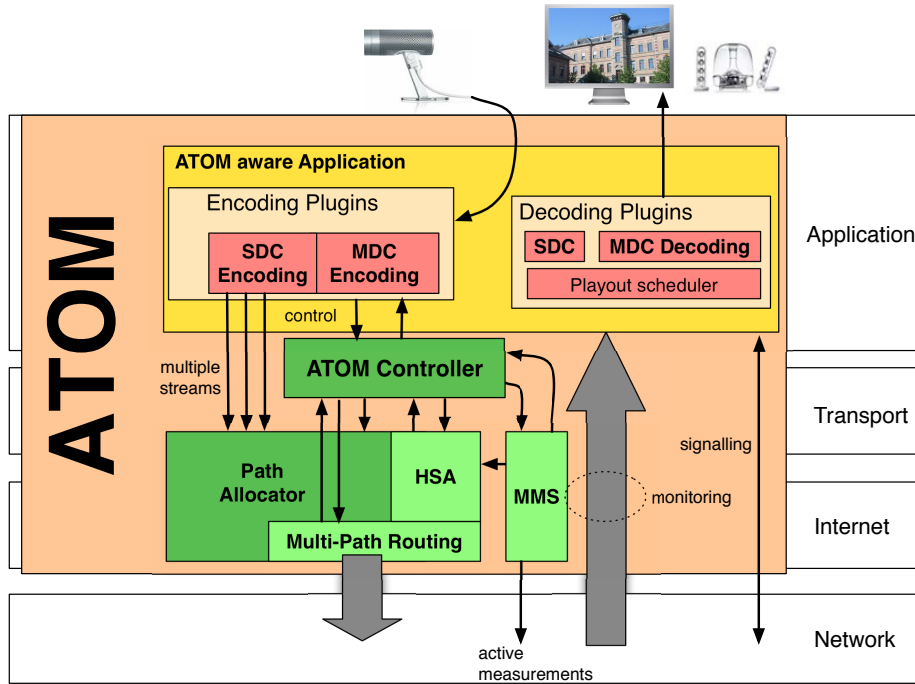


Fig. 1: ATOM end system

The ATOM controller is the central component of the system. It gathers control data and makes the decision about the used communication configuration. The ATOM controller receives data about the available encodings from the application(s), available paths from the multi-path routing, the robustness of the path or individual links from the HSA, and current network conditions from the MMS. Considering this data set, it optimises the parameters for the transmission. This includes the paths to be used, the encoding algorithm (multi-description or layered coding), the number of streams to be encoded, and the mapping of the streams to the paths. If necessary, the ATOM controller triggers the discovery of new paths at the multi-path routing. According to the feedback from the MMS or the application the controller reevaluates the current situation and adapts the transmission parameters accordingly.

The ATOM aware application has to implement the ATOM application programming interface (API) in order to use the ATOM framework. The ATOM API permits the communication with the ATOM controller to exchange the available configuration options and to get the configuration parameters back. The ATOM

aware application informs the controller about available codecs and coding options at the initialisation of the transmission. It then receives the configuration set (encoding, number of streams) from the controller. During an ongoing communication, the application (or user) can give feedback to the ATOM controller in order to revise its decision and to release new parameter settings for this transmission.

ATOM introduces a history and statistical analyser (HSA) to further support the decision process of the ATOM controller. HSA analyses the received data from MMS and provides robustness information about individual paths, regularities and periodicities to the ATOM controller. The received measurement data from the MMS is analysed and stored as a compact meta data description. The description contains information about the stability (fluctuations in availability, bandwidth, and delay) of the paths and the links during the last week, days, hours, and minutes. It further lists remarkable regularities and periodicities in bandwidth, delays, and outages.

The monitoring and measurement system (MMS) observes the network conditions and the quality of ongoing transmissions. This is done by active (network probes) and/or passive measurements. The measurement results are then sent to the ATOM controller as description of the current network state and to the HSA for statistical evaluation. In addition, MMS monitors the current transmissions and provides quality feedback to the remote ATOM controller at the destination.

The required multiple paths from source to destination are delivered by a multi-path routing protocol to the ATOM controller. There are several multi-path routing protocols existing, e.g., AODVM [8], SMR [9], and MP-DSR [10]. We therefore propose to adapt an existing multi-path routing protocol to the ATOM architecture. The routing protocol has to deliver paths that are as independent as possible with certain quality of service (QoS) requirements, e.g., delay below 150ms. The independence of the paths is important in order to take profit of the path diversity and not to be affected by single link and node failures. Moreover, the routing protocol has to deliver the paths marked with a degree of independence (shared links, shared nodes) and QoS parameters to the ATOM controller for the decision making process. We currently favor source routing protocols like SMR and MP-DSR as they deliver the full path in the route discovery process. This eases the path allocation according to QoS parameters. In addition, each packet includes the whole path, which simplifies the monitoring. However, we will evaluate several candidate protocols for routing.

Multi-channel communication can significantly increase the throughput of WMNs. The routing protocol has to consider the presence of multiple channels and radios in its routing decision in order to fully take advantage of the multi-channel communication. This requires more advanced routing metrics than hop count. The hop count metric is based on the wrong assumption that a link either works or not. Packet loss, bandwidth, varying delays, and (self-)interference are not considered. Different enhanced routing metrics are existing, e.g., MIC (Metric of Interference and Channel-Switching) [11], and iAWARE [12]. iAWARE seems to be the most appropriate metric to start with as it incorporates packet

loss, bandwidth, intra-flow interference, and inter-flow interference considering signal strength and traffic amount of the interfering flows. We will therefore evaluate and adapt iAWARE as routing metric for our multi-path routing.

The path allocator is responsible for the allocation of the individual streams from the application to the paths according to the settings received from the ATOM controller. It also delivers the received streams to correct application at the destination.

End-to-end signalling is required for adjusting the parameters at both communication peers. The ATOM aware applications have to signal their encoding options to the ATOM controller and the ATOM controller has to inform the applications and its remote ATOM peer about the used parameter set. A possible signalling protocol for this purpose is SIP (Session Initiation Protocol).

The ATOM components can be placed at different locations in the WMN. Either all components are included in the end system (see Fig. 1) or the core components are only situated in the WMN at the first and last hop router. The placement in the end system requires modifications at each individual client. Besides the ATOM aware application, the ATOM core components have to be installed on the client. This requires modifications in operating system components and all ATOM components have to be provided for the different operating systems of the client devices. Another possibility is to include the ATOM components only in the mesh nodes. In this case, only the ATOM aware application has to be installed on the client devices. There is no need for porting the ATOM core components to the different client operating systems. They have only to be available for the mesh nodes. The ATOM aware application then communicates with the ATOM controller over TCP/IP. Compared to the approach with ATOM at the end system, the mesh only ATOM system may depend on the transmission quality of the link between client and ATOM mesh node. In addition to these approaches, the ATOM architecture supports a combination of the both approaches (some clients with ATOM core, some without) or even legacy applications by interception of the traffic at the border mesh nodes.

### 3 Summary and Future Work

Wireless Mesh Networks provide a reliable and robust communication infrastructure even in situations where other systems may fail. But, it is a challenging task to use them as communication backbone for real-time emergency communications such as IP telephony and video conferencing. The main problem are transmission errors, link errors, packet loss, high delays, and high delay variations caused the unreliable and erroneous wireless medium. This makes the deployment of real-time applications difficult as the user perceived audio/video quality is degenerated (stumbles, artefacts, high delays, or connection losses). Our proposed ATOM (Adaptive Transport over Multipath) architecture provides a comprehensive solution for the support of real-time communication in WMNs. It reduces the effects of the unreliable and erroneous wireless medium by using path diversity, multi-stream coding, and dynamic adaptation.

The presented ATOM architecture is work in progress and we are currently specifying its details. We have also started to implement and test parts of the architecture in a network simulator. Further ongoing activities are the elaboration of decision making process of the ATOM controller and the statistical analysis in HSA. We also plan to implement and evaluate the ATOM architecture on our Linux based wireless mesh network. In order to perform experiments in the WMN at our institute, we have developed a remote management and software distribution solution [13].

## References

1. Akyildiz, I.F., Wang, X.: A survey on wireless mesh networks. *Communications Magazine*, IEEE **43**(9) (2005) S23–S30
2. Bicket, J.C., Aguayo, D., Biswas, S., Morris, R.: Architecture and evaluation of an unplanned 802.11b mesh network. In: *MOBICOM 2005*, Cologne, Germany (August 28 - September 2 2005) 31–42
3. Karrer, R., Sabharwal, A., Knightly, E.: Enabling large-scale wireless broadband: The case for taps. In: *2nd Workshop on Hot Topics in Networks (Hot-Nets II)*, Cambridge, MA, USA (November 2003)
4. Apostolopoulos, J.G., Trott, M.D.: Path diversity for enhanced media streaming. *IEEE Communications Magazine*, special issue on "Proxy Support for Streaming on the Internet" **42**(8) (August 2004) 80–87
5. Liang, Y.J., Steinbach, E.G., Girod, B.: Real-time voice communication over the internet using packet path diversity. In: *MULTIMEDIA '01*, Ottawa, Canada, ACM Press (2001) 431–440
6. Gogate, N., Chung, D.M., Panwar, S., Wang, Y.: Supporting image and video applications in a multi-hop radio environment using path diversity and multiple description coding. *IEEE Transactions on Circuits and Systems for Video Technology* **12**(9) (September 2002) 777–792
7. Chakareski, J., Han, S., Girod, B.: Layered coding vs. multiple descriptions for video streaming over multiple paths. *Multimedia Systems* **10**(4) (2005) 275–285
8. Ye, Z., Krishnamurthy, S.V., Tripathi, S.K.: A framework for reliable routing in mobile ad hoc networks. In: *IEEE Infocom 2003*, San Francisco, CA, USA (May 30 - April 3 2003)
9. Lee, S.J., Gerla, M.: Split multipath routing with maximally disjoint paths in ad hoc networks. In: *IEEE International Conference on Communications (ICC)*. Volume 10., Helsinki, Finland (June 11-14 2001) 3201–3205
10. Leung, R., Liu, J., Poon, E., Chan, A.L.C., Li, B.: MP-DSR: a qos-aware multipath dynamic source routing protocol for wireless ad-hoc networks. In: *IEEE LCN*, Tampa, Florida, USA (November 14 - 16 2001) 132–141
11. Yang, Y., Wang, J., Kravets, R.: Designing routing metrics for mesh networks. In: *First IEEE Workshop on Wireless Mesh Networks (WiMesh)*, Santa Clara, CA, USA (September 26 2005)
12. Subramanian, A.P., Buddhikot, M.M., Miller, S.: Interference aware routing in multi-radio wireless mesh networks. In: *2nd IEEE Workshop on Wireless Mesh Networks (WiMesh 2006)*, Reston, Virginia, USA (September 25 2006) 55–63
13. Staub, T., Balsiger, D., Lustenberger, M., Braun, T.: Secure remote management and software distribution for wireless mesh networks. In: *ASWN 2007*, Santander, Spain (May 24-26 2007) 47–54



# Analysis of cell residence time in WLAN under different mobility models

Enrica Zola, Francisco Barcelo-Arroyo

Universitat Politècnica de Catalunya, UPC  
c/ Jordi Girona 1-3, 08034 Barcelona (Spain)  
{enrica, barcelo}@entel.upc.edu

**Abstract.** The handover process in WLAN involves the use of bandwidth for the necessary signaling and protocols. Several mobility models are available for simulating WLAN and MANET having different impact on the network performance. This paper shows the impact of the mobility model on the cell residence time (i.e. time connected to the same access point) in networks with infrastructure designed for pedestrians. To this purpose, ns-2 simulations are run with different mobility models and number of access points. The time between changes of access point (i.e. handover) is studied as a random variable. The research proves that the impact of the model is very strong when compared to the predictable impact of the terminal speed. In some cases the handover rate is almost doubled by only changing the mobility model while maintaining the average speed. The coefficients of variation are low or very low showing that the connection process (i.e. arrivals to access points) follows distributions with low or very low peakedness factor.

**Keywords:** Mobility models, handover, cell residence time, WLAN.

## 1 Introduction

Mobility models play an important role in the analysis of several issues in wireless networks, such as handover, channel holding time, resource allocation, location updating, and others. In [1], the authors provide a comprehensive survey of mobility models and prove through simulation that the nodes' movements have a noticeable impact on the performance of an ad hoc network. In [2], Qin et al. simulate a CDMA network to study the relationship between the handover rate and the mobile speed. In [3], results of the cell residence time and the channel holding time distribution in cellular mobile systems are obtained after simulating a cellular network. Authors show that the generalized gamma distribution is adequate to describe the cell residence time distribution of handover calls. In [4] and [5], a proposal is presented to dynamically construct the mobility pattern followed by a node and use this history in order to predict future location and then reserve resources for the handover.

The handover process has an impact on the network performance since it uses bandwidth for the necessary signaling and protocols. For a given layout of Access Points (AP) in a WLAN it is predictable that the faster the terminals' movement the

higher the handover rate [2]. It is also predictable that for more APs covering the same area the handover rate will increase. The consequence of more access points is a better coverage, reliability and traffic capacity. However, it cannot be neglected that more APs introduce complexity and lower traffic capacity per AP due to the increased handover rate.

This paper deals with the time that a terminal remains under the coverage of the same AP in a WLAN of small size designed for pedestrians. This is carried out by running ns-2 simulations for different number of AP and mobility models. For a terminal continuously connected to the WLAN, the cell residence time (i.e. time connected to the same AP) is the time between handovers, hence the inverse of the handover rate.

## 2 Mobility Models

In this Section the mobility models used in this work are briefly described: Random Walk, Random Waypoint, Gauss-Markov (see [1] for a comprehensive survey on the models considered in this paper and others).

In the Random Walk model (RWalk), a node progresses from its current location by randomly choosing a direction and speed in which to travel; both direction and speed are selected from predefined ranges,  $[0, 2\pi]$  and  $[v_{\min}, v_{\max}]$  respectively. The node is allowed to travel in such direction and with the given speed for a predefined time or for a predefined distance [1]. If a node reaches a simulation boundary, it “bounces” off the simulation border with an angle determined by the incoming direction, and then continues along this new path. RWalk produces highly unpredictable movements, such as sudden stops and sharp turns: they occur because current speed and direction is independent of past speed and direction.

The Random Waypoint model (RWpnt) was originally proposed in [6]. In this model, each node is assigned an initial location ( $p_0$ ), a destination ( $p_1$ ) and a speed; both  $p_0$  and  $p_1$  are chosen independently and uniformly on the region in which the nodes move. The speed is chosen uniformly (or according to any other distribution) on an interval ( $v_{\min}$  and  $v_{\max}$ ) independently of both the initial location and destination. After reaching  $p_1$  (i.e. after moving on a *leg*, as movement with a given direction and speed is defined), a new destination and a new speed are chosen according to their distributions and independently of all previous destinations and speeds. The node may also remain still for a random pause time before starting its movement towards the next destination. Such a model is used in many prominent simulation studies of ad-hoc network protocols [1, 7, 8]. Similarity of RWpnt with pause time set to zero with RWalk has been stressed in [1].

Spatial node distribution for the RWpnt mobility model has been investigated in recent years. Starting from a formal description of the model in terms of a discrete-time stochastic process, in [8] Navidi and Camp derive the stationary distribution for location, speed and pause time for the RWpnt model in a rectangular area, and provide an implementation of their method for ns-2 simulator. In [12], Bettstetter et al. extend results for circular regions, using approximations at certain steps; in [7], Hyytiä et al. derive an analytical expression for the node distribution in an arbitrary

convex region, without using approximations; the analysis is at the basis of [13] in which authors present an exact analytical formula for the mean arrival rate at a cell.

The Gauss–Markov Mobility Model (Gauss) was originally proposed for the simulation of a PCS [9]; it introduces the concept of a drift in the node’s movement. Initially, each node is assigned a current speed and direction. At fixed intervals of time, movements occur by updating the speed and direction of each node. The next location is calculated based on the current location, speed and direction of movement, according to the following equations:

$$s_i = \alpha s_{i-1} + (1 - \alpha) \bar{s} + \sqrt{(1 - \alpha^2)} s_{x_{i-1}}, \quad (1)$$

$$d_i = \alpha d_{i-1} + (1 - \alpha) \bar{d} + \sqrt{(1 - \alpha^2)} d_{x_{i-1}}, \quad (2)$$

where  $s_i$  and  $d_i$  are the new speed and direction of the node at time interval  $I$ ;  $\alpha$ , where  $0 \leq \alpha \leq 1$ , is the tuning parameter is used to vary the degree of randomness in the mobility pattern;  $\bar{s}$  and  $\bar{d}$  are constants representing the mean value of speed and direction as  $n \rightarrow \infty$ ;  $s_{x_{i-1}}$  and  $d_{x_{i-1}}$  are random variables from a Gaussian distribution: with  $\alpha=0$  Random Walk movements are obtained, while with  $\alpha=1$  linear motion is generated [1]. This model forces a node which is moving away from the area boundaries, to go back to it; therefore, the field size is automatically adapted to the node movements after scenario generation.

In [1] several examples of paths followed through different mobility models are depicted and analyzed proving that both RWalk and RWpnt tend to concentrate node’s movements in the center of the area, while Gauss does not. On the other hand Gauss-Markov can eliminate the sudden stops and sharp turns of RWalk since next movement depends on the current one.

### 3 ns-2 Simulator and Scenarios

This study is based on simulation performed with ns-2 [10]. An 802.11 Mobile Node (MN) is moving in the simulation area while continuously sending data to a target node that remains static (i.e. AP1). The handover process is identified through the analysis of the messages interchanged with the AP. Different scenarios and different mobility models have been taken into account. A brief description follows.

The *setdest* tool from the CMU Monarch group is the mobility generator included in ns-2. The *setdest* tool determines each leg movement (i.e. a destination  $(x,y)$  and a speed) and computes how long does it take to reach destination at such speed: next leg will start from that point. If pauses are settled, in next movement MN may remain still for a predefined time (i.e. according to pause distribution). It is clear that the length of the leg is not constant, but depends on the speed used (i.e. higher speeds mean shorter legs). This implementation follows the RWpnt mobility model described in Section 2. If pause time is set to zero (i.e. RWpntU and RWpnt0), we obtain a movement which is very similar to that of RWalk [1].

**Table 1.** Mobility models and parameters used in simulation.

	<b>RWpntU</b>	<b>RWpnt0</b>	<b>RWpnt1</b>	<b>RWpnt10</b>	<b>Gauss</b>
<b>Minimum speed [m/s]</b>	0.7	0.7	0.7	0.7	0.7
<b>Maximum speed [m/s]</b>	2.0	2.0	2.0	2.0	2.0
<b>Speed type</b>	Uniform	Normal	Normal	Normal	Normal
<b>Actual average speed [m/s]</b>	1.23	1.28	1.26	1.14	1.30
<b>Pause type</b>	Constant	Constant	Uniform	Uniform	Uniform
<b>Pause [s]</b>	0	0	1	10	1

In this work,  $v_{min}$  and  $v_{max}$  are fixed to 0.7 and 2.0 m/s, respectively, as shown in Table 1; simulation was performed setting different pause time and speed distributions. Note that the average speed is not the mean between  $v_{min}$  and  $v_{max}$ , since the duration of each movement at a given speed is not constant. In [13], authors provide the probability distribution of the node's speed at an arbitrary point of time, which is expressed by the following formula:

$$f_v^*(v) = \frac{1}{E[1/v]} \cdot \frac{1}{v} \cdot f_v(v), \quad (3)$$

where  $f_v(v)$  is the speed distribution settled by the user (i.e. uniform or normal).  $E[1/v]$  is assumed to be finite. In case of use of pause time, the probability distribution of the node's speed is expressed by:

$$f_v^*(v) = f_v(v | Paused) \cdot P_{pause} + f_v(v | NotPaused) \cdot (1 - P_{pause}), \quad (4)$$

where the conditional density  $f_v(v | NotPaused)$  equals the density  $f_v^*(v)$  in Eq. (3). It is possible to compute the average time-weighted speed for RWpnt model. As an example, we derive the average time-weighted speed for RWpntU, for which speed  $f_v(v)$  is uniformly distributed between 0.7 and 2.0:

$$E[v] = \int_{0.7}^{2.0} v \cdot f_v^*(v) dv = \int_{0.7}^{2.0} v \cdot \frac{1}{E[1/v]} \cdot \frac{1}{v} \cdot f_v(v) dv = \int_{0.7}^{2.0} v \cdot \frac{1}{0.80756 \cdot v} dv = 1.23, \quad (5)$$

where  $E[1/v]$  can be evaluated, according to [8], as:

$$E[1/v] = \int_{0.7}^{2.0} \frac{1}{v} \cdot f_v(v) dv = \frac{\ln(2.0/0.7)}{2.0 - 0.7} = 0.80756. \quad (6)$$

Analytical calculations for the average speed match the values obtained from simulation and showed in Table 1. An example of the time-weighted speed distribution  $f_v^*(v)$  for RWpnt0 is shown in Fig. 1.

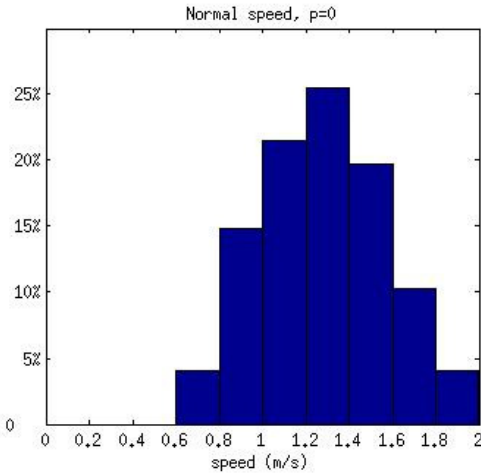


Fig. 1. Time-weighted speed distribution for RWpnt0 model ( $v_{min}=0.7$  m/s,  $v_{max}=2.0$  m/s).

For the Gauss-Markov model, the Bonn Motion Java software [11] has been used, which slightly differs from the original Gauss-Markov model. The main difference is that the new speed and direction are simply chosen from a normal distribution with a mean of the respective old value. In the remainder of the paper, we will refer to this “modified” Gauss-Markov model as “BM-Gauss”.

Three scenarios have been considered in this study. The simulation area is a square with side of 200 meters, in which a number of static nodes (i.e. AP) have been placed. The number of APs is 4, 9 and 16, representing three designs with minimum full coverage, reasonable overcoverage and overcoverage for high capacity respectively. The important rule while positioning these AP’s is to guarantee that they can always stay connected considering that the maximum distance is 100m. Table 2 displays ( $x$ ,  $y$ ) coordinates of the location of the static nodes in each scenario.

To study the dependency of the cell residence time with the mobility pattern we put a mobile node (MN) moving inside the simulation area according to one of the mobility models. Table 1 shows the parameters used for each model in our simulations. The total time of the simulation is 50 hours leading to samples of 400 to 1400 residence times, enough for achieving trustable statistics.

**Table 2.** AP locations in different scenarios: coordinates ( $x$ ;  $y$ ).

4AP		9AP		16AP	
<b>AP1</b>	50.1; 50.1	<b>AP1</b>	33.3; 33.3	<b>AP1</b>	25; 25
<b>AP2</b>	149.9; 50.1	<b>AP2</b>	100; 33.3	<b>AP2</b>	75; 25
<b>AP3</b>	149.9; 149.9	<b>AP3</b>	166.7; 33.3	<b>AP3</b>	125; 25
<b>AP4</b>	50.1; 149.9	<b>AP4</b>	166.7; 100	<b>AP4</b>	175; 25
-	-	<b>AP5</b>	100; 100	<b>AP5</b>	175; 75
-	-	<b>AP6</b>	33.3; 100	<b>AP6</b>	125; 75
-	-	<b>AP7</b>	33.3; 166.7	<b>AP7</b>	75; 75
-	-	<b>AP8</b>	100; 166.7	<b>AP8</b>	25; 75
-	-	<b>AP9</b>	166.7; 166.7	<b>AP9</b>	25; 125
				<b>AP10</b>	75; 125
				<b>AP11</b>	125; 125
				<b>AP12</b>	175; 125
				<b>AP13</b>	175; 175
				<b>AP14</b>	125; 175
				<b>AP15</b>	75; 175
				<b>AP16</b>	25; 175

### 3.1 Handover handling in ns-2

The handover is identified through the analysis of the messages interchanged with the AP. In ns-2 simulator, a handover is performed when MN loses connection with its AP. According to the antenna parameters used in our simulation, the transmission range is 100 m. We detect the start of HO through the AODV error message that MN will send in order to find a new path to its destination (i.e. AP1). The path is always chosen as the shortest path between them: it follows that, with different APs locations, since MN always follows the same path with a given mobility model, the time at which MN loses connection with the current AP will be different in each scenario; depending on its position at that time, AODV will select a different path. As an example, Fig. 2 shows the path followed by MN during the first 1000s according to RWpntU model in a scenario with 4 and 9 APs, respectively. MN starting position is (164.4; 110.9). In the first scenario, MN will associate to AP2 (i.e. positioned at (149.9; 50.1), as shown in Table 2) which delivers its data to the destination AP, since the distance between MN and AP1 is higher than 100m at  $t=0$ ; after 150s, it will be too far from AP2 and then performs HO and associates to AP1, since at that time it is within its coverage area. In 9AP scenario, MN firstly associates to AP5 which is at one step from AP1 (i.e. farer than 100m from MN); after 342s, it loses connection, performs HO and finally associates to AP6, in order to reach its destination.

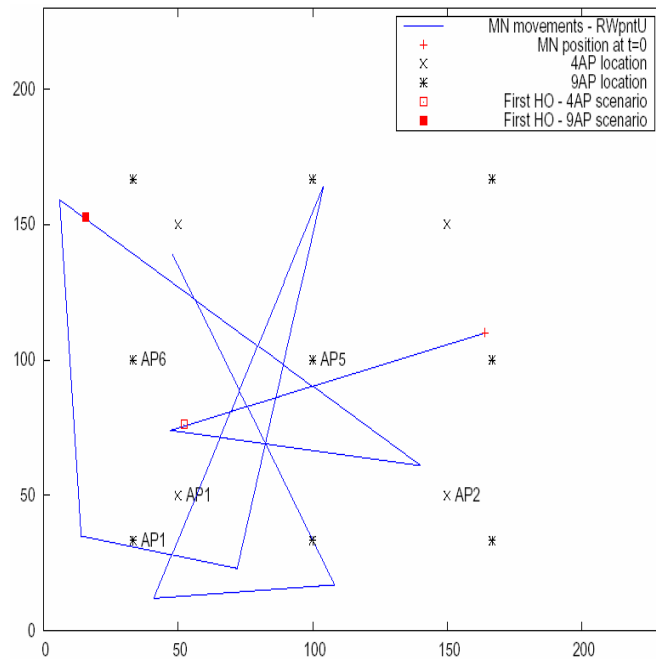


Fig. 2. RWpntU movement pattern during the first 1000s and MN position at first HO in scenarios with 4 and 9 AP.

It can be observed that, in 4APs scenario, the probability for MN to stay in the area covered by AP1 (which means that it will directly connect to it) is 49.79%. It can be calculated as the proportion between the area of the circular sector centered in AP1, with 100m radius and limited by the intersections with the simulation area boundaries (i.e. points (136.6; 0) and (0; 136.6)) and the total simulation area (i.e. 200x200 square). The same probability in 9APs scenario is reduced to 38.74% (i.e. the circular sector is now centered in (33.3; 33.3), which is AP1 location in this scenario), has the same radius of 100m and intersects simulation area boundaries at (127.6; 0) and (0; 127.6)). The probability that MN in 4APs scenario will connect to one of the two AP that stay at one step from AP1 (i.e. AP2 and AP4) is the probability for MN to stay in the area covered by AP2 or by AP4, but not covered by AP1. It can be computed as proportion between the union of the circular areas centered at AP2 and AP4, minus the circular area centered at AP1 (of course, only the area inside the simulation square is considered): that is 41.90%. The same probability in 9APs scenario grows to 67.30%, since there are three AP at just one step from AP1 (i.e. AP2, AP5, and AP6) and one of them nearly covers the whole simulation area (i.e. AP5). We should expect, then, that cell residence time will increase in the latter scenario.

## 5 Results and Analysis

Table 3 displays the mean cell residence time. Since the MN is continuously connected (i.e. the average time during which MN is not connected is always less than 100 ms) this is the time between two consecutive handovers. The duration of the residence time is longer for slower average speeds, produced by longer pause times in this case: only BM-Gauss does not follow this figure and provides the highest value. The importance of the mobility model applied is proved by the noticeable difference between the residence time for the Gauss-Markov and the RWpnt0, both with nearly the same average speed (i.e. 1.30 m/s and 1.28 m/s, respectively). The residence time is about the double for the Gauss-Markov meaning that the number of necessary handovers is about the half. The movement pattern of the Gauss-Markov model is not composed of straight lines and is much smoother than the movements created by the RWpnt. Straight lines tend to increase the handover ratio while bends can keep the terminal under the coverage of the same AP for a longer time.

We also observe that scenario with 9AP provides the highest average residence time, except for BM-Gauss model. As commented before, this increase with respect to

**Table 3.** Average cell residence time (in seconds) and its squared coefficient of variation for each scenario and mobility model.

	4AP	9AP	16AP
<b>RWpntU</b>	148 / 0.56	191 / 1.30	144 / 1.48
<b>RWpnt0</b>	135 / 0.49	168 / 1.27	127 / 1.38
<b>RWpnt1</b>	138 / 0.51	174 / 1.33	128 / 1.44
<b>RWpnt10</b>	155 / 0.57	189 / 1.18	145 / 1.34
<b>BM-Gauss</b>	382 / 0.82	250 / 1.24	247 / 1.46

**Table 4.** Average number of handovers per hour and AP.

	<b>4AP</b>	<b>9AP</b>	<b>16AP</b>
<b>RWpntU</b>	6.08	2.09	1.56
<b>RWpnt0</b>	6.67	2.38	1.77
<b>RWpnt1</b>	6.52	2.29	1.75
<b>RWpnt10</b>	5.81	2.11	1.55
<b>BM-Gauss</b>	2.36	1.6	0.9

a scenario with less AP is consequence of an higher probability that MN moves in the area covered by AP2, AP5 or AP6 (i.e. one of the three APs that offers one-step-distance path to destination AP1) and not covered by the same AP1. In 16APs scenario, such probability decreases since the number of APs at less than 100m from AP1 is still 3 and the area covered by them is less than in the previous scenario: this leads to a decrease in the average cell residence time, as observed in column 4 of Table 3. Squared coefficients of variation of the residence time show figures that are never high (i.e. always lower than 1.5). For a low number of APs, coefficients of variation lower than one suggest that handover arrival process is smooth; approximating this process by Poisson would lead to conservative figures in the design. For overcoverage, the coefficient of variation remains near to unity; approximating by Poisson can keep a certain degree of accuracy. It must be noticed that in all cases the coefficient of variation increases together with the number of AP.

Table 4 shows the average number of handovers per hour and AP. This figure decreases when the number of AP is increased. This is mainly a consequence of the overcoverage: the share of layout covered by more than one AP is larger for more APs.

## 6 Conclusions

In this work the impact of different mobility models in scenarios with a different density of AP has been studied. Simulation proves that it is not always true that, with a higher number of nodes, the average residence time decreases as one should expect. This is almost a consequence of the algorithm used for performing a handover, of the cell coverage and the routing protocol (i.e. shortest path in this study). With the same number of AP, the average residence time is longer if MN moves according to Gauss-Markov mobility model (i.e. smoother changes of direction and higher probability of staying around) and shorter if a memory-less model is used. The impact of using a specific mobility model can be almost twice the handover ratio for the same layout and average speed, stressing the importance of its choice when studying wireless network performances.

**Acknowledgments.** This work was supported by the Spanish Ministry of Science and Technology through CICYT project TEC2006-09466/TCM.



## References

- [1] Camp, T., Boleng, J., Davies, V.: A survey of mobility models for ad hoc network research. *Wireless Communication and Mobile Computing (WCMC)*, vol. 2, no. 5, pp. 483-502 (2001)
- [2] Qin, Y., Xu, X., Zhao, M., Yao, Y.: Effect of user mobility on soft handoff performance in cellular communication. *TENCON '02. IEEE Region 10 Conference on Computers, Communications, Control and Power Engineering*. Vol. 2, pp. 956 – 959 (2002)
- [3] Zonoozi, M., Dassanayake, P.: User mobility modeling and characterization of mobility patterns. *IEEE Journal on Selected Areas in Communications*. Vol. 15, issue 7, pp. 1239-1252 (1997)
- [4] Jayasuriya, A.: Handover channel allocation based on mobility predictions. *Advanced Wired and Wireless Networks*. Vol. 26, pp- 147-169 (2005)
- [5] Sharma, S., Zhu, N.: Low-latency mobile IP handoff for infrastructure mode WLAN. *IEEE Journal on Selected Areas in Communications*. Vo. 22, issue 4, pp. 643- 652 (2004)
- [6] Johnson, D. B., Maltz, D. A.: *Dynamic Source Routing in Ad Hoc Wireless Networks*. Mobile Computing. Vol. 353, pp. 153-181. Kluwer Academic Publishers (1996)
- [7] Hyttiä, E., Lassila, P., Virtamo, J.: Spatial Node Distribution of the Random Waypoint Mobility Model with Applications. *IEEE Transactions on Mobile Computing*. Vol. 5, issue 6, pp. 680 – 694 (2006)
- [8] Navidi, W., Camp, T.: Stationary Distributions for the Random Waypoint Mobility Model. *IEEE Transactions on Mobile Computing*. Vol. 3, issue 1, pp. 99 – 108 (2004)
- [9] Liang, B., Haas, Z.: Predictive distance-based mobility management for PCS networks. In *Proceedings of the Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*. (1999)
- [10] The VINT Project. The network simulator -ns-2. <http://isi.edu/nsnam/ns/>
- [11] <http://web.informatik.uni-bonn.de/IV/Mitarbeiter/dewaal/BonnMotion/>
- [12] Bettstetter, C., Hartenstein, H., Pérez-Costa, X.: Stochastic properties of the Random Waypoint Mobility Model. *ACM/Kluwer Wireless Networks: Special Issue on Modeling and Analysis of Mobile Networks*, vol. 10, num. 5, pp. 555-567 (2004).
- [13] Hyttiä, E., Virtamo, J.: Random waypoint mobility model in cellular networks. *ACM Wireless Networks*, vol. 13, issue 2, pp. 177-188 (2007).



## Author Index

- Aapaoja, Tommi, 25
- Barcelo-Arroyo, Francisco, 107
- Bargh, Mortaza S., 79
- Braun, Torsten, 101
- Castro, Alfredo A. Villalba, 65
- Dunaytsev, Roman, 99
- Dunkels, Adam, 15
- Gómez, Angel M., 49
- González, José A., 49
- Heijenk, Geert, 79
- Kärnä, Juha, 25
- Karlsson, Jonas, 95
- Kassler, Andreas, 95
- Konstantas, Dimitri, 65
- Koucheryavy, Yevgeni, 99
- Laganier, Julien, 79
- Latvakoski, Juhani, 25
- Lo, Anthony, 37
- Moltchanov, Dmitri, 99
- Niemegeers, Ignas, 37
- Österlind, Fredrik, 15
- Peccia, Nestor, 3
- Peinado, Antonio M., 49
- Prasad, Anand R., 79
- Sánchez, Victoria, 49
- Samaras, Christos V., 3
- Serugendo, Giovanna Di Marzo, 65
- Staub, Thomas, 101
- Tsaoussidis, Vassilis, 3
- Voight, Thiemo, 15
- Zhou, Jinglong, 37
- Zola, Enrica, 107

