International Journal on Advances in Networks and Services

2011 vol. 4 nr. 1&2

The *International Journal on Advances in Networks and Services* is published by IARIA.
ISSN: 1942-2644
journals site: http://www.iariajournals.org
contact: petre@iaria.org

Sponsored by IARIA
www.iaria.org

International Journal on Advances in Networks and Services
Volume 4, Number 1 & 2, 2011

Internet and Web Services

- Thomas Michael Bohnert, SAP Research, Switzerland
- Serge Chaumette, LaBRI, University Bordeaux 1, France
- Dickson K.W. Chiu, Dickson Computer Systems, Hong Kong
- Matthias Ehmann, University of Bayreuth, Germany
- Christian Emig, University of Karlsruhe, Germany
- Geoffrey Fox, Indiana University, USA
- Mario Freire, University of Beira Interior, Portugal
- Thomas Y Kwok, IBM T.J. Watson Research Center, USA
- Zoubir Mammeri, IRIT – Toulouse, France
- Bertrand Mathieu, Orange-ftgroup, France
- Mihhail Matskin, NTNU, Norway
- Guadalupe Ortiz Bellot, University of Extremadura Spain
- Dumitru Roman, STI, Austria
- Monika Solanki, Imperial College London, UK
- Vladimir Stantchev, Berlin Institute of Technology, Germany
- Pierre F. Tiako, Langston University, USA
- Weiliang Zhao, Macquarie University, Australia

Wireless and Mobile Communications

- Habib M. Ammari, Hofstra University - Hempstead, USA
- Thomas Michael Bohnert, SAP Research, Switzerland
- David Boyle, University of Limerick, Ireland
- Xiang Gui, Massey University-Palmerston North, New Zealand
- Qilian Liang, University of Texas at Arlington, USA
- Yves Louet, SUPELEC, France
- David Lozano, Telefonica Investigacion y Desarrollo (R&D), Spain
- D. Manivannan (Mani), University of Kentucky - Lexington, USA
- Jyrki Penttinen, Nokia Siemens Networks - Madrid, Spain / Helsinki University of Technology, Finland
- Radu Stoleru, Texas A&M University, USA
- Jose Villalon, University of Castilla La Mancha, Spain
- Natalija Vlajic, York University, Canada
- Xinbing Wang, Shanghai Jiaotong University, China
- Qishi Wu, University of Memphis, USA
- Ossama Younis, Telcordia Technologies, USA

Sensors

- Saied Abedi, Fujitsu Laboratories of Europe LTD. (FLE)-Middlesex, UK
- Habib M. Ammari, Hofstra University, USA
- Steven Corroy, University of Aachen, Germany
- Zhen Liu, Nokia Research – Palo Alto, USA
- Winston KG Seah, Institute for Infocomm Research (Member of A*STAR), Singapore
- Peter Soreanu, Braude College of Engineering - Karmiel, Israel

Underwater Technologies

Energy Optimization

Mesh Networks

Centric Technologies

Multimedia

- Atsushi Koike, KDDI R&D Labs, Japan
- Chung-Sheng Li, IBM Thomas J. Watson Research Center, USA
- Parag S. Mogre, Technische Universitat Darmstadt, Germany
- Eric Pardede, La Trobe University, Australia
- Justin Zhan, Carnegie Mellon University, USA

Additional reviewers

- Yunyue Lin, The University of Memphis, USA

International Journal on Advances in Networks and Services
Volume 4, Numbers 1 & 2, 2011

CONTENTS

On two Routing Mechanisms for Wireless Sensor Networks

Adrian Fr. Kacsó
Computer Science Department
University of Siegen
57068 Siegen, Germany
Email: adrian.kacso@uni-siegen.de

***Abstract*—In this paper we extend our previous implementation of the T-MAC protocol inside the sensor network simulator with a receiver-based routing (RBR) service and we propose and implement several performance optimizations. We investigate the impact of several MAC protocol parameters (listen time, receiver contention window, radio switch time, etc.) on the performance of routing protocols used in resource constrained wireless sensor networks. The main performance criteria we are interested in are the energy consumption (reflected by the active time the node is operational), the throughput and latency of the network in delivering replies to users' requests.**

Simulation results have shown that using the proposed optimizations improve significantly the performance of the RBR. Moreover, we compare the performance of receiver-based routing against the unicast within our implementation of the T-MAC protocol. Although in direct comparison the RBR approach is outperformed by unicast, we show that RBR can be efficiently employed for opportunistic aggregation inside monitoring areas with many sources or in dynamic network scenarios.

***Index Terms*—wireless sensor network, simulation framework, MAC and routing protocols, collisions.**

I. Introduction

A wireless sensor network (WSN) is a communication network consisting of a large number of sensor nodes that are randomly and densely deployed in a geographical area. The nodes operate unattended and are forced to self-organize themselves (in a multihop wireless network) as a result of frequent topology changes (due to node transient failures, addition or depletion) and to adjust their behavior to current network conditions. Each of the distributed nodes in the WSN senses individually the environment and they collaboratively preprocess and communicate the information to a sink.

Typically, a sensor node has limited energy and memory, restricted communication range and computation capabilities. The communication cost is often higher (several orders of magnitude) than the computation cost. For optimizing the communication cost in order to conserve energy, different data-centric routing protocols and in-network processing techniques have been proposed.

In query-driven WSNs, routing protocols determine on which routes messages (query and data) are forwarded between the sink and sources (nodes able to deliver the requested data) using data-centric approaches. In such *data-centric* routing schemes, the destination node of messages is specified by tuples of attribute-value pairs of the data carried inside the packets and not using globally unique identifiers (node addr).

When the distance between source(s) and sink is large, intermediate nodes forward the messages from hop to hop until they reach the intended destination. Selecting the next hop in order to establish a path (to a source or sink) can be either initiated by the sender or delegated to receiver nodes. In the first approach, the sender decides itself by analysing its internal tables where to send the message, whereas in the second approach the sender delegates the decision to all its neighbors, which distributively elect the best receiver. The strategy to select the next hop employs various metrics which allow to find different paths, e.g., energy-efficient, shorter, rapid, reliable paths, depending on the application goals.

Typically, the information collected in a sensor network is highly correlated, yielding a spatial and temporal correlation between successive measurements. Exploiting the data-centricity and the spatial-temporal correlation characteristics allows to apply effective in-network data aggregation techniques, which further improve the energy-efficiency of the communication in WSN. Aggregation can eliminate the inherent redundancy of the raw data collected and, additionally, it reduces the traffic in the network, avoiding in this way congestions and induced collisions.

The paper is an extension of [1] and is structured as follows. Section II presents the state-of-art and the motivation behind designing energy aware protocols for WSNs using cross-layer design. Section III describes the basic approach of receiver-based routing (RBR). Section IV presents the design (using cross-layering) of the RBR service (RBRS) inside our Timeout-MAC (T-MAC) protocol implementation. Section V discusses several optimizations made to RBRS. Section VI illustrates the performance of the RBR service by giving various simulation results and comparing it with unicast. Section gives more comparison results and Section VIII concludes the paper.

II. Related work and objectives

The main impact on the energy consumption of the nodes is given by the MAC protocol and only secondly by the routing strategy. A real energy benefit is achieved when using MAC protocols with an active-sleep regime and/or low duty cycles (such as S-MAC [2], B-MAC [3], T-MAC [4]). Considering the scarce energy, communication and processing resources of WSNs, a joint optimization of the networking layers by employing a cross-layer design is a promising alternative to maximize the network performance, while reducing the global energy consumption.

Many of the current routing protocols are sender initiated [5][6][7], that is, the decision to which neighbor to route the just received message is taken by the sender. The sender maintains some internal neighborhood table (e.g., gradient table or routing table), which is inspected when messages need to be forwarded. Other protocols [8][9][10][11] use the receiver-based approach; in [9][10][11], the receiver contention scheme is used to develop a unified cross-layer protocol and in [8] to build mechanisms that lead to efficient data aggregation without maintaining a structure, namely the Data-Aware Anycast (DAA) and the Randomized Waiting (RW).

The spare energy and processing resources of battery powered sensor nodes require energy efficient communication protocols in order to fulfill the application objectives of WSNs. The use of both cross-layer design techniques [9][11][12] and aggregation [7][8][13] improves the overall network performance in terms of energy conservation.

The use of **cross-layer design** aims optimizing jointly several layers of the communication stack. Since for a resource constrained node strict layering is inappropriate [14] [15], we employ [12] a cross-layer design by allowing exchange of information (mainly) across application, routing, MAC and physical layers in order to optimize them.

Based on the application's requirements, the network topology, source placement and the aggregation function, a near to optimal **aggregation structure** (tree) can be constructed [16]. Various structured aggregation mechanisms (centralized [13][17] or distributed [7]) have been proposed. For query-driven sensor network applications, where several source nodes periodically report data to the sink, structured aggregation mechanisms are well suited, since the traffic pattern lasts for a long time and the overhead of construction and maintenance of the structure is low, compared with the energy benefits achieved through aggregation. For sensor network applications, where the sources are spread or the network topology is dynamic, the high construction and maintainance overhead for the aggregation structure can outweight the benefits of data aggregation. In such dynamic scenarios, mechanisms are required that achieve data aggregation without the construction and maintenance of a structure.

Concerning the simulator, we proposed in [18] and extended in [12] a modular, energy-aware network architecture of a sensor node as a flexible approach to design and plug-and-play various protocols at network and MAC layers, and to combine and analyse the impact of different parameters on the performance and lifetime of the WSN. We implemented our simulation framework SNF (Sensor Network Framework) using the OMNeT++ 3.4b2 discrete event simulation package and its Mobility Framework [19].

In the present paper we focus on the implementation of an additional RBR service to T-MAC for enabling applications to use both the unicast and the RBR service. An example of such an application is the opportunistic aggregation, where data packets are aggregated, if they meet each other on some node. Inside the source area the data packets are aggregated using the RBR service, while outside it the aggregated data packets are sent using *RTS/CTS* unicast ([20]). Source nodes having matching data (same type and required timestamp) are potential aggregators. If there is a potential additional aggregator closer to sink, it gets a higher priority in the RBR-associated transmission than an aggregator that is farther away.

III. Receiver-based routing (RBR)

The RBR service employs the use of *BRTS* (Broadcast Request-To-Send) control packet to get *BCTS* (Broadcast Clear-To-Send) responses from neighbors, which take initiative to participate in the transfer of the relevant information to sink. The *BRTS* control packet serves as a negotiation between the sender and all its potential receivers. After receiving the *BRTS*, each node determines (according to the information carried in the packet), wherever it participates in the transfer. In order to route a packet to destination the next hop should be *more appropriate* than the sender. Since there are several potential receivers, one needs to separate these receivers in priority groups, according to the available and propagated routing information. Nodes that achieve an *increasing progress* (i.e., are better placed or have more energy or data packets to aggregate, etc.) are placed in a higher priority group than others. The priority of a receiver node (i.e., its priority group) is established by the routing component (and communicated through the cross-layer to the MAC) and is based on the progress a packet would made if the node forwarded the packet. This prioritization is introduced to avoid *BCTS*-collisions (as more receivers may try to respond simultaneously). It is performed by a receiver contention mechanism to access the channel and is actually a computation of a random delay for the *BCTS*.

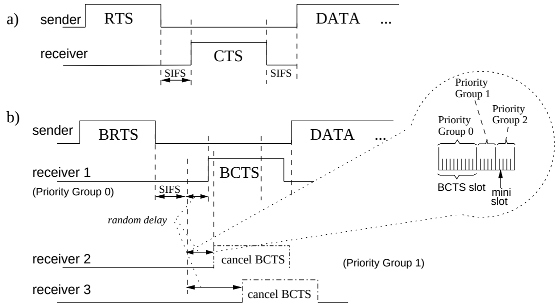


Fig. 1. a) Unicast (using RTS/CTS handshake) b) RBR contention.

Figure 1 shows the difference between the sender initiated next-hop selection (using *RTS/CTS*) and the randomized *BCTS* generation. According to which priority group j the node belongs, it waits for $\sum_{i=0}^{j-1} CW_{pG_i} + cw_j$, where CW_{pG_i} is the contention window corresponding to priority group i ($j \leq n-1$, assuming n priority groups) and $cw_j \in [0, CW_{pG_j}]$ is the delay time corresponding to j. This waiting scheme differentiates nodes of different progress into different priority groups, and attempts to assign different delays to nodes inside the same priority group. The node getting the smallest delay wins the contention and sends a *BCTS* packet to the sender of the *BRTS*. If during the receiver contention, potential receivers hear a *BCTS*, they conclude that a node (with a shorter receiver contention) has accepted to forward the packet. Nodes that overhear a *BCTS* can switch to sleep state. However, in the case of *BCTS* collision (of nodes inside the same priority group) special attention should be paid (see §IV). When the sender receives the *BCTS* packet from the receiver that won the contention, it concludes that the receiver contention ended and sends a *DATA* packet to the intended receiver. Both *BCTS* and *DATA* packets indicate the other contending receivers the sender-receiver pair and the duration of the transmission (the latter only in the *DATA* packet). If the sender node does not receive a *BCTS* packet after $\sum_{i=0}^{n-1} CW_{pG_i}$, it resends the *BRTS* in order to restart the transmission. More details will be given in §IV. Finally, the receiver acknowledges the transmission with an *ACK* packet.

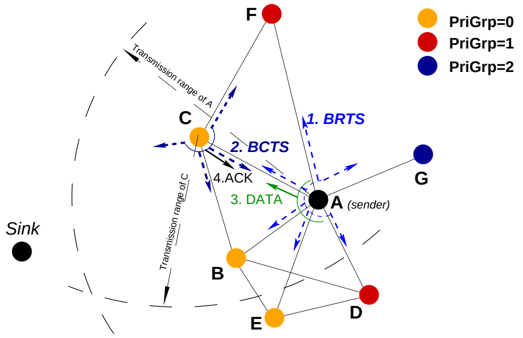


Fig. 2. The exchange of messages for a transfer between node A and C.

Assuming the neighborhood given in Fig. 2 the RBR algorithm is briefly described below and is illustrated in Figure 3.

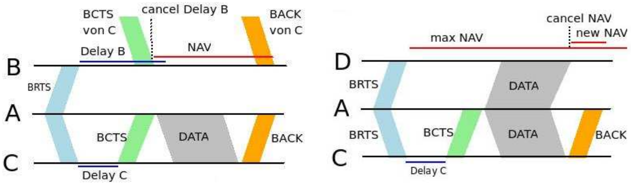


Fig. 3. Node A is the sender: **(a)** Node B and C compete for the reception **(b)** Node D remains quiet and adjusts its NAV-timer upon *DATA* reception.

1) Node A sends a *BRTS* with routing information.
2) Nodes B,C and D receive *BRTS* and compute the priority group (at network layer). More appropriate receivers calculate a lower priority (B and C), unsuitable receivers (only D) are passive (NAV) (see IV-A).
3) Receivers compute a *random time delay* according to the *RCW* of their priority group. Receivers (B and C) of the same priority group compete for the reception (see Figure 3(a)).
4) After expiration of the delay the receiver C (assume that C computed a lower time delay than B) sends a *BCTS*.
5) Potential receivers (B) who receive *BCTS* cancel their receiver contention and go passive (see Figure 3(a)).
6) A sends *DATA* to the *intended* receiver (C). Nodes still in receiver contention, which didn't overhear the *BCTS* (neighbors of A, but not of C, e.g., node E in Fig. 2) but are hearing the *DATA* will go passive. Passive nodes (including D) adjust their NAV timer (see Fig. 3(b)).

In which way receiver nodes are elected in different priority groups is a routing decision, which a node takes according to its local routing information. For example, when the routing uses geographic coordinates, the sender sends in the *BRTS* the sink and its local coordinates. Having this information, a potential receiver determines if it is closer to sink and correspondingly the node becomes a member of one of the predefined priority groups. The same principle is used when routing metrics as hop count or combinations of hop count and residual energy of nodes are used. Moreover, we may include in the priority groups some criteria to promote aggregation (see VII-3).

IV. T-MAC WITH RECEIVER-BASED ROUTING SERVICE

The T-MAC protocol uses a synchronized schedule in which nodes follow a listen-sleep regime. The main states of the protocol are illustrated in Figure 4. All nodes start in the `Startup` state by setting randomly a local timer and listening the channel. Each node switches to `Active Startup` state as soon as its own timer has expired or a foreign SYNC message has been received. At the end of the `Active Startup` state the node is synchronized and switches into *Active-Sleep* regime. In `Active Own` state the node has its own schedule during which it can receive and transmit. The protocol states for a unicast communication are illustrated in Figure 5.

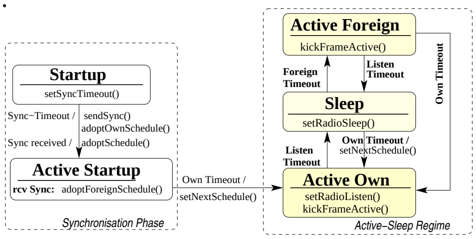


Fig. 4. Main states of T-MAC protocol.

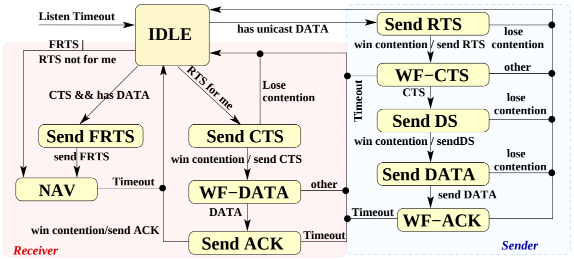


Fig. 5. T-MAC protocol state diagram for unicast (*Active Own* state).

The above states are almost self-explanatory and are common to RTS/CTS handshake mechanism ([20]). The reason to send data packets using the Request-To-Send (RTS) and Clear-To-Send (CTS) control packets is to reduce collisions, when two or more nodes transmit near the same time (hidden-station problem). This handshake mechanism is useful when the data packets are long, since if the packets collide, they are discarded, the energy is wasted and a later retransmission requires additional energy consumption both at sender and receiver. Broadcast packets are never sent using the RTS/CTS handshake and are not acknowledged (using ACK packets).

In case of `Active Foreign`, the node is in active state of a foreign schedule, where it can only receive. Therefore, the state transitions are the same as in the left side of Figure 5. The Future-Request-To-Send (FRTS) packet is an extension meant to avoid the early sleeping problem ([4]).

We extend our previous implementation of T-MAC with the receiver-based routing service (RBRS). The protocol states for the receiver-based routing (RBR) are illustrated separately in Figure 6 for clarity reasons. The entry point for both state diagrams is the `IDLE` state, from where either the RTS/CTS handshake or the RBR service can be used.

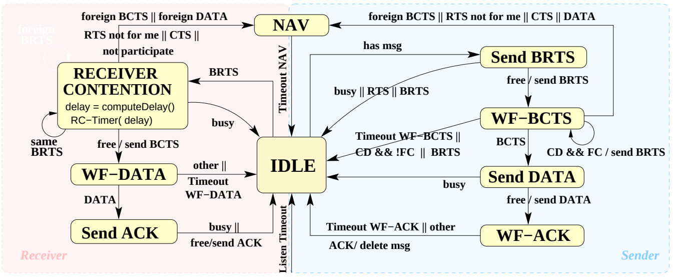


Fig. 6. T-MAC protocol state diagram for RBR (*Active Own* state).

In the following, we explain the protocol by describing the conditions and actions of the state diagram. Not mentioned terms/conditions and actions are self-explanatory. Later, we focus on particularities and optimizations.

We used the following abbreviations for conditions:

CD: collision detected between *BCTS* packets
FC: first collision between *BCTS* packets
same BRTS: reception of a previous *BRTS* packet (resent)
foreign X: reception of a packet *X* with different destination
other: reception of another message as expected
busy or *free*: channel is used or not during carrier sense
not participate: not a potential receiver (see *computeDelay*).

Moreover, we employ the action *computeDelay*, which calculates the receiver contention (*RC*) time to set the node's corresponding timer (`RC-Timer`) until it sends the *BCTS*.

Concerning the behavior of the T-MAC protocol, there are some aspects that need special attention. The first one is to customize the T-MAC's active period. The active period of a node ends, if during the timeout activation (TA) it does not detect any activity (e.g., an incoming packet, collision); then the node goes to sleep. Otherwise, if the node overhears or starts a communication, it schedules again a timeout after this communication finishes. The timeout value is set to stretch a small contention window and an *RTS-CTS* packets exchange. Hence, collisions and the resulted retransmissions extend the node's active time and increase, therefore, the energy consumption. The second important aspect is related to retransmission. Hereby, as the nodes are synchronized and we want to avoid them sending simultaneously after waking up, we need a large enough contention time before retransmitting. Note that a relatively large contention time helps avoiding collisions, but it also extends the active time of the node.

A. Cross-layer implementation of RBR

Receiver-based routing describes a communication method, in which each node has the choice to participate (or not) in the communication. In classic layer based protocols the communication is initiated by the application layer and the message passes the complete protocol stack, starting with the application layer and forwarded by each subsequent layers, network, data link and physical layers at sender, while on the receiving end it is forwarded by the physical layer in the opposite direction to upper layers until it reaches the receiver application layer.

According to the local information at sender, the routing protocols of the network layer decide to which neighbor node to forward the message. However, in case of receiver-based routing, the routing decision is not taken at the sender. Instead, the sender initializes the communication and potential receivers compete for the reception. The winner becomes the intended receiver for the sender. By shifting the routing decision to the receiver node, it is possible to use information for the routing decision, which is not local to the sender.

Due to the relocation of the decision, strictly speaking, the sequential forwarding of the message through the layers cannot be met. The sender initializes the communication with a message *BRTS*, which contains all the relevant routing information decision. Each node that receives the message will analyze this information and decide if it is a potential receiver. This decision is made at the network layer of the receiver. Thus, the network layer of the receiver is already involved in the communication before the actual data flow.

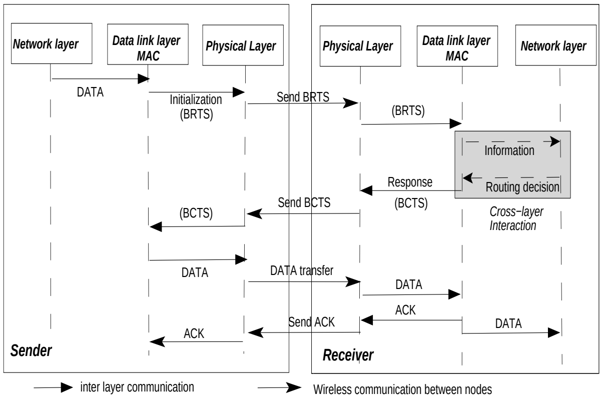


Fig. 7. Packet flow using the receiver based routing service.

When sending the reply all potential recipients are competing. The response time of the receiver depends on the decision of the network layer. The receiver which responds first (with a *BCTS*), becomes the communication partner for the sender in the current transfer.

As shown in Figure 7, in RBR mainly cooperate the data link layer and network layer in the potential receiver with each other. Therefore, the receiver-based routing application can be divided into different components, which can be assigned to the individual layers:

- *T-MAC RBR service*: service on the link layer for the actual communication. Realizes the communication between nodes, but does not make any decisions.

- *Routing Unit*: service at the network layer. Handles communication with the MAC layer in the context of decision making.
- *Strategy Unit:* implements a routing strategy on the network layer. Take the actual decision on the basis of the transmitted routing information.

Figure 8 illustrates the message flow until the first *BRTS* packet is sent by the sender.

1) The application layer of a source node generates a DATA packet and sends it to the network layer. In the control information of the DATA packet there is also the interest identifier (*iID*) necessary to map the corresponding routing information stored at nodes. This step is omitted at relay nodes.
2) The routing unit calls the strategy unit assigned, which writes the required routing information (related to the received iID) in the dynamic part of the network header.
3) The network layer use a special target address (*L2RBR*) to signal the link layer to use the RBR service.
4) The RBR service from T-MAC copies the contents of the dynamic part of the network header in the dynamic header part of *BRTS* packet.
5) The link layer sends the *BRTS* packet as broadcast.

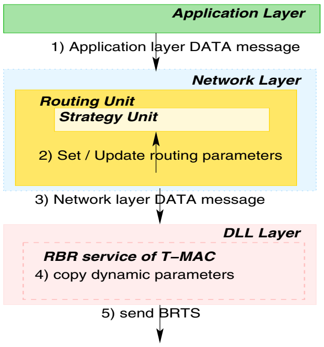

Application Layer
1) Application layer DATA message
Network Layer
Routing Unit
Strategy Unit
2) Set / Update routing parameters
3) Network layer DATA message
DLL Layer
RBR service of T–MAC
4) copy dynamic parameters
5) send BRTS

Fig. 8. Flow of messages (at sender) until the first *BRTS* is sent

Figure 9 shows the sequence flow of operations at the receiver after receiving a first *BRTS* message. Since the RBR service inside T-MAC must be flexible, in order to be able to process various RBR-strategies, at the first BRTS reception the communication between the RBR service and the Routing Unit and its associated Strategy Unit must be initialized using a cross-layer component. This is mandatory since the type and number of routing information parameters depends on the used strategy/strategies. Accordingly, it changes the number of parameters for the decision function call.

Figure 9 shows the sequence of steps until a BCTS response packet is sent.

1) The T-MAC RBR service receives the first *BRTS* packet.
2) The RBR service reads the dynamic part of the *BRTS* header and registers the individual parameters in the cross-layer. Only for the last parameter the notification service of the cross-layer is activated. This parameter is used in future receptions of a *BRTS* packet to trigger the call of strategy function (the actual routing decision) at the network level.
3) The RBR service stores the values of all routing parameters inside the cross-layer.
4) The number of parameters is registered in the cross-layer. This information type was registered at the initialization in the cross-layer with active notification. The Routing Unit has been registered as a subscriber. This information type serves as a trigger for the registration function of the routing unit to subscribe for actual routing information parameters.

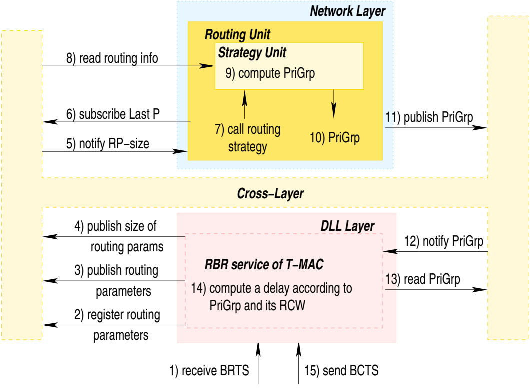


Fig. 9. Cooperation between the components at a receiver node upon reception of the first BRTS packet.

5) The notification service of the cross-layer informs the routing unit that the information type for the number of routing information parameter has been updated (and for subsequent receptions that the routing parameters are updated).
6) The registration function subscribes itself to be notified for updates of the routing information parameters (the last parameter update triggers the notification).
7) The registration function calls explicitly the strategy function, since this is not automatically called by the first update of the routing information at the first *BRTS* reception.
8) The strategy function reads the routing information parameters from the cross-layer.
9) The strategy function calculates the priority group (*PriGrp*) according to the received routing information parameters.
10) The priority group is passed to the routing unit.
11) The routing unit publishes the computed priority group in the cross-layer. This information type was registered at initialization and the cross-layer has registered itself as a subscriber.
12) The RBRS is notified about the updating of the information type for the priority group.
13) The RBRS reads the priority group from the cross-layer.
14) Using the calculated priority group the RBR service computes the delay for its *BCTS* packet. (If the node does not participate in the communication, it skips in the NAV state.)
15) When the timer expires the node sends a *BCTS* response if during the delay no *BCTS* or a *DATA* packet was received.

At subsequent receptions of *BRTS* the handling is analog, excepting the steps: the routing information parameters are already registered and the routing unit has subscribed to be notified when the routing information is updated, i.e., the call of the strategy function is triggered automatically.

V. RBR OPTIMIZATIONS

To design an effective receiver-based service implies to avoid collisions whenever possible and, if they still occur, to handle them efficiently. To that end, we propose in the following several optimizations.

A. First Group Weight optimization

Potential receivers compete for reception only within the same priority group. Each priority group has an own *receiver contention window (RCW)*. The smaller the *RCW* is, the higher is the probability that collisions occur. Collisions within the highest priority group have the largest negative impact on the performance of the RBR-service. In order to extend the receiver's contention window for the highest priority group (to reduce the likelihood of collisions) we provide an optimization referred as *First Group Weight*. The weight of the first *RCW* is set through a configuration file. For a larger *RCW* there will be fewer collisions, but the average duration of a data transmission extends also. The weighting should reflect the density of the network, i.e., it must scale with the number of neighboring nodes. For example, for a network with 5-7 neighbors we set the weight for the highest priority group to 40% and the rest of 60% is equally divided between the remaining groups (see Figure 10).

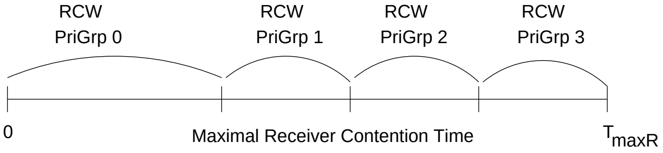


Fig. 10. Division of the T_{maxRC} for four priority groups.

Knowing the maximal neighbors density of a node one can analytically determine the minimal *RCW* size, such that the probability of no collisions has a given value p_{no_coll}. The *RCW* is given by

$$RCW = t_{sw}\sqrt[k]{\frac{k\sum_{i=1}^{n-1} i^{k-1}}{p_{no\_coll}}},$$

where t_{sw} is the switching time of the transceiver. The number of slots is given by $\frac{RCW}{t_{sw}}$.

B. Early Resend optimization

Potential receivers check after their own *receiver contention* expiration whether the medium is free. Note that the nodes have a single-channel radio, i.e., they are not full-duplex and require a switching time between the transmit and receive mode. Even though the medium is checked before each transmission, the switching time and the finite speed of radio waves propagation may lead to collisions.

The denser a network is, the more potential receivers compete for reception, which increases the probability of *BCTS*-collisions. Collisions of *BCTS* packets have a negative impact on the performance of the RBR-service, since the data transmission needs to be re-initialized. The retransmission includes the initial contention of the *BRTS* packet. In addition, nodes that heard the *BRTS* packet or one of the two collided *BCTS* packets go in *NAV* state. Since during this time the medium is not used, the throughput decreases while the latency and the power consumption increase. The optimization *Early Resend* ensures a faster recovering after *BCTS*-collisions by repeating the receiver contention process as soon as possible.

To that aims, after a collision of *BCTS* packets, a new *BRTS* packet is sent without an initial contention period. Nodes that have caused the collision don't notice instantly, since the single channel radio has a relatively long switching time from send to receive. These nodes require a short *WF-DATA* timeout until they reach the `IDLE` state (the *WF-DATA* timeout and the delay to resend *BRTS* are small compared to the average contention time). Nodes that observe the collision break their receiver contention and go to `IDLE` state. Since the nodes that received only one of the two collided *BCTS* packets are in `NAV` state, all neighboring nodes of the sender are either in `IDLE` or `NAV` state when the *BRTS* packet is resent; thus, the risk of a collision does not increase significantly. Usually, nodes in the `NAV` state are passive and do not respond to a *BRTS* message, excepting a retransmission of the *BRTS*. The nodes detect that they received a copy of the *BRTS* and start a priority group calculation. After the retransmission of the *BRTS*, all neighbors of the sender start a new priority group calculation and possibly a new receiver contention. The sender remains in state `WF-BCTS` (see sender state diagram in Figure 6). When a second collision occurs, the sender transits in `IDLE` state and restarts the communication completely from the scratch.

The scope of the *Early Resend* optimization is to recover faster after *BCTS*-collisions by skipping the initial contention at sender. By omitting the initial contention time of the *BRTS* the risk of a collision does not increase significantly since adjacent sender's nodes are either in the `NAV` or just switched into the `IDLE` state.

C. Change Priority-Group optimization

Depending on the topology of the network and the strategies applied (since the receiver priority group is computed by a routing strategy), it may happen that a sender cannot find an optimal receiver. Getting a non-optimal priority group at all potential receivers means a long *RC* time, which leads to a higher latency of the transmission and a higher energy consumption, as the active phase of T-MAC is extended. The optimization aims to prevent this by raising the group priority of all potential receivers, until at least one belongs to the highest priority group. This is achieved through the interest ID (*iID*) and *flag* fields inside the header of the RBR-service messages. The *iID* is necessary to map the data to the given interest (request). The one byte flag field (see Figure 11) is divided into a 1-bit field used in the *BCTS* response to notify the sender that the receiver has raised its priority group, and a 7-bit field for the value of the decrease in the priority group (in the *BRTS*) or the current computed priority group (in the *BCTS*). A potential receiver sends in the *BCTS* its current adjusted priority group (*PriGrp*).

Upon receiving the *BCTS* response, the sender verifies the priority group. If this does not correspond to the optimal *PriGrp*, it increments a counter for the specified *iID*. If the counter reaches a threshold (specified in a configuration file), at the next transmission of a data message for the same interest, the sender sets in the flag the required decrease (a multiple of the threshold) in order to raise the *PriGrp* of all potential receivers. The potential receivers read the flag from the sender's *BRTS* message and, if the value is greater than zero, they raise their own *PriGrp* with the given value. Receivers send in their *BCTS* response the new *PriGrp* and the flag that indicates that they have raised their priority group.

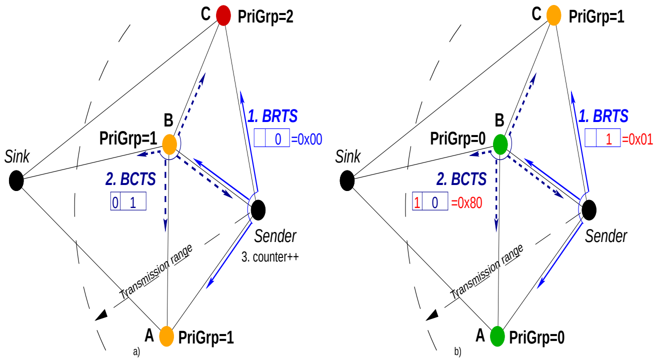


Fig. 11. Change Priority-Group: **(a) 1:** Sender sends *BRTS* **2:** Node B computes *PriGrp* 1 and since no node has the smallest priority group, it wins the *RC* and sends *BCTS* with its *PriGrp*. **3:** Sender increments a local counter for not optimal *BCTS* response. **(b) 1:** Counter reaches threshold: sender sends *BRTS* with request to adjust the *PriGrp*. **2:** Receivers increase their *PriGrp*. B wins the *RC* and sends *BCTS* by setting the first bit, i.e., it has raised the priority, and its current computed *PriGrp* (flag = 0x80).

If during the priority increase optimization a new potential receiver is added to the neighborhood of the sender, the new node computes a better priority group that the optimum. This receiver will not set the 1-bit field in the *BCTS* response, notifying the sender that it has computed the highest priority group, without using the priority group increase request. In addition, the receiver contention is reduced by half time, so it is likely that this node wins the contention and this receiver can send its *BCTS* response. If the sender receives a *BCTS* with no flag set, it resets the counter for the corresponding interest. For the next transmission the sender cancels its request for raising the priority group of all its potential receivers.

VI. Performance Evaluation

For the evaluation of the RBR service, we analyze the following performance parameters: active time (which highly impacts on the energy consumption), throughput and latency.

Simulator settings: in the simulation we used the Chipcon CC1000 (used by MiCA2) and CC2420 (used by Telos) single channel radio transceivers with the following parameters:

	current [mA]			**power** [mW]			
	SL	RX	TX	SL	RX	TX	Switch
CC1000	0.11	10	8.3	0.33	30	33	25
CC2420	0.02	24	14	0.04	48	28	30

	switching time [μs]				
Transceiver	*SL→RX*	*SL→TX*	*RX,TX→SL*	*RX→TX*	*TX→RX*
CC1000	850	850	10	850	850
CC2420	580	580	10	580	580

For T-MAC we set the listen time to 30ms and the frame time to 600ms. The overhearing avoidance flag is disabled.

A. Active time

The active time of a node significantly influences its energy consumption. Activities in the node's neighborhood extend the node's active time, since they reset the active timeout. For the measurements we used a multihop sensor network with m hops between source and destination, and a variable neighborhood density of n neighbors (see Figure 12). The source generates data packets at each 200ms, and the simulation time is 1 minute. To minimize the effects of subsequent transfers in the first measurement we chose $m = 1$.

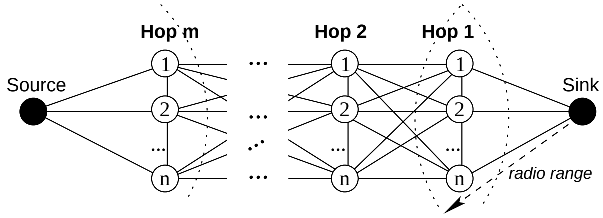


Fig. 12. Simulation scenario.

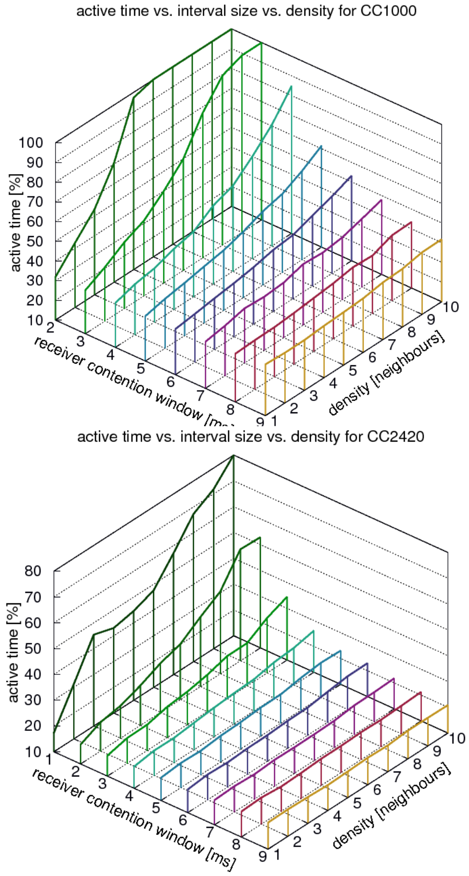


Fig. 13. Impact of the network's density and *RCW* size on active time.

Figure 13 shows the effects of the density of the neighbors and the size of the *receiver contention window* (*RCW*) on the active period of both transmitters.

The CC1000 transceiver has a much higher active time than the CC2420 transceiver. The difference cannot be explained only by the higher transmission rate (250 kbps compared to 76.8 kbps), since the proportion of time in which the transceiver is in transmitting mode is approximately 2%. Rather, it seems likely that additional causes generate this behavior. To investigate this closer one needs to analyse the number of events that occur during a transfer. Events that negatively affect the behavior and extend the active period are collisions of messages and their consequences.

For small *RCW*, the active time increases quickly with increasing of the network's density. This leads to frequent collisions of *BCTS* responses, whereby the receiver contention needs to be repeated. Therefore, the active timeout is set again. For large *RCW*, the active time remains relatively constant. The negative impact on the active time by a long-lasting transmission (due to the large *RC*) will be compensated by rare occurrence of *BCTS*-collisions.

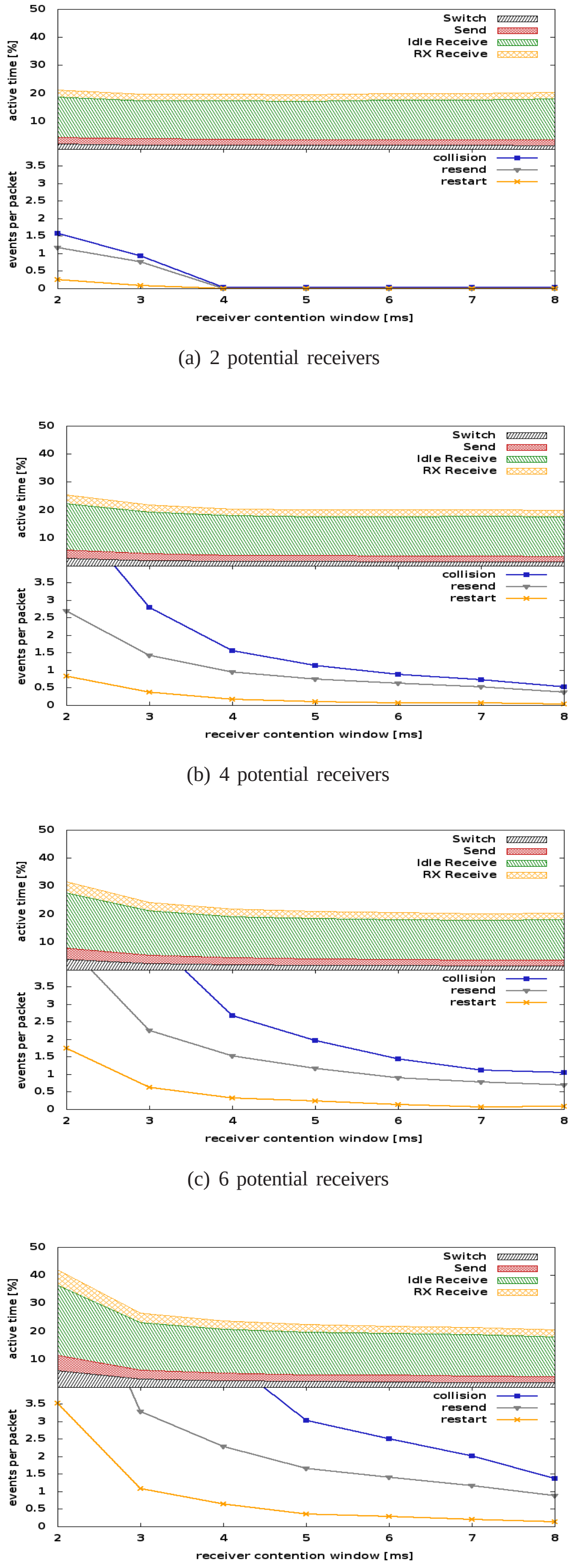
Next we investigate the possible causes for an extended active time. The measurements in Figure 14 show the influence of the *BCTS*-collisions on the active period of the CC2420 transmitter. Each graph shows the number of events (per packet) occurring (lower part) and the active time of the transceiver (upper part) according to various *RCW* sizes and different neighbors densities.

A *BCTS*-collision occurs mainly due to the fact that the difference of two or more calculated receiver contention times is smaller than the transmitter's switch time. Using a single channel transmitter, a potential receiver node is able to check the channel for activity until its transmitter switches from receive to send. It is possible, that during switching another node starts to transmit its *BCTS*. The first node cannot notice that and, therefore, the length of the *RCW*, especially the difference between two receiver contentions is important. The shorter the switching time of the transmitter is, the smaller the difference between two receiver contentions can be. That means, the smaller the switching time of the transmitter is, the more opportunities have other nodes to compute a receiver contention that does not lead to a collision.

During a transfer, the *resend event* occurs at the first collision of the *BCTS* responses. This event is triggered by the *Early Resend* optimization, when after the first *BCTS* collision, a new *BRTS* is sent without contention (see V-B) . Since the initial contention time is omitted, this event has a relatively small influence on the extension of the active time compared to restart the transmission. If a second *BCTS*-collision happens, the whole transfer must be restarted, including the initial contention. In graphs this corresponds to the *restart event*, which has a much higher impact on the active period, since it increases the fraction of time that the radio spent in *Idle Receive* state as part of the whole active time.

Hence, for increasing neighbor density and large *RCW*, the active time remains relatively constant, despite the increase of the number of negative events, since if a transfer was successful, it is likely that a low delay time has won the contention. The number of retransmissions will be higher, but the transfers are in average completed faster and this compensates partially the negative effect.

In case of the CC1000 transmitter the same measurements lead to a larger number than for CC2420 (figures are omitted here). This is due to the fact that the CC1000 has larger switching time, which increases the frequency of *BCTS*-collisions. A



(a) 2 potential receivers

(b) 4 potential receivers

(c) 6 potential receivers

(d) 8 potential receivers

Fig. 14. CC2420: Impact of *RCW* size on the active period of the transmitter (reflected by the procentage of each active radio state (sleep is what left until 100%) for different density of neighbors. Each graph gives also the number of relevant events (collision, resend, restart) for different size of *RCW*.

larger switching time means that during a node is checking the medium and switching to send the probability that another node (during this time) starts to transmit is higher and, thus, more *BCTS*-collisions occur.

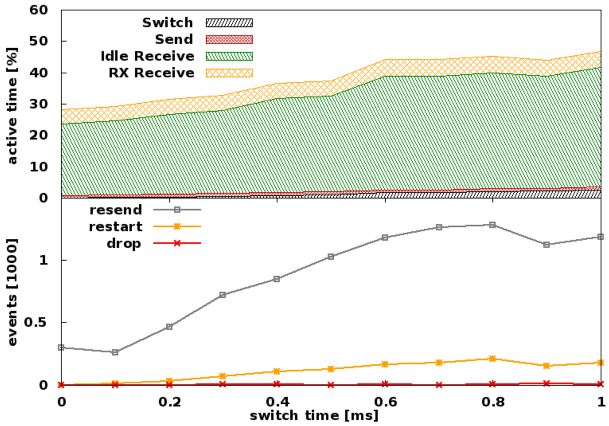


Fig. 15. Impact of the switch time on the active time.

Figure 15 shows the impact of the switch time on the active time for a fixed *RCW* (4ms) and a given network density. For smaller switching times, the active time decreases. By increasing the switching time, the active time increases also, but more than the sum of the individual switching times. The cause is the higher number of *BCTS*-collisions when the switch time increases.

B. Throughput and latency

In order to measure the maximum possible throughput and the source-sink latency in dense network, we set m=5 and n=3.

Figure 16 shows the drop rate and the latency for different *RCW* sizes. In low traffic networks the latency is independent of *RCW*. With increasing data rate increases also the latency and its variance. This occurs rather for small than for large *RCW*. An increasing latency leads also to packet loss, as can be seen in Figure 16.

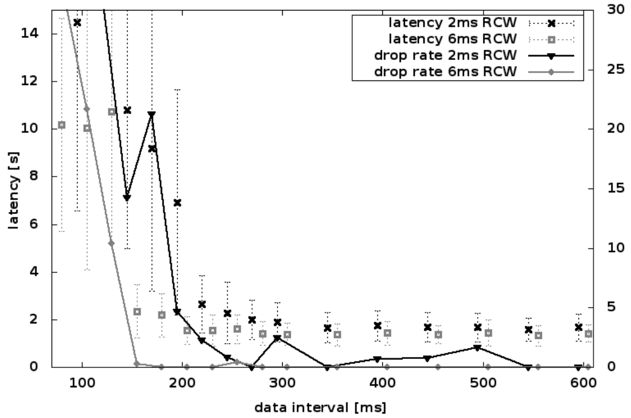


Fig. 16. Throughput and latency for *RCW* of $2ms$ and $6ms$.

The reason lies in the interaction of different transfers within a region. The RBRS uses instead of the SIFS (Short Interframe Space) between the *RTS* and *CTS* control packets a receiver contention, which defers the *BCTS* response. As a potential receiver is in the receiver's contention, a node in the neighborhood, which has not received the original *BRTS*, can start itself a transmission by sending a *BRTS* packet. If this is the case, only one of the two transmissions can be successfully completed, assuming that no collision has happened. The hidden-station problem cannot be effectively solved by the RBRS. If a collision occurs, both transfers must be re-started. These two cases occur more often when the data rate increases. Additionally, the *BCTS* collisions mentioned in the previous measurements occur very frequently in small *RCW*. If the number of retransmissions exceeds a given threshold, the packets are deleted.

VII. Comparisons and Evaluations

In order to compare the energy savings achieved through the proposed optimizations we consider here the sensor network given in Figure 17.

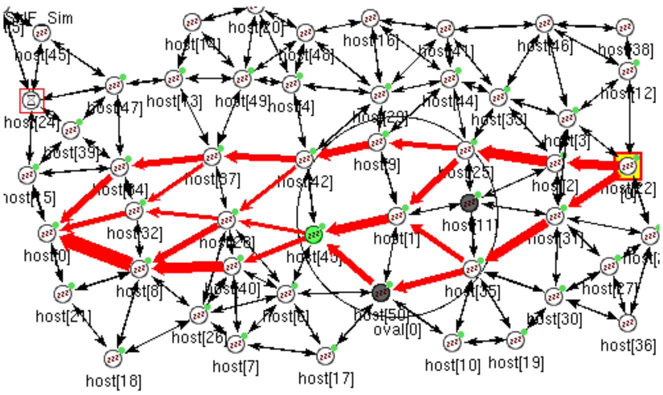


Fig. 17. A simulation network with 51 sensor nodes.

For this network the source is node 22 situated on the right side of the figure and the sink is node 0 situated on the left side of the figure, where all the red arrows end.

1) Comparison RBR with and without optimizations: In order to compare the energy savings achieved through the proposed optimizations, we have enabled and disabled the optimizations.

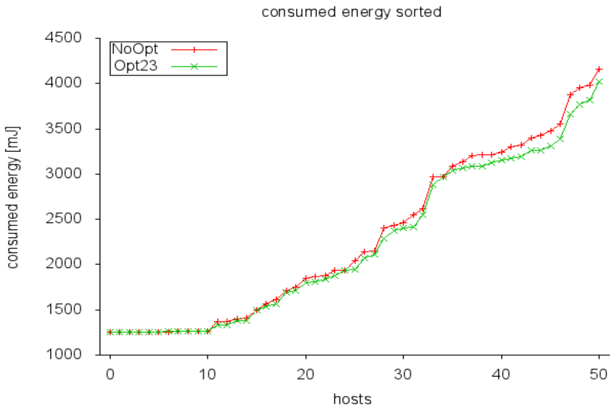


Fig. 18. Energy consumption using RBR with and without optimizations.

The comparison of the energy consumed in both cases for a simulation time of 3 minutes considering three priority groups and a weight for the highest priority group of 60%) is illustrated in Figure 18. One can observe that with all three optimizations the energy consumed by some nodes improves up to 7%, but there are also nodes where the energy consumption increases. The overall energy consumption is reduced when using optimizations by at least 4%.

2) Comparison RBR with unicast: Next we compare the RBR service with the unicast in our variant of T-MAC [1]. We set up two scenarios, one for the RBR and one for the unicast; both use the same application and network layer. At network layer, the routing information is propagated through interest refreshes without extra traffic for routing. For data routing we use in both cases a strategy based on hop count and node's residual energy. The simulation time is 3 minutes, the source generates data packets at each $200ms$ and the *RCW* for the RBR-service is $4ms$. For the unicast scenario the overhearing avoidance flag inside the T-MAC is enabled, meaning that nodes in the NAV state turn off their radio to conserve energy.

Figure 19 illustrates the energy consumption for the RBR-service and unicast respectively.

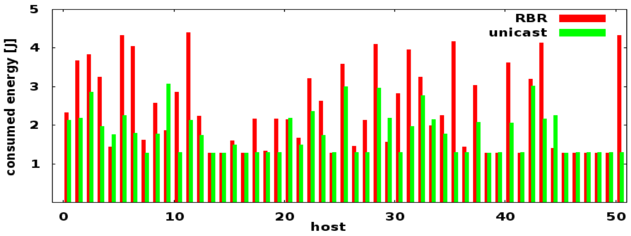
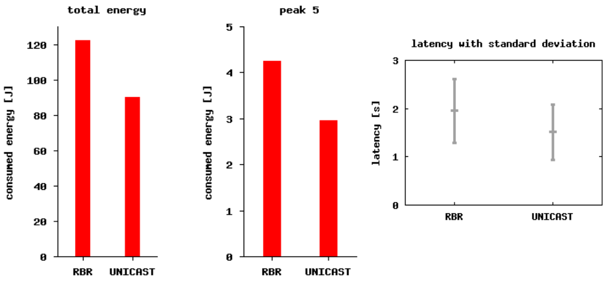


Fig. 19. Energy consumption for unicast and RBR services of T-MAC.

The comparison of energy consumption shows a higher energy consumption when using the RBR service. The reason is the additional receiver contention of the RBRS and the *BCTS*-collisions, since both increase the active time of a node. In case of unicast, since the hidden station problem is successfully solved, there are fewer adverse events and a transfer can be faster terminated.

Using the same routing strategy, the total energy consumption for unicast is $91J$ in comparison to $123J$ for RBR (see Figure 20(a)). The situation remains similar when we compare the energy consumed by the five most heavily loaded nodes (see Figure 20(b)).



(a) total (b) peak five (c) latency

Fig. 20. Energy consumption: **(a)** total energy consumption **(b)** average energy consumption of the five highest loaded nodes. **(c)** latency

Comparing the source-sink latency in the two cases we got an average of $2s$ for RBR and $1.5s$ for unicast. Here too, the lower latency of the unicast is due to the fact that in the RBR case the additional receiver contention increases the transfer time per hop.

Thus, under these settings, the unicast outperforms the RBR service. We discuss in the sequel a scenario, where the situation can be different.

3) Networks with both modes enabled: We consider a wireless sensor network with many sources placed in a closer area; here aggregation of the sensor data is necessary inside the source area in order to significantly reduce the amount of data traffic (otherwise each source establishes individual paths to the sink, leading also to increased energy consumption and, thus, to a shorter lifetime of the network).

The aggregation inside the monitoring area can be realized either by constructing an aggregation tree using unicast or by employing the RBR service. The RBR mechanism allows to route data packets in this area without maintaining information about the next hop, and to aggregate without the construction and maintenance of an aggregation structure. On the other side, when using unicast a significant overhead (in terms of communication cost to spread the information about the network topology) occurs for construction and maintenance of cache tables and aggregation structures.

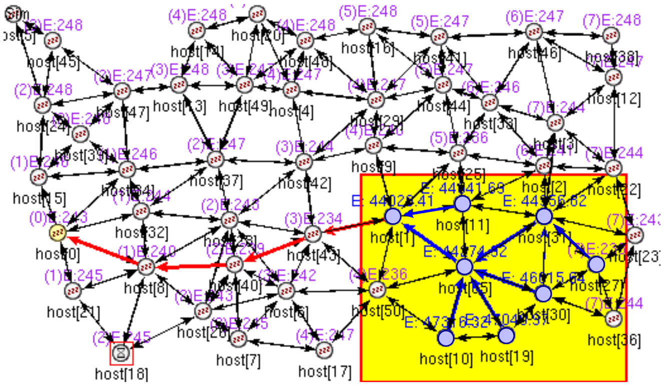


Fig. 21. Simulation network using both *RBR*-service and unicast.

For simulations we use our SNF with both unicast and RBR service enabled inside T-MAC protocol implementation and consider the 51 nodes network illustrated in Figure 21. It has 8 sources situated in the buttom-right corner and a sink (node 0) placed on the left side of the figure. Two source nodes (50 and 36) situated in the rectangle area are not equipped with the required sensors, i.e., they cannot deliver the requested data packets. We set the simulation time to $3min$, the data generation interval $200ms$ and each source sends 750 packets.

We discuss two scenarios: one using unicast and the other using RBR for aggregation inside the observed area (the yellow rectangle); outside the area the aggregated data packets are always sent using unicast.

The strategy used by unicast can be, for example, a simple hop count strategy as illustrated in Figure 21 (see red line).

In this case, outside the observed area the path used to reach the sink is always the same and the energy consumption of the nodes along the path remains the same.

Of course, in order to balance the total energy consumption among the nodes between the source zone and the sink, one can use an energy-aware strategy, e.g., a routing metric based on hop count and the path's residual energy which leads to more paths to sink (see the red lines in Figure 22). This strategy redistributes the traffic load uniformly and avoids bottleneck nodes on the path to sink with less residual energy.

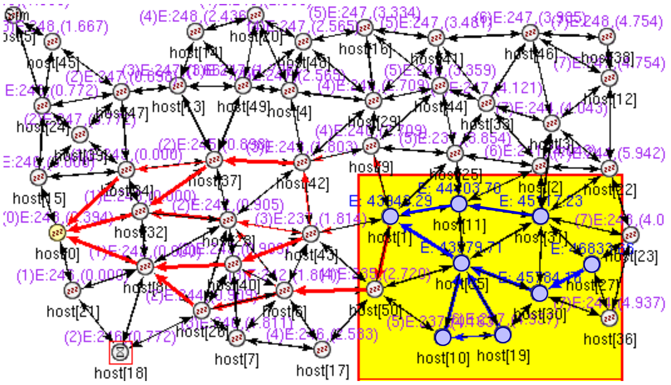


Fig. 22. Using a energy-aware routing strategy between sources and sink.

In the unicast scenario (regardless of the chosen routing metrics outside the monitoring area), in order to be able to aggregate data, an aggregation algorithm is needed to construct the aggregation structure (tree); source nodes having matching data are potential aggregators.

One possibility to construct an aggregation tree is to use the same mechanism as for flooding the interest. When an interest reaches the first source, this one initiates an aggregation interest which is flooded only inside the source region. This leads to a local greedy aggregation tree rooted at the first source.

Another alternative is to design an own aggregation protocol. When a source gets the first interest, it starts to find eligible (best) aggregation nodes in the zone. This can be achieved by sending an invitation to other sources to be aggregated, and the algorithm works as follows:

- The invitation control packet ($CAN\_I\_AGG\_YOU$) contains the sender id and information about the aggregation possibilities of the sender such as its distance to the sink, energy reserve, number of sources surrounding it (optionally their aggregator), its connectivity, etc. The invitation control packet is rebroadcasted by each source (to include farther sources).
- When a source receives the invitation it checks if it is already aggregated by another node or is an aggregator itself. If yes, the control packet is discarded. Otherwise, the receiver decides according to its local information and the received information if it is a better (closer to sink or has more sources, etc.) aggregator than the sender. When the receiver source accepts to be aggregated, it sends a confirmation control packet (YES_AGG_ME). The confirmation packet must be acknowledged (ACK_AGG) by the aggregator.
- Finally, each source knows which node is its aggregator. This information can be broadcasted to the neighbors (for more reliability) using a notify control packet ($I\_AM\_AGG\_BY$).

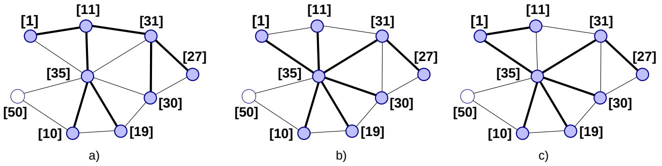


Fig. 23. (a), (b) Aggregation trees for unicast; (c) *RBR*-service

Using the latter aggregation algorithm, the resulted aggregation tree can be one of the trees illustrated in Figure 23 a) and b). In the first case we have three intermediate aggregators, nodes 11, 31 and 35 and in the second case only two, nodes 31 and 35. In the above simulation the establishment of the aggregation structure is not repeated periodically, but is realized only once.

In the RBR scenario, the aggregation is achieved without additional communication by including aggregation criteria in the definition of the priority groups, namely

- **Group 0**: Receivers having DATA packets with the same type and being closer to the sink than the sender.
- **Group 1**: Receivers having DATA packets with the same type, but farther away from the sink than the sender.
- **Group 2**: Receivers without same type of DATA, but closer to the sink.
- All receivers not belonging to one of the groups do not participate in the communication.

As a potential intermediate aggregator closer to sink gets a higher priority than an aggregator farther away, the aggregation is opportunistic. (For the considered network the aggregation flow is illustrated in Figure 23(c))

For these two scenarios, we compared the total energy consumption, the source to sink latency and the throughput. It turns out that all three performance criteria are quite similar for both scenarios.

In case of unicast, each aggregation node waits until all its children sent their data, then aggregates these and sends the result data to its parent. This increases the node's active time in the unicast scenario and reaches the same level as for the RBR (induced by its higher receiver contention).

So for both scenarios we get this time almost the same source to sink latency, throughput and total energy consumption.

Hence, for such applications the unicast does not outperform the RBR service even if the aggregation tree is constructed only once (as we considered in our simulations). Obviously, for highly dynamic networks or networks with longer activity, the aggregation structure needs to be reestablished periodically, which finally leads to weaker overall performance.

VIII. Conclusion and future work

In the present paper, we strive for more modularity at MAC layer, mainly to embed at this layer more customizable services. We supplement here the sender initiated unicast by a receiver-based contention in order to provide another perspective to the interlayer communication. The RBR service of T-MAC allows (reactive) applications, in which a sender does not know its potential destination or applications with a dynamic network topology where the construction and maintainance overhead for the cache tables and/or aggregation structures is expensive (in terms of communication cost to spread the information about the network toplogy). In such dynamic scenarios the RBR mechanism allows to route, without maintaining information about the next hop, and to aggregate, without the construction and maintenance of an aggregation structure.

The accurate energy model integrated in our simulator allows us to quantify the impact of transceiver's switch time, the *RCW* and the occurrence of collisions and their retransmissions on the energy consumption of the sensor node. The possible collisions of the *BCTS* responses and their consequences must be minimized (using transceivers with smaller switching time) for a good performance of the RBR service. Therefore, we proposed and implemented several optimizations of the RBRS. Simulation results have shown that these optimizations improve significantly the performance of the RBR. We have analyzed the performance parameters of the RBR service, namely its energy-efficiency, throughput and latency. Moreover, we compared (VII-2) the efficiency of both forwarding approaches inside T-MAC: the sender initiated one using unicast versus the receiver-based routing. For our simulation scenario it turned out that the RBR approach outperformed by unicast in terms of energy consumption, throughput and latency. Nevertheless, the RBR can be efficiently employed for opportunistic aggregation inside monitoring areas with many sources or in dynamic network scenarios (VII-3); here the routing performance of RBR and unicast are similar.

As future work we intend to build in our simulator different simulation scenarios in order to closer investigate and compare the performance of RBR versus different aggregation algorithms using unicast.

References

[1] F. Kacsó and U. Schipper, "Receiver-based routing service for t-mac protocol," in *Proc. 4th Int. Conf. on on Sensor Technologies and Applications (SENSORCOMM 2010)*. Venice/Mestre, Italy, July 2010, pp. 489–494.

[2] W. Ye, J. Heidemann, and D. Estrin, "Medium access control with coordinated, adaptive sleeping for wireless sensor networks," *IEEE/ACM Trans. on Netw.*, vol. 12, no. 3, pp. 493–506, 2004.

[3] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in *Proc. 2nd ACM Int. Conf. on Embedded Networked SenSys*. NY, USA, November 2004, pp. 95–107.

[4] T. Dam and K. Langendoen, "An adaptive energy-efficient mac protocol for wireless sensor networks," in *Proc. 1st Int. Conf. on Embedded Networked SenSys*. LA, California, USA, 2003, pp. 171–180.

[5] M. Busse, T. Hänselmann, and W. Effelsberg, "Energy-efficient forwarding schemes for wireless sensor networks," in *Proc. Int. Symp. on WoWMoM*. New York, USA, June 2006, pp. 125–133.

[6] F. Ye, G. Zhong, S. Lu, and L. Zhang, "Gradient broadcast: A robust data delivery protocol for large scale sensor networks," *Wireless Networks/Springer, The Netherlands*, vol. 11, no. 2, pp. 285–298, 2005.

[7] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva, "Directed diffusion for wireless sensor networking," *IEEE/ACM Transactions on Networking*, vol. 11, no. 1, pp. 2–16, 2003.

[8] K.-W. Fan, S. Liu, and P. Sinha, "Structure-free data aggregation in sensor networks," *IEEE Trans. Mob. Comput*, vol. 6, pp. 929–942, 2007.

[9] I. Akyildiz, M. Vuran, and O. Aka, "A cross-layer protocol for wireless sensor networks," in *Proc. CISS*. Princeton, NJ, March 2006.

[10] T. Watteyne, A. Bachir, M. Dohler, D. Barthel, and I. Aug-Blum, "1-hopmac: An energy-efficient mac protocol for avoiding 1-hop neighborhood knowledge," *Sensor and Ad Hoc Communications and Networks*, vol. 2, pp. 639–644, Sept 2006.

[11] P. Skraba, H. Aghajan, and A. Bahai, "Cross-layer optimization for high density sensor networks: Distributed passive routing decisions," in *Proc. Ad-Hoc Now04*. Vancouver, July 2004, pp. 266–279.

[12] F. Kacsó and R. Wismüller, "A simulation framework for energy-aware wireless sensor network protocols," in *Proc. 18th Int. Conf. on Computer Communications and Networks (ICCCN'09), Workshop on Sensor Networks*. San Francisco, CA, USA, August 2009, pp. 1–7.

[13] J. Wong, R. Jafari, and M. Potkonjak, "Gateway placement for latency and efficient data aggregation," in *Proc. 29th Annual IEEE Int. Conf. on Local Computer Networks*, Nov 2004, pp. 490–497.

[14] J. Polastre, J. Hui, P. Levis, J. Zhao, D. Culler, S. Shenker, and I. Stoica, "A unifying link abstraction for wireless sensor networks," in *Proc. 3rd ACM Int. Conf. SenSys*, November 2005, pp. 76–89.

[15] C. Ee, R. Fonseca, S. Kim, D. Moon, A. Tavakoli, D. Culler, S. Shenker, and I. Stoica, "A modular network layer for sensornets," in *Proc. 7th Symp. OSDI*. Seattle, WA, USA, 2006, pp. 249–262.

[16] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion a scalable and robust communication paradigm for sensor networks," in *Proc. ACM MobiCom*. Boston, 2000, pp. 56–67.

[17] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proc. 33rd Annual Hawaii Int. Conf. on System Sciences (HICSS'00)*, 2000, pp. 3005–3014.

[18] A. Kacsó and R. Wismüller, "A framework architecture to simulate energy-aware routing protocols in wireless sensor networks," in *Proc. IASTED Int. Conf. on Sensor Networks*. Greece, 2008, pp. 77–82.

[19] M. Löbbers and D. Willkomm, *Mobility Framework for OMNeT++ (API ref.)*. http://mobility-fw.sourceforge.net: OMNeT++ Ver.3.2, 2006.

[20] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "Macaw: A media access protocol for wireless lans," in *Proc. of SIGCOMM Conf.* London, UK, September 1994, pp. 212–225.

GeoOLSR: Extension of OLSR to support Geocasting in Mobile Ad Hoc Networks

Volker Köster, Andreas Lewandowski, Dennis Dorn and Christian Wietfeld
Communication Networks Institute (CNI)
TU Dortmund University, Germany
Email: {Volker.Koester|Andreas.Lewandowski|Dennis.Dorn|Christian.Wietfeld}@tu-dortmund.de

***Abstract*—Safety critical applications of IEEE802.15.4 networks require autonomous network reconfiguration and dynamic meshing in case of node failures or changing environmental influences. This paper demonstrates the application of Ad hoc On Demand Distance Vector (AODV) and Optimized Link State Routing (OLSR) on IEEE802.15.4 nodes based on an IP layer. The modular concept leads to the proposed extension of Optimized Link State Routing (OLSR) protocol to provide location-based services and addressing inherently in the protocol design. We demonstrate the changes in message flows and information exchange that are necessary to develop a geo-implementation of OLSR, which we call GeoOLSR throughout this paper. For performance evaluation, we will first examine mobile ad hoc relevant metrics like time delay, maximum throughput and generated overhead. Furthermore, the lifetime of the novel node architecture is evaluated in comparison to the ordinary IEEE802.15.4 configuration and IEEE802.11. Finally, the real-world protocol behavior of GeoOLSR is shown for different mobility speeds by using realistic ray tracing for modeling the physical transmission in a harsh industrial environment. Thus, it is proven by results that GeoOLSR is able to support both IP enabled unicast traffic and geographical addressing even in resource constrained networks like IEEE802.15.4.**

***Keywords*-Geocasting; OLSR; AODV; IEEE802.15.4; Ray Tracing.**

I. Introduction

The application of wireless sensor networks (WSNs) or mobile ad hoc networks (MANETs) based on IEEE802.15.4 in safety-critical processes requires a fault tolerant network design, which supports autonomous reconfiguration [1]. Recent developments in the area of Wi-Fi networks propose meshing algorithms on ISO/OSI layer 3 like the reactive Ad hoc On demand Distance Vector (AODV) Routing [2] or the proactive Optimized Link State Routing (OLSR) [3], which are capable to update communication paths in case of failures and mobility of network nodes. In contrast to that, original IEEE802.15.4 networks rely on topologies like star or cluster tree, in which failures of single nodes can isolate even complete network trails.

Therefore, the IEEE task group *802.15.5 Mesh Networking* currently examines necessary mechanisms that are designed for the physical (PHY) and medium access control (MAC) layer. In order to enable a flexible network setup for different application scenarios within heterogeneous networks, we analyze multihop forwarding via IP routing mechanisms (on layer 3 – also known as *Route-over* [4]). We propose a node design (see Figure 1), which includes an IPv4 layer – instead of utilizing the ZigBee protocol stack [5]. The integration of the IP protocol for WSNs is proposed in [4] and [6]. The big advantage of these approaches is the seamless integration into the Internet. In [7], it has been shown that the implementation of a tiny TCP/IP protocol is feasible for the integration on low power devices, such as IEEE802.15.4 without major changes of the PHY- or MAC layer. Following these approaches, this paper demonstrates the application of OLSR and AODV routing schemes as proposed in [8] by applying a peer-to-peer network topology. Here, every node is assumed to operate as a router and uses CSMA/CA channel access.

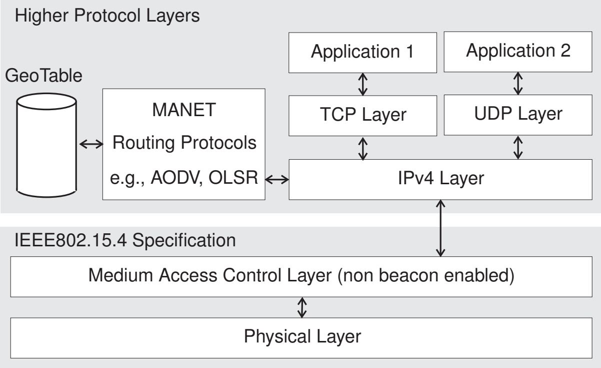


Figure 1. The extended Layer Model of our proposed architecture to support IP enabled unicast traffic and geographical addressing in IEEE802.15.4

Besides an easy integration of IEEE802.15.4 nodes into preexisting infrastructures, diversified application domains are one key performance indicator of MANETs. Thus, there is an increasing need for a simultaneous support of geographical addressing to realize e.g., location based messaging and alarming. As an extension to [1], it is an objective of this paper to show an extension of OLSR – which will be called GeoOLSR throughout this thesis – to build up a routing protocol, which supports unicast (IP-based) as well as geographical multicast communication inherently. Therefore, a slight modification to the original protocol architecture is made by adding an additional routing table (called GeoTable), which contains positions of reachable nodes (cf. Figure 1). The major benefit of using a proactive routing protocol like OLSR is the periodical exchange of routing information in discrete time intervals. This enables knowledge of a node's last position even in case of malfunctions. In contrast to that, on-demand routing mechanisms only search for a new route prior a specific communication request.

The work is part of a research project with one of the world's largest steel fabricants ThyssenKrupp Steel. They will install the presented solution to increase the security of the factory employees in case of emergency. The developed solution is integrated in a gas sensor network, which consists of stationary and mobile equipment. Hence, not only factory employees, but also first responders profit from this solution, as they do not have to carry additional devices for navigating through the incident scene. Thus, this work presents several major contributions:

- Demonstration of the general applicability of meshed network approaches within IEEE802.15.4 networks by implementing our IP enabled sensor node architecture based on the physical layer of IEEE802.15.4 using a peer-to-peer enabled CSMA/CA MAC layer.
- Introduction of a detailed performance evaluation of AODV and OLSR in IEEE802.15.4 networks.
- Proposal of an OLSR extension, which enables geocasting as well as IP-based unicast messages combined with high node mobility support.
- Comparison of different geocast routing protocols e.g., Location Based Multicast (LBM), flooding GRID, ticket GRID and GeoTORA with GeoOLSR.
- Evaluation of the influence of different moving speeds and patterns on GeoOLSR.
- Brief identification of the resulting overheads of GeoOLSR compared to OLSR.
- Analysis of the proposed GeoOLSR protocol in a real-world scenario considering realistic radio channel effects by application of the Actix Radiowave Propagation Simulator (RPS) [9], which includes a high-precision 3-dimensional CAD drawing of the application scenario within the steel production plant.

This paper is organized as follows. Section II discusses related works. Afterwards we demonstrate the design of the new node in Section III in detail, before the implementation of the applied simulation model in OMNeT++ 4.0 is shown as well as details of the simulation measurements in Section IV. After that, we illustrate the protocol extension of GeoOLSR within Section V by presenting necessary changes in message flows and information exchange to realize a geo-implementation of OLSR, followed by corresponding analysis in Section VI. Performance evaluations via OMNeT++ simulation together with a sophisticated PHY layer model based on the ray tracing tool RPS are presented in Section VII. Finally, Section VIII draws conclusions.

II. Background and Related Work

In this Section, we will give a brief introduction into state-of-the-art routing protocols divided into four groups – unicast MANET protocols, geographical multicast protocols, mobile agents protocols and hierarchical routing protocols.

A. *Unicast Mobile Ad-Hoc Network Protocols*

Linking an IP address with a geographical location has been of interest for quite some time already. On the other hand, there has also been significant research to increase network redundancy in general, based on unicast routing protocols for MANETs, in which all mobile hosts typically behave as routers. A route between a pair of nodes in a MANET may go through several other mobile nodes. Due to the mesh network approach these routes may vary when nodes change their locations. Many attempts have been made on MANET protocols [2], [3], [10]. There are two major types of networking protocols defined in the literature for this application field [11]:

- *Proactive routing:* A node manages the whole network topology in a periodically updated routing table, which causes additional traffic.
- *Reactive routing:* The route is determined when a packet has to be transmitted. Hence, the delay for a single packet transmission is higher in comparison to proactive routing; however, the additional traffic for route maintenance is minimized.

In the following paragraphs, basic principles of OLSR as a proactive and AODV as a reactive routing scheme are described in detail.

OLSR

The Optimized Link State Routing is specified in the RFC 3626 [3]. Simulative and experimental performance evaluation on Wi-Fi devices is presented in [11]. *Route table calculation* is done by topology information, which is gathered from topology control messages (TCM). If a node generates its neighbor list, the TCMs are transmitted through the network. A node is defined as a neighbor, if a bi-directional physical connection between two nodes is available. Following RFC 3626, OLSR communicates using a unified packet format for all data related to the protocol. This is meant to facilitate extensibility of the protocol without breaking backwards compatibility. This also provides an easy way of piggybacking different "types" of information into a single transmission like geographic data in the field of GeoOLSR. A RFC 3626 standard implementation is embedded in IPv4. The basic layout of any packet in OLSR consists of an OLSR header, which includes three types of messages:

- *OLSR-Hello* To perform link sensing, neighborhood detection and Multi-Point-Relay (MPR) selection, Hello messages are exchanged between 1-hop neighbors periodically. This message is sent as the data-portion of

the general packet format with the ”Message Type” set to HELLO_MESSAGE, the Time-to-live field set to one and Vtime set accordingly to the value of NEIGHB_HOLD_TIME.
- *OLSR-Topology-Control* The link sensing and neighbor detection part of the OLSR protocol basically offers a neighbor list in each node, which contains a list of neighbors to which a direct communication is possible. In combination with the packet format and forwarding mechanism, an optimized flooding through Multi-Point-Relays (MPRs) is implemented. This mechanism is based on the OLSR-Topology-Control (TC) message format, which disseminates topology information through the whole network.
- *OLSR-Multiple-Interface-Declaration* The OLSR-Multiple-Interface-Declaration (MID) message is used to map more than one IP address to one node. Therefore, all interface addresses other than the main address of the originator node are put into the MID message.

The use of multipoint relays (MPRs) reduces the network load by concentrating the traffic on dedicated nodes. The speed of topology update processes can be regulated by varying Hello and TC intervals. The main performance indicators of OLSR are summarized in Table I. The willingness for a MPR is defined by the remaining battery power of the node.

Table I
PARAMETERIZATION OF OLSR NODES

Hello Interval	inter-arrival time of hello packets
Hello Jitter	maximum deviation from the hello interval
TC Interval	inter-arrival time of TC packets
TC Jitter	maximum deviation from the TC interval
Hello Timeout	maximum timeout of hello messages until the node is removed from the neighbor list
Willingness	willingness of a node to act as MPR

To reduce the negative influence of packet losses due to high mobility in OLSR, Benzaid et al. proposed a new method of integrating fast mobility in the OLSR protocol [12].

AODV

The Ad hoc On Demand Distance Vector routing is specified in RFC 3561 [2].

An application for IEEE802.15.4 networks has been proposed by [13] without applying an IP layer. Each node operates as a router and determines point to point connections on demand without periodical updates. Thereby, memory and energy demand is optimized for battery driven mobile devices and the additional network load is minimized. An included sequence number avoids the count-to-infinity routing problem [2]. In contrast to other routing protocols, the quality of a connection is determined by the actuality and not by the length of the path. The main configuration parameters of AODV are summarized in Table II.

Table II
PARAMETERIZATION OF AODV NODES

Active Route Timeout	defines the validness of a route
Hello Interval	defines the inter-arrival time of hello packets
Allowed Hello Loss	defines the maximum hello packet loss until a route is deleted
Delete Period	defines the limit of route from node A and B to D, if node A has deleted the route
Net Diameter	maximum number of hops between two nodes
Node Transversal Time	estimated for a 1-hop transmission
Net Transversal Time	2* Node Transversal Time * Net Diameter
Path Discovery Time	2* Net Transversal Time
RREQ Retries	number of attempts for route determination

B. Geographical Multicast Protocols

In addition to the work mentioned before, there has also been significant work on multicasting based on the location of the particular nodes. Several approaches have been proposed [14] [15]. The schemes for multicasting can be broadly divided into two types: flooding-based schemes and tree-based schemes. Flooding-based schemes (like Location-Based Multicast [16]) do not need to maintain as many network states as tree-based protocols. On the other hand, flooding-based schemes can potentially deliver multicast packets to many nodes that are currently outside the location, which is energetic inefficient. Tree-based schemes (cf. GeoTORA [14] and GeoGrid [15]) reduce the amount of sent messages. However, a higher overhead is needed to maintain the network's tree.

C. Mobile Agents Protocols

Other routing schemes are based on mobile agents and are inspired from social insects' behavior [17]. One of the main ideas of ant algorithms is the indirect communication of a colony of agents, based on so called pheromone trails. Pheromones are used by real ants for communication purposes. The ants know the other ants' paths by the pheromone trails, and the amount of pheromone on a trail reflects its importance.

D. Hierarchical Routing Protocols

Besides the location based routing approach some attempt has been made to support a routing algorithm that integrates geo-coordinate and table-driven IP addressing [18]. This routing protocol called ”GeoLANMAR” uses link-state routing in a local scope and geo-routing for out-of-scope packet forwarding. The protocol keeps track of the routes to destinations up to a certain distance away from the source whereas the geo-routing scheme applied in GeoLANMAR is used to route packets to the remote landmark nodes outside the local scope.

III. Design of an IP-enabled Wireless Sensor Node

The simulation model is implemented in the discrete, event-based network simulator OMNeT++ [19] and the INET framework. Figure 1 shows the implementation of the communication node. The IEEE802.15.4 physical layer implementation [20] of OMNeT++ is used for the proposed extensions. By using the IP layer, also the existing UDP and TCP protocol implementations of the INET framework can be evaluated for new services. An additional 20 byte IP header and an 8 byte UDP header decrease the overall capacity. But 74 byte payload are left, which is enough for sensor monitoring applications and additional services, as the maximum payload size of the messages in this application area is usually inherently small.

In order to highlight the generated overhead in comparison to a conventional IEEE802.15.4 network, the resulting throughput is measured in a simple point-to-point scenario. The new node is operating with AODV in the first case and OLSR in the second case. For system startup, a time off of 10 s is set before mesurement values are captured. The applied parameters of the meshing protocols are summarized in Table III.

Table III
Parameterization of the test scenario

Traffic profile	OLSR settings	AODV settings
74Byte (UDP Payload) every 30ms ≈19.73kbit/s	Willingness = 3 Hello Interval = 1s TC Interval = 2s MID Interval = 2s	Active Route Timeout = 6s Hello Interval = 1s Allowed Hello Loss = 2 Delete Period = 10s Net diameter = 2 Hops Node transversal time = 40ms RREQ Retries = 2

This parameterization is assumed for all following performance evaluations. The values for the hello interval (HI) of OLSR and AODV are chosen equally for an optimal comparison.

Figure 2 depicts the resulting overhead generated by the new implementation. About 10kbit/s overhead must be calculated for the application of IP-based meshing protocols in a simple point-to-point scenario. The following performance evaluation will also clarify the scalability up to an 8 hop scenario.

IV. Evaluation of the novel peer-to-peer approach

To demonstrate general feasibility of the novel architecture, we first evaluate important performance indicators like end-to-end transmission delay, goodput during handover processes, handover delay and achievable throughputs in an OMNeT++ simulation environment. However, typical PHY layer issues like CCA delay [21] or co-channel interferences [22] are neglected at this point. Figure 3 depicts a hidden-station hop-to-hop scenario. The setup consists of 8 hops placed in a hop distance of $d_{max} = 150m$, which represents the maximum radio range. Here, the string topology represents the worst case for OLSR due to the fact that the MPR forwarding becomes obsolete.

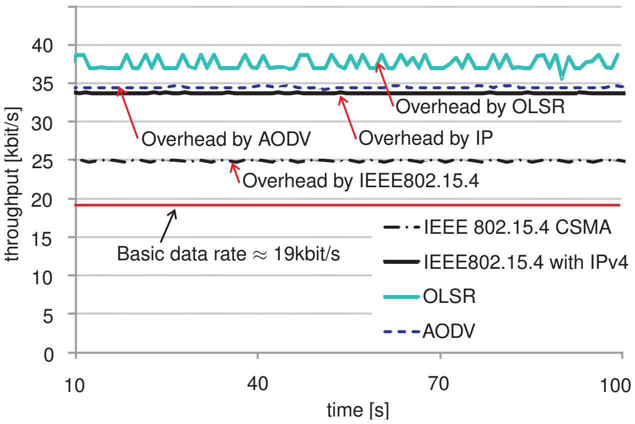


Figure 2. Comparison of network load between IEEE802.15.4 and the peer-to-peer implementation with applied AODV and OLSR routing algorithm for a simple point-to-point setup without mobility. The parameters of Table III are applied.

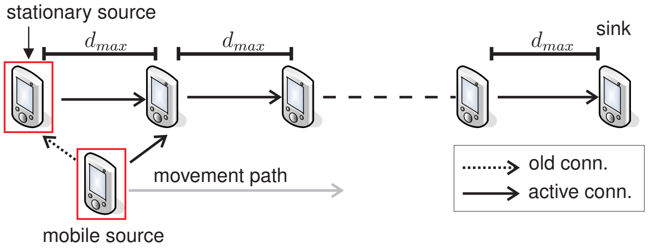


Figure 3. Hop-to-hop Scenario for Performance Evaluation

In each test, 74Byte packets (payload) are sent over the network from the stationary source each 30ms until the mobile node reaches the end of the playground. The performance evaluation is then structured as follows. First, we analyze the end-to-end delay in stationary node constellations before an analysis of handover scenarios between fixed network nodes and mobile nodes is achieved. Finally, the energy consumption is compared to an IEEE802.15.4 node implementation.

A. Evaluation of End-to-End Transmission Delay

The end-to-end delay is a good indicator to measure the response behavior and the real-time capability of the network. The parameterization of this experiment is described in Table III. The test is repeated 100 times, before the distribution function is calculated (cf. Equation 1) to determine the $\mu \pm 2\sigma$ interval, which includes 95.4 % of all possible end-to-end delay values.

$$F(x) = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{x} \mathrm{e}^{-\frac{1}{2}\left(\frac{t-\mu}{\sigma}\right)^2} dt \qquad (1)$$

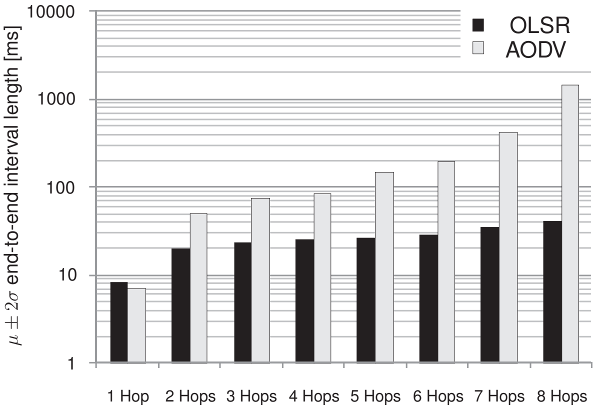


Figure 4. Interval length of the end-to-end delay for OLSR and AODV depending on the hop count for a stationary scenario depicted in Figure 3. The interval contains $\mu \pm 2\sigma$.

The results are assembled in Figure 4. The expected value for end-to-end delay of AODV is lower compared to OLSR in the one hop case, but in all other cases OLSR seems to be predominant. As a consequence, AODV exhibits an interval of [0 ms, 1435.1 ms] for the 8 hop case, which means that 95.4 % of the examined cases fit into this interval, whereas OLSR features an interval of only [0 ms, 41.5 ms]. The high delay of AODV can be explained by the route determination process. With an *Active Route Timeout* of 6s and a *Delete Period* of 10s, routes are updated frequently assuming constant bitrate (CBR) traffic. The needed additional traffic for the route determination process rises with an increasing number of hops.

B. Performance Evaluation of Hand-Over Processes

As mobile sensors are regarded for typical application scenarios, fast handover processes are needed for reliable measurement transmission. The following experiments base on the measurement setup shown in Figure 3 with one moving source node transmitting data continuously (cf. Table III) over the next fixed node to the sink at a predefined constant speed for the mobility.

Figure 5 depicts the achievable goodput at different mobility speeds and hop counts for AODV. The reference line at 0 m/s shows the impact of the hop count on the maximum goodput. It can be seen that the throughput is almost constant until a hop count of 4. This finding correlates to the end-to-end delay for AODV depicted in Figure 4. A higher hop count decreases the achievable throughput, as the delay is nearly doubled from 4 Hops to 5 Hops. The same observation can be made for moving nodes. Here, the impact increases with higher mobility speeds, which is caused by the switching time of the accomplished handover processes.

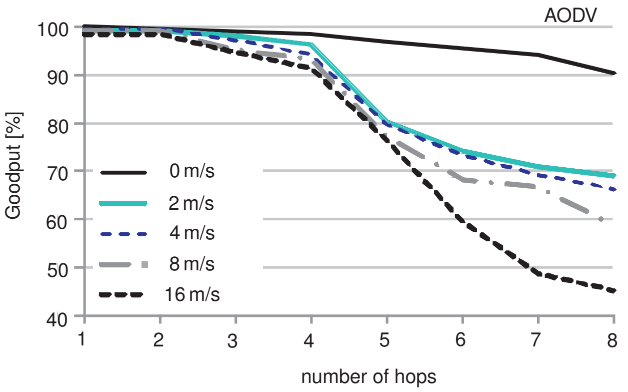


Figure 5. Goodput in percent for AODV depending on the hop distance to the sink with a mobile source transmitting to the next stationary node in range, which then forwards the information to the sink.

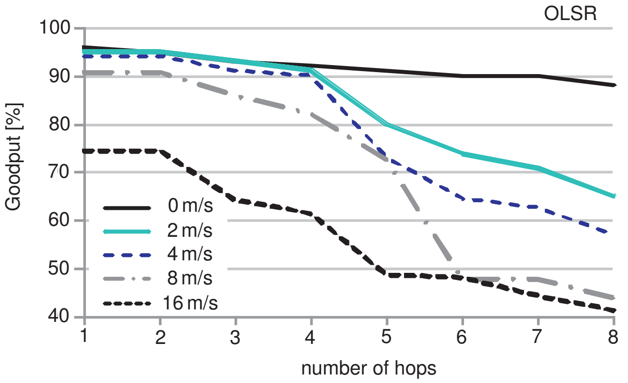


Figure 6. Goodput in percent of OLSR depending on the hop distance to the sink with a mobile source transmitting to the next stationary node in range, which forwards the information to the sink. Whilst performing route updates, the traffic is interrupted.

For comparison of the performance of OLSR and AODV, Figure 6 depicts the achievable goodput for OLSR in the same network and measurement setup. OLSR starts at a lower goodput for the reference measurement at 0 m/s, as OLSR gathers – as a proactive routing scheme – the routing information for the entire network in advance, which takes about 6 seconds for this setup before the data transmission can start. As a consequence, higher overhead decreases the achievable goodput. Due to continuous traffic for route updates, the probability of collisions between OLSR control and data packets rises with the number of intermediate hops. As a consequence, the goodput decreases with a higher number of hops between source and sink.

The handover process itself decreases the goodput. A tradeoff between HI, which causes additional traffic (cf. Figure 2) and switching time for the handover has to be determined. Figure 7 depicts the OLSR handover delay for

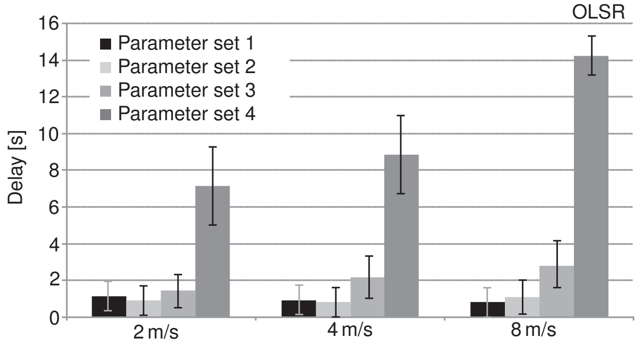


Figure 7. OLSR Handover delay for different mobility speeds and network configurations. *Parameter set 1:* Hello=1s, TC=1s, MID=1s; *Parameter set 2:* Hello=1s, TC=2s, MID=2s; *Parameter set 3:* Hello=2s, TC=5s, MID=5s; *Parameter set 4:* Hello=5s, TC=5s, MID=5s;

different network configurations and mobility speeds. It can be observed, that the main performance indicator is the hello interval (HI). As the HI is small, high speeds are supported by the network. As the HI is enlarged (e.g., to reduce the traffic overhead), the handover delay rises. This finding is comparable to a Wi-Fi study on OLSR [23], where the hello interval is described as the main performance parameter.

Analyzing performance related parameters of OLSR and AODV has been subject of many papers in recent years [23] [24]. However, each publication assumed IEEE 802.11 as the physical and data link layer protocols. To ensure a good comparability of our measurement results with the preexisting ones, we analyzed the mean values of the average throughputs and their standard deviations at varying speeds and parameter sets for both OLSR and AODV. Here, we let the mobile sink of Figure 3 move towards and away from the destination node and calculated the mean throughput for the whole distance. The results are depicted in Figure 8.

As expected, AODV outperforms OLSR in terms of mobility support due to periodical route maintenance of the pro-active routing algorithm. Considering the relative high throughput of our measurements and the shorter coverage areas of IEEE802.15.4, one can conclude that both findings are nearly congruent. In [23] the decrease of average throughput between a node speed of 0 m/s and 15 m/s varies from 25 % (HI = 1s, TC = 5s) to 26 % (HI = 2s, TC = 5s), whereas our simulative results show a difference of 28.59 % (HI = 1s, TC = 2s) and 25.97 % (HI = 2s, TC = 5s) respectively. The results for AODV comparison behaves equally, concluding that AODV still ensures a delivery ratio of more than 90 % even at high speeds for hop distances of up to 8 hops.

C. Energy consumption of meshed sensor nodes

Energy consumption is a critical issue for the design of wireless sensor nodes. IEEE802.15.4 standard divides the network in node classes, where *routers* and *coordinators* are always switched on for maintaining connection between nodes. The *end device* is the node class, which is designed for transmitting sensor information. It operates with low energy consumption due to sleep phases and is only connected to a coordinator or cluster head. If the next higher node in hierarchy fails, the end device will be isolated from the rest of the network.

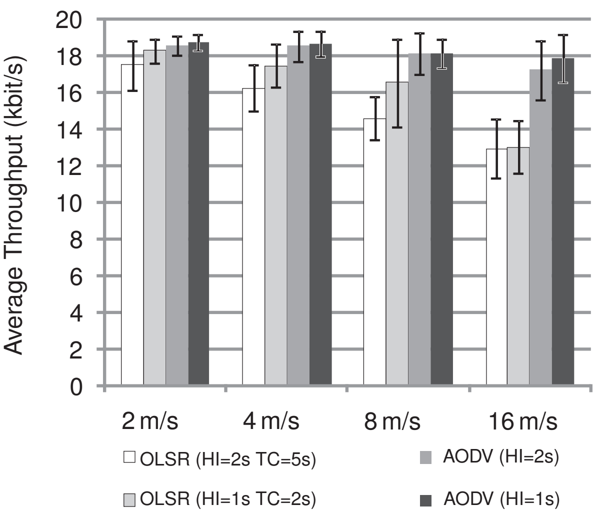


Figure 8. Comparison of the average throughputs of OLSR and AODV at different speeds

Table IV
Parameterization of Battery Model for TI CC2420 (IEEE802.15.4) and MAX2822 (IEEE802.11b)

	TI CC2420 (Pout = 0dBm)	MAX2822 (Pout = +3dBm)
Supply Voltage	3V	3V
Standby-Mode Supply Current	1.38mA	25mA
Receive-Mode Supply Current	9.6mA	80mA
Transmit-Mode Supply Current	16.24mA	98mA
Rx Sensitivity	-95dBm	-85dBm

The applied battery model of OMNeT++ utilizes the parameterization shown in Table IV based on the data sheets of the *TI CC2420* [25] transceiver for IEEE802.15.4 and the *MAX2822* [26] for IEEE802.11b. Adaptive bit rate adjustment and changing power levels are neglected in this study; only worst case assumptions are evaluated, which means that always a transmit power of 0 dBm is applied for the CC2420 transceiver. Nevertheless, the parameter d_{max} is adjusted for maximum transmission range for Wi-Fi (250 m) and IEEE802.15.4 (150 m) respectively.

Following this parameterization, Figure 9 shows the energy consumption of applied AODV and OLSR in comparison to a regular IEEE802.15.4 end device and IEEE 802.11b.

Following this parameterization, Table V shows the simulated battery lifetimes of AODV and OLSR in comparison to a regular IEEE802.15.4 end device and IEEE 802.11b, which also applies both AODV and OLSR. We found that

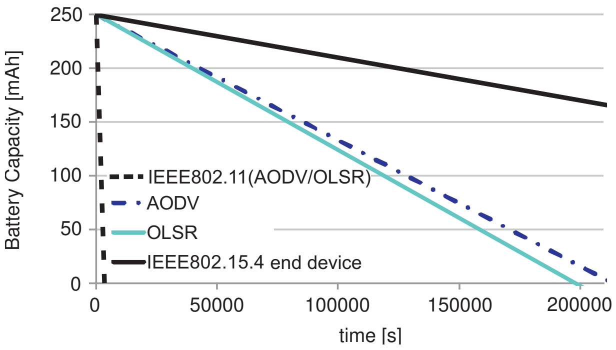
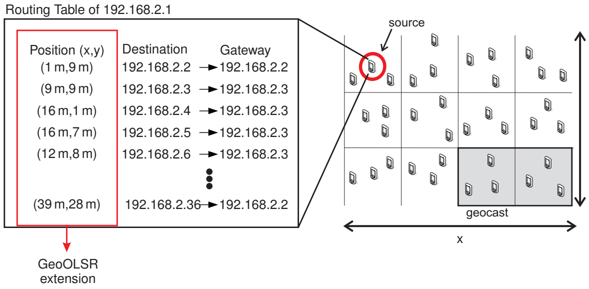


Figure 9. Lifetime of a 250mAh battery for different operation modes with the applied traffic pattern in Table III in a point-to-point scenario without mobility

Table V
BATTERY LIFETIME DEPENDING ON APPLIED ROUTING SCHEME FOR PARAMETERIZATION IN TABLE III FOR IEEE802.15.4 AND IEEE802.11

IEEE802.15.4 CSMA/CA	173h 36min
IEEE802.15.4 AODV	59h 20min
IEEE802.15.4 OLSR	54h 57min
IEEE802.11b (1Mbit/s) OLSR	55min
IEEE802.11b (1Mbit/s) AODV	55min


Routing Table of 192.168.2.1

Position (x,y)	Destination	Gateway
(1 m,9 m)	192.168.2.2 →	192.168.2.2
(9 m,9 m)	192.168.2.3 →	192.168.2.3
(16 m,1 m)	192.168.2.4 →	192.168.2.3
(16 m,7 m)	192.168.2.5 →	192.168.2.3
(12 m,8 m)	192.168.2.6 →	192.168.2.3
	⋮	
(39 m,28 m)	192.168.2.36 →	192.168.2.2

Figure 10. Basic idea of extending OLSR to map location based services on IP-based unicast messaging

OLSR consumes slightly more energy than AODV in this configuration, which is caused by the relative high rate of control packets to maintain overall network topology information in each node. However, in comparison to Wi-Fi networks operating at 1MBit/s, the node lifetime is about 60 times higher for both cases.

V. GeoOLSR

In this chapter, we extend the original OLSR with geocasting capabilities. The main idea of GeoOLSR is shown in Figure 10. Each node within the whole network administrates a modified routing table, which contains the IP address and position of every neighbor node. This enables a direct mapping of position information to regular IP addresses, which facilitates efficient forwarding of location based information. However, the performance of maintaining moving nodes in the routing table strongly depends on the update process, which is regulated by the periodic emission of OLSR control packets. Hence, relevant parameters have to be optimized for a sophisticated use within wireless sensor networks. We assume that each node participating in the entire network is aware of its position, which may be expressed by absolute or relative coordinates to a given fixed-point. For performance analysis, we use a random mobility model.

A. Extension of OLSR with Geocasting Capabilities

Due to the periodical exchange of Hello and Topology-Control messages in OLSR networks (cf. Section II), the key assumption is to use these two packet formats to broadcast position information as well as regular IP-based topology information within the network. The new *GeoOLSR Hello* frame extends the standard OLSR Hello packet with an additional header as follows:

- *Type (1 Byte)* The type field indicates the applied position format e.g., GPS-RMC (GPS-Recommended Minimum Sentence C), GPS- or Cartesian coordinates (8 Byte floating point for each x- and y-coordinate).
- *Length (2 Bytes)* Due to the variable length of GPS payload, this field denotes the byte length of the additional (position) payload.
- *Reserved (1 Byte)* This field enables future extensions like geo-referenced rescue maps, situation photos etc.

In contrast to that, *Topology Control (TC)* messages are used to broadcast information beyond 1 hop distances. TC messages are only forwarded by Multi Point Relays (MPRs), which are used to decrease the number of transmissions required for OLSR related control mechanisms. To broadcast position information of each node participating in the considered network, the TC message format also has to be adapted to GeoOLSR. In contrast to the *GeoOLSR Hello* packet, a *GeoOLSR TC* message may include more than one node position. Hence, a separate position data header is denoted for each advertised neighbor's main address. This additional header also includes a *Type*, *Length* and *Reserved* field. However, the two packet formats *GeoOLSR Hello* and *GeoOLSR TC* are only used during initialization. After network setup phase, recently joined or moving nodes can be added or updated to the topology by using the proposed GeoOLSR frames *Fast Hello* and *Fast TC*, which will be explained later in this section. The application of the modified *GeoOLSR Hello and TC* packets only at network startup enables a fast convergence to the original OLSR algorithm without changes on specific OLSR route and MPR selection. As a consequence, there will not be any position related update after initialization when there is no node mobility within the network. However, each node needs to maintain a node list, in which the coordinates are saved together with the corresponding IP addresses, which enables a mapping of position information to regular IP addresses.

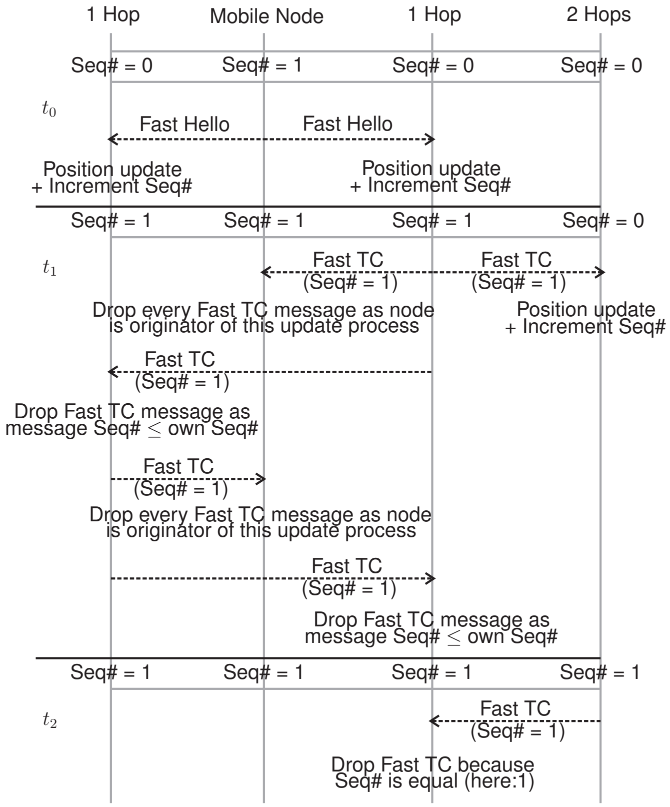


Figure 11. Update Process exchanging Fast Hellos and Fast TCs

In order to enable accurate position updates at high node mobilities even in far-off nodes, we modified the Fast OLSR approach of Benzaid et al. [12] (cf. Figure 11).

In this paper, we also use fast hello messages to track the fast moving nodes' motion sufficiently. To achieve this goal, a moving node (or a node, which recently joined the regarded network) emits position update packets to its direct neighbors at a high frequency in form of *GeoOLSR Fast Hello* messages. In contrast to the original Fast OLSR approach, we do not apply *GeoOLSR Fast Hellos* to increase overall network redundancy, but rather accuracy of position information. That means we reduce fast hello message fields to a minimum, including only position data. Here, no additional IP address of the sending node is required as this information is already denoted in the regular OLSR header. The frequency of *GeoOLSR Fast Hello* emission is determined by the new parameter *Fast Hello Interval*. Thus, our *GeoOLSR Fast Hello* packet format is developed for resource constrained IEEE802.15.4 nodes and allocate a minimum of payload.

In contrast to *GeoOLSR Fast Hello* messages, *GeoOLSR Fast TC* messages are used to transfer position updates to far-off nodes within the network. In contrast to regular Topology-Control messages of OLSR, *GeoOLSR Fast TC* messages are distributed using broadcast. To limit the broadcasts after each node has received the updated position, a 2 Byte sequence number is integrated in the *GeoOLSR Fast TC* message format besides the IP address and the new position of the moving node. This allows a fast distribution of position updates without profound changes of the OLSR protocol. An example of the update process is shown in Figure 11. At the beginning the mobile node recognizes that it is moving and thus sends *GeoOLSR Fast Hellos* to all neighboring nodes. All nodes, which receive a *GeoOLSR Fast Hello* update the corresponding position information of their geocast table. The explicit assignment is achieved by fixed IP addresses. Each geocast table entry is equipped with an additional sequence number that is incremented when a position update is performed. After that, another *GeoOLSR Fast TC* message is generated and broadcasted, which includes the recently updated position, the IP address of the moving node and the incremented sequence number. Every node, which receives a *GeoOLSR Fast TC* - except the originator of the update process - checks if the packet's sequence number is larger than its own. If true, the new position is updated and forwarded, otherwise the packet is discarded. In Table VI the most important GeoOLSR parameters and their behavior on network performance are shown.

Table VI
MAIN PARAMETERS OF GEOOLSR

Hello Interval	Emission interval of *GeoOLSR Hello* messages.
TC Interval	Emission interval of *GeoOLSR TC* messages.
Fast Hello Interval	Emission interval of *GeoOLSR Fast Hello* messages.
Network Init Time	Based on the network size this value limits the time until *GeoOLSR Hello and TC* messages are used for position updates. After Network Init Time only *GeoOLSR Fast Hellos* and *GeoOLSR Fast TCs* are used for position updates.

B. Broadcasting data using geocast regions

A general problem that occurs using location based services in a Wireless Sensor Network (WSN) is the limited payload of IEEE802.15.4 Medium Access Control (MAC) layer. Thus, the MAC layer, on which an IP and an UDP layer are based, offers only a maximum payload of 74 Byte.

Assuming many nodes to be situated in the considered geocast region, it is not advisable to route the file to each destination node separately. Therefore, the packets are first delivered to one or more gateways. After that, they will be broadcasted within the corresponding geocast region. This method is shown in Figure 12. Due to the proactive approach of OLSR, the source node has a full overview over all node positions in the entire network. Thus, it can calculate, which nodes are situated in the destination region. Then, the source node determines a node, which is placed most closely in the middle of the desired geocast region.

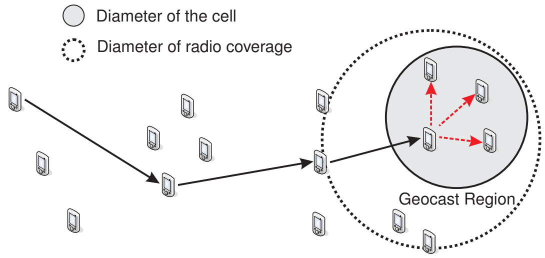


Figure 12. Broadcasting location based data via GeoOLSR

To ensure an adequate connectivity within the geocasting regions, the cell sizes must be smaller than the radio coverage of the nodes. If the resulting coverage area does not overlap fully with the desired destination region, two or more gateway nodes must be selected by the corresponding source node. Furthermore, the particular recipients are always aware of their own positions and may drop data packets, if the node is currently situated outside the desired geocast region.

The amount of gateways depends on the size of the considered geocast region and the cell size, which is strongly influenced by particular application environment properties (i.e., outdoor or indoor). Furthermore, environmental conditions influence the radio coverage and have to be estimated with certain channel models in advance. The selection of the cell size has a high impact on the connectivity between neighboring grids. Hence, a smaller cell size means more number of gateways in the network, resulting in a higher overhead of delivered packets and decreased battery lifetime especially in WSNs. However, this discussion has already been made in the GeoGrid thesis of Wen-Hwa Liao et al. (cf. [15]) and is not part of the present work.

VI. GeoOLSR Protocol Analysis

To evaluate the performance of GeoOLSR, we implemented a full mesh capable node based on the physical layer of the IEEE802.15.4-2006 standard in OMNeT++ 4.0 [19]. To achieve this goal, a new developed non beacon enabled MAC layer was used with an IP and UDP layer based on it. This step was necessary because regular IEEE802.15.4 nodes usually imply a network coordinator to synchronize the nodes of the entire network. On the other hand, network coordinators depict a Single-Point-of-Failure (SPOF), which is not desired in safety critical applications. Furthermore, the non beacon enabled MAC layer enables the WSN nodes to perform peer-to-peer communication, which is essential for mesh networks. In addition to that, the IPv4 compliant approach enables various standard applications that are widely-used on the Internet like VoIP or Email. Those VoIP capabilities are appropriate to push voice alarming or warning messages into endangered zones addressed by geocast regions. For a more intelligible comparison of GeoOLSR with various Geocast algorithms, we first analyze only the performance impact of the applied protocols and neglect PHY layer issues. These effects will be demonstrated in detail in Section VII.

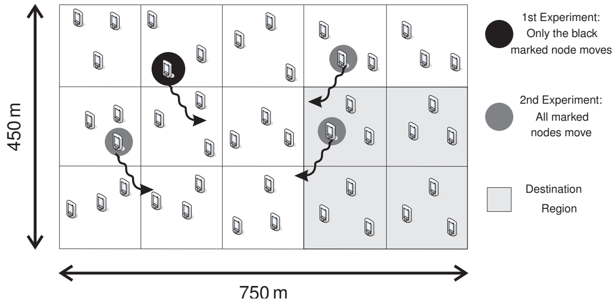

450 m
750 m
1st Experiment: Only the black marked node moves
2nd Experiment: All marked nodes move
Destination Region

Figure 13. Performance evaluation scenario

A. Validation Scenario

To analyze the performance of GeoOLSR, the following scenario (Figure 13) will be used for all test setups. The scenario measures 750 m x 450 m and consists of 45 nodes distributed homogeneously. To compare GeoOLSR with other geocast algorithms, the source node forwards data into the marked destination zone. In this application scenario the destination region consists of four neighbored zones.

B. Comparison between GeoOLSR and widely used Geocast Algorithms

This section analyzes and evaluates the performance of GeoOLSR with various Geocast algorithms in the scenario mentioned above (cf. Figure 13). In this scenario we omit node mobility and analyze the resulting overhead for a data transmission of 10 kByte from source to the marked destination region. This data transmission is repeated 100 times before a mean value e.g., End-to-End Delay or Transmission Time is calculated. In this experiment the destination region measures 300 m x 300 m and the cell ranges are set to 150 m, which is the maximal free space range of IEEE802.15.4 applying 1 mW transmission power. The results are shown in Table VII.

Table VII
Comparison of the different Geocasting Algorithms

	LBM	flooding GRID	ticket GRID	GeoTORA	GeoOLSR
Effective Data Rate [$kBit/s$]	4.5	12.4	9.8	35	36.2
End-to-End Delay [ms]	45.6	15.2	39.4	12.8	12.8
Transmission Time [s] (UDP Payload = 10 kB)	19.2	7.1	10.7	2.5	2.4
Number of Packets for Transmission	4943	2269	1800	668	648
Overhead [Byte]	14	12	21	4	2
Payload to overall frame size [%]	75.7	78.4	66.2	89.1	91.9
Inherent IP support	no	no	no	no	yes

We observe, that only the route maintaining algorithms support high effective data rates and low end-to-end delays (cf. Section II-B). However, the resulting differences between GeoTORA and GeoOLSR regarding effective data rate and transmission time are caused by our implementation of GeoTORA on ISO-OSI layer 7 whereas GeoOLSR is implemented on ISO-OSI layer 3. Thus, GeoOLSR is able to support slightly higher effective data rates and a little lower transmission time for each delivered packet than GeoTORA. Another important fact that can be omitted is the real time capability of LBM, flooding and ticket based GRID, GeoTORA and GeoOLSR. If we interpret the 10 kByte of Payload as a 5 s speech packet (16 kBit/s sampling rate and G.726 voice codec), we see that GeoTORA and GeoOLSR need 2.5 s and 2.4 s respectively to forward this voice alarm message into a certain destination area. In comparison to that LBM and the two GRID derivates show significant higher transmission times than the original speech length contained in the 10 kByte data packet. Hence, we can conclude that only the two route maintaining algorithms are able to support real time simplex voice transmissions. Furthermore, GeoTORA and GeoOLSR are able to save battery lifetime significantly as the overall number of packets needed for the transmission is smaller than the values for LBM, flooding GRID and ticket GRID. Finally, GeoGrid uses 4 Byte for Next Hop, Message Type and Packet Number signaling whereas GeoOLSR uses only 2 Bytes for packet sequence numbering.

Thus, we can postulate that GeoTORA and GeoOLSR are both suited for an application in Wireless Sensor Networks. However, we neglected node mobility until now, which is a very important issue for the aimed application in safety critical scenarios where nodes can exhibit relatively high mobilities. Nevertheless, GeoOLSR depicts two main advantages in comparison to GeoTORA. First, GeoOLSR as a proactive MANET algorithm is able to send an alarm message out immediately form a certain control center, whenever a threatening situation occurs without initiating a previous polling mechanism. Another advantage of GeoOLSR is the simultaneous support of IP-based traffic and location based traffic. That means, no additional geocasting algorithm is needed and the network management is completely integrated in ISO-OSI layer 3.

C. Node Mobility

The previous section neglected nodes' mobility. However, the knowledge of the correct position of each node has a high influence on geocast algorithms. As a quality indicator we regard the position deviation of all fixed nodes between the routing table entry and the real position. That means the difference of the predicted mobile node's position and the real location is calculated for 100 seconds in each node. After that, an overall mean value of the position deviations of all nodes is computed. To allow easier comparability, we

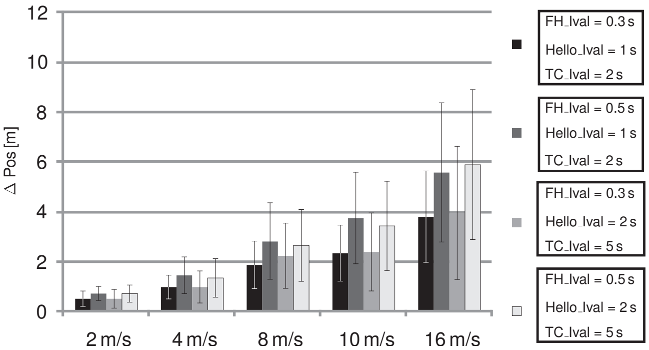


Figure 14. Position Deviation of GeoOLSR depending on varying moving speeds and parameter sets regarding 1 moving node

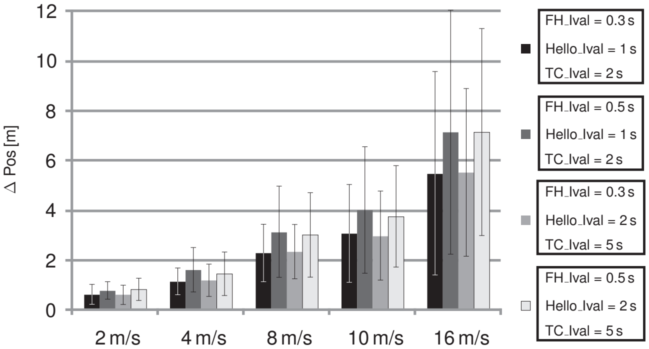


Figure 15. Position Deviation of GeoOLSR depending on varying moving speeds and parameter sets regarding 4 moving nodes

show the same scales for varying motion patterns. In the first experiment, we consider only one node moving around at different speeds. In the second setup, four nodes move through the scenario (cf. Figure 13). The results of the first experiment are shown in Figure 14. The overall mean value of position differences increases as expected with higher node mobility. Here, the same phenomenon can be observed with standard deviations, which indicate a successive rise with higher speeds. However, we can conclude that the ratio of the position deviation to the observed movement speed is always constant. That means, there is a constant average time, in which no communication between nodes is possible due to route maintenance, disconnections etc. Furthermore, we do not see an obvious impact of Hello and TC intervals on position accuracy in contrast to the key parameter Fast Hello interval. As a consequence, position deviation and standard deviation values using equal parameters for Hello and TC interval show nearly the same Δ positions. In the next step we evaluate the influence of higher node mobility within our scenario. Therefore, we compare position deviations of four moving nodes (Figure 15) with those of only one moving node (Figure 14). It is obvious that the increased number of moving nodes does not have a significant influence on the position accuracy. The difference between position deviations caused by one moving node and four mobile nodes does not exceed 30 % when the most network load generating parameters (Fast Hello = 0.3 s, Hello = 1 s and TC = 2 s) are applied at a node speed of 16 m/s. Furthermore, the average increase of position deviations between the one moving node scenario and the four moving nodes scenario is 15.55 %. This leads to an important question whether higher route maintaining updates imply higher position accuracies. It is visible that the application of the parameters Fast Hello = 0.3 s, Hello = 2 s and TC = 5 s leads to similar position accuracies like using the parameter set Fast Hello = 0.3 s, Hello = 1 s and TC = 2 s. Due to battery and resource constrains it is advisable to use the parameter set with lower Hello and TC intervals as this approach saves battery life and decreases the number of collisions.

D. Analysis of resulting overheads

In this section, we will analyze the overhead evoked by GeoOLSR in comparison to regular OLSR. The test scenario is the same as shown in Figure 13. As a reference, regular OLSR is considered without geocasting functionalities. In this experiment we consider one moving node and four moving nodes for two different OLSR parameter sets. Here, we neglect varying speeds as GeoOLSR only uses time triggered route maintenance packets, which are independent of different speeds. The results are shown in Table VIII.

Table VIII
RESULTING OVERHEADS COMPARED TO REGULAR OLSR

OLSR	regular OLSR with 45 static nodes 44.42 $kBit/s$		
GeoOLSR	1 moving Node	4 Moving nodes	Hello Interval = 1 s TC Interval = 2 s
Fast Hello = 0.3 s	60.88 $kBit/s$	79.5 $kBit/s$	
Fast Hello = 0.5 s	51.57 $kBit/s$	71.94 $kBit/s$	

OLSR	regular OLSR with 45 static nodes 25.61 $kBit/s$		
GeoOLSR	1 moving Node	4 Moving nodes	Hello Interval = 2 s TC Interval = 5 s
Fast Hello = 0.3 s	51.42 $kBit/s$	72.69 $kBit/s$	
Fast Hello = 0.5 s	42.29 $kBit/s$	65.74 $kBit/s$	

In contrast to Section III, we did not evaluate goodputs here, because the use of the random mobility model leads to fluctuating goodput values and are not comprehensible due to variable hop distances between source and sink.

We see that even the most accurate parameter set shows an overhead of 35.08 kBit/s (OLSR compared to GeoOLSR with *Fast Hello Interval* = 0.3 s and 4 moving nodes). Furthermore, the overhead of one moving node compared to 4 moving nodes is in-between 18.62 and 23.45 kBit/s (*Fast Hello Interval* = 0.3 s versus *Fast Hello Interval* = 0.5 s). Thus, even in the most data rate consuming parameterization there are still 170.5 kBit/s left for payload traffic.

VII. Performance Evaluation in a Real World Environment

Industrial scenarios pose a challenging network environment for IEEE802.15.4 networks due to the special fading conditions. In order to analyze the performance of protocols and applications, network simulators like OMNeT++ only apply a deterministic free space loss propagation model. This model, however, poorly reflects the channel characteristics of real world conditions. Therefore, a sophisticated ray tracing tool (Radiowave Propagation Simulator) is used to represent shadowing effects and multipath propagation. To increase accuracy of the simulation results, we used a 3D laser scan for creating a CAD model of the scenario, which considers every pipe, tube and steel girder included in the observed basements underneath a batch annealing plant of a ThyssenKrupp cold rolling mill (cf. Figure 16).

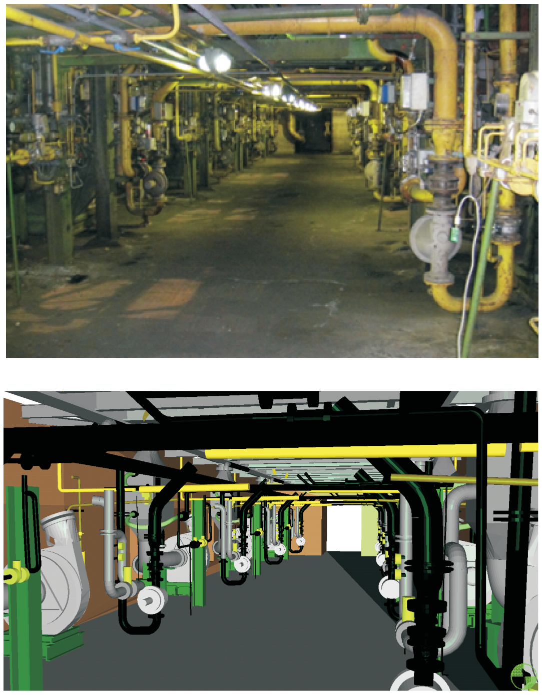


Figure 16. Top: Image of the supply machinery basement underneath the analyzed cold rolling mill. Bottom: Detailed 3D CAD model of the scenario shown above.

The combination of a highly detailed CAD model with ray tracing allows an accurate determination of Received Signal Strength Indicator (RSSI) values as well as Signal to Noise plus Interference Ratios (SNIR). IEEE802.15.4a-CSS, as applied in the gas concentration monitoring scenario for employee localization in case of emergency, possesses a minimal RSSI value of -95 dBm and a minimal SNIR of -17 dB. Hence, we modified the applied OMNeT++ PHY

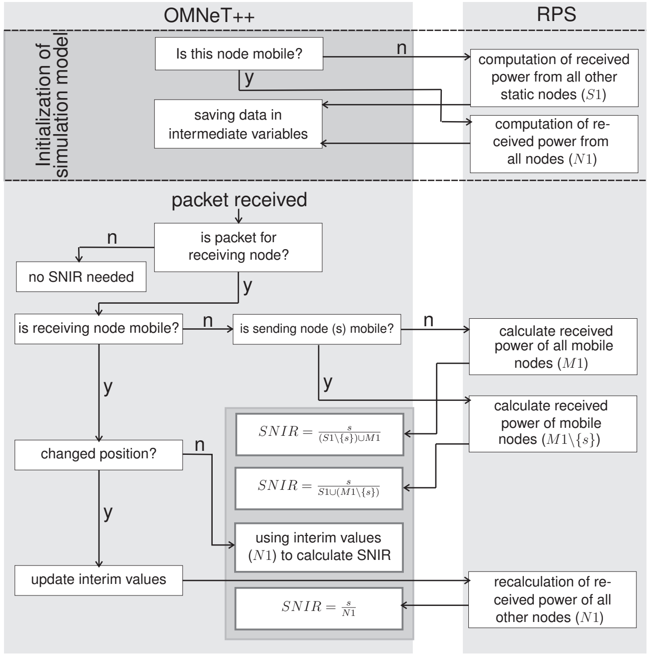

OMNeT++
RPS
Initialization of simulation model
Is this node mobile?
n
y
computation of received power from all other static nodes ($S1$)
saving data in intermediate variables
computation of received power from all nodes ($N1$)
packet received
is packet for receiving node?
n
no SNIR needed
y
is receiving node mobile?
n
is sending node (s) mobile?
n
calculate received power of all mobile nodes ($M1$)
y
y
calculate received power of mobile nodes ($M1\backslash\{s\}$)
$SNIR = \frac{s}{(S1\backslash\{s\})\cup M1}$
changed position?
n
$SNIR = \frac{s}{S1\cup(M1\backslash\{s\})}$
y
using interim values ($N1$) to calculate SNIR
update interim values
recalculation of received power of all other nodes ($N1$)
$SNIR = \frac{s}{N1}$

Figure 17. Simulation Architecture consisting of OMNeT++ and Radiowave Propagation Simulator (RPS) to increase PHY layer modeling accuracy. For computational time reduction, two intermediate result sets ($S1$ and $N1$) are applied during initialization period of the OMNeT++ model.

layer implementation to discard incoming packets that do not exhibit these minimum values and extended it with a direct connection to the ray tracing tool. However, to reduce computational complexity, we perform a special SNIR computation, in which two different intermediate results are saved, which may be reused on every SNIR calculation. The simulation architecture is shown in Figure 17. During initialization of OMNeT++, the sum of all adjacent stationary nodes is calculated for each non mobile node ($S1$) (anchor nodes that are mounted to the wall), whereas the received power of all adjacent nodes is cumulated for every mobile node in the scenario ($N1$). The intermediate value of each stationary node remains constant during simulation process and must be modified by the sum of all mobile nodes ($M1$) that do not participate in the observed communication process. $M1$ must be recomputed every time a SNIR value is requested by the PHY layer due to the mobility of this node set and the consecutive changes in RSSI and SNIR values. To reduce the amount of computational steps once again, the intermediate result set $N1$ is updated every 1 m only. The applied movement paths are shown in Figure 18.

In this setup, we apply a radio channel, which occupies 80 MHz of bandwidth with a center frequency of 2.45 GHz as our installed localization tags and anchor nodes use IEEE802.15.4a-CSS. Furthermore, we use dipole antennas with 2.2 dB gain and a transmission power of 0 dBm. The sent traffic profile applies packets encapsulating a payload of 74 Byte with an interarrival time of 30 ms (as applied in the evaluations before). First, we analyze the resulting goodput for different speeds in this scenario (cf. Figure 19) including a basement change (moving direction No. 1) and an exemplary maintenance of an anchor point (moving direction No. 2). During maintenance of machinery or stationary anchor points the service employees might be shadowed by surrounding tubes or pipes. Here, reliable handover processes must ensure connectivity of the mobile personnel.

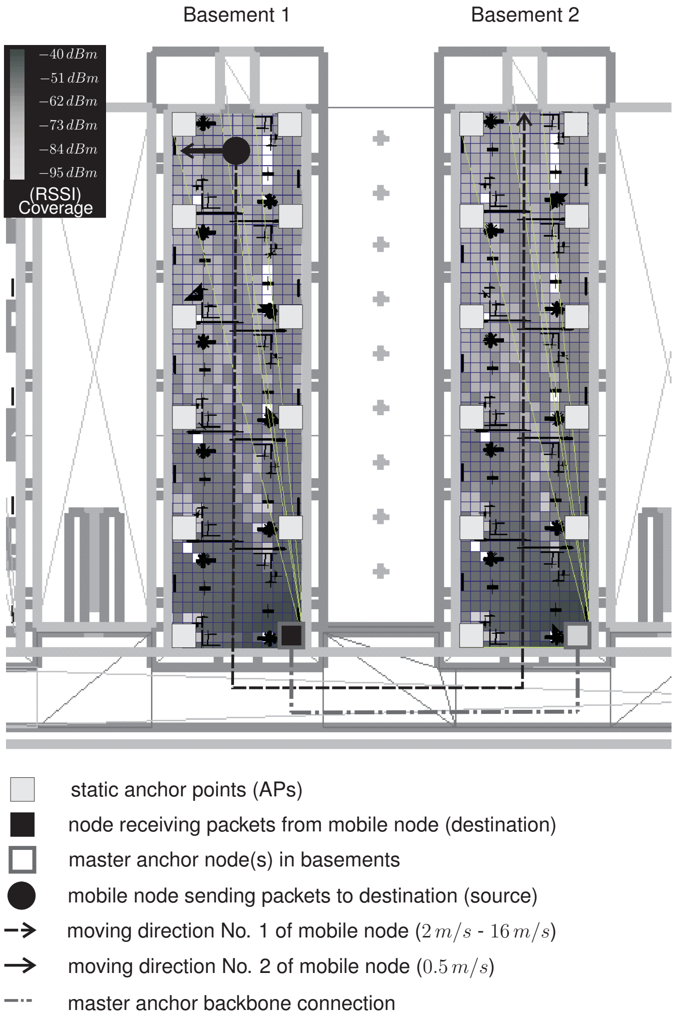

Basement 1
Basement 2
$-40\,dBm$
$-51\,dBm$
$-62\,dBm$
$-73\,dBm$
$-84\,dBm$
$-95\,dBm$
(RSSI) Coverage
static anchor points (APs)
node receiving packets from mobile node (destination)
master anchor node(s) in basements
mobile node sending packets to destination (source)
moving direction No. 1 of mobile node ($2\,m/s$ - $16\,m/s$)
moving direction No. 2 of mobile node ($0.5\,m/s$)
master anchor backbone connection

Figure 18. Applied moving direction of a mobile node which sends data to a static anchor point (AP). The other APs depict potential interferers in the observed positioning system.

As the scenario omits a very good radio coverage (as shown in Figure 18), there are only connections with a maximal 2 hop distance between source and destination. However, the resulting SNIR affect the radio channel significantly. Hence, the main influencing factor for the resulting goodputs of the first four measurements (0.5 m/s, 2 m/s, 4 m/s and 8 m/s) is multipath propagation (for both desired

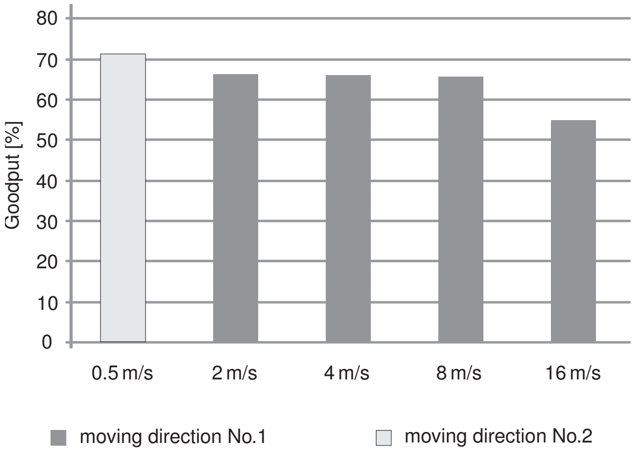


Figure 19. Resulting Goodput of GeoOLSR depending on varying moving speeds and parameter sets regarding 1 moving node under real-world channel conditions. ($FH\_Ival = 0.5\,s$,$HI = 2\,s$, $TC\_Ival = 5\,s$)

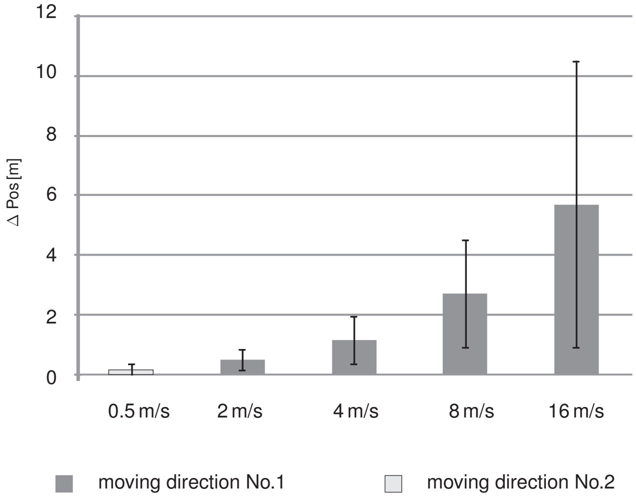


Figure 20. Position Deviation of GeoOLSR depending on varying moving speeds and parameter sets regarding 1 moving node under real-world channel conditions. ($FH\_Ival = 0.5\,s$,$HI = 2\,s$, $TC\_Ival = 5\,s$)

connections and undesired interference signals). Here, we only show exemplary results in Figure 19 without declaration of mean values or standard deviations. Compared to the original OMNeT++ analysis of OLSR (cf. Figure 6), a 20 % worse result is achieved considering the 1 and 2 hop cases. The 16 m/s measurement is subject to the increased speed as well as in the original OMNeT++ simulation. Nevertheless, the goodputs are still satisfying in such a scenario if low pedestrian speeds are assumed. Usual movement speeds for employees would be around 1 m/s up to 2 m/s.

Another important metric for safety critical localization systems is the position deviation as analyzed in Section V. The position deviation of GeoOLSR in a real-world scenario (Figure 20) is nearly equal to the previous scenario setup (Figure 14). This may be explained by the slightly reduced maximal hop count in the real-world scenario compared to the original measurement in Section VI. Furthermore, the deviation is still relatively small for low mobility speeds, which are typical due to the construction type of industrial environments where fast movements of employees do not occur frequently. Thus, fast evacuation is ensured as the resulting position deviations correspond an "arm's length" (for mobility speeds of up to 2 m/s), which enables firemen to rescue people quickly and reliable even if sight is limited.

VIII. Conclusion

We presented a novel peer-to-peer enabled IEEE802.15.4 node design for meshed network topologies and compared it against the original IEEE802.15.4 solution. We have seen that the energy consumption of our GeoOLSR nodes is about 3 times higher (3297 minutes) than the energy optimized end devices of the IEEE802.15.4 standard (10416 minutes), but the lifetime is enhanced in comparison to IEEE802.11 (55 minutes). The major advantage of the node is the enhanced fault tolerance against node failures and the autonomous reconfiguration capability. The routing algorithms provide good performance in handover processes in terms of switching times and goodput.

Subsequently, we also outlined a geocasting algorithm based on Optimized Link State Routing (OLSR) that is able to support high mobilities at a reasonable traffic overhead. Due to the proactive nature of the underlying OLSR protocol this extension is well suited for real time alarming services in safety critical scenarios, which do not permit an additional polling mechanism e.g., if danger zones must be evacuated immediately.

The deployment of IP in wireless sensor networks enables an easy integration of sensor nodes into preexisting infrastructures, without the need of special gateways, as well as a wide variety of services, which are widely accepted within the Internet community. Finally, GeoOLSR is able to use IP-based unicast traffic as well as location based services without the need of an additional geocasting algorithm beside the applied mesh network algorithm.

Acknowledgment

This work is conducted within the SAVE Project (Geographic Information System for Alarming using autonomous, networked Gas Sensors) and is funded by the German Federal Ministry of Education and Research (BMBF) – 16SV3711

References

[1] A. Lewandowski, V. Koester, and C. Wietfeld: *Performance Evaluation of AODV- and OLSR-meshed IP-enabled IEEE802.15.4*, 2010 Third International Conference on Advances in Mesh Networks (MESH 2010), Venice, Italy, July 2010

[2] C. Perkins, E. Royer, and S. Das: *Ad hoc On-Demand Distance Vector (AODV) Routing*, IETF RFC 3561, 2003

[3] T. Clausen and P. Jacquet: *The Optimized Link State Routing Protocol (OLSR)*, IETF RFC 3626, October 2003

[4] Z. Shelby and C. Bormann: *6LoWPAN: The wireless Embedded Internet*, John Wiley & Sons Ltd, ISBN: 978-0-470-74799-5, 2009

[5] ZigBee Alliance. ZigBee Specification. Technical Report Document 053474r17, Version 1.0, ZigBee Alliance, October 2007

[6] J. W. Hui and D. E. Culler: *IP is dead, long live IP for wireless sensor networks*, In Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems (SenSys '08), Raleigh, NC, USA, November 2008

[7] A. Dunkels, T. Voigt, and J. Alonso: *Making TCP/IP Viable for Wireless Sensor Networks*, In Proceedings of the First European Workshop on Wireless Sensor Networks (EWSN 2004), Berlin, Germany, January 2004

[8] K. Kim, S. D. Park, G. Montenegro, and S. Yoo: *6LoWPAN Ad Hoc On-Demand Distance Vector Routing (LOAD)*, draft-daniel-6lowpan-load-adhocrouting-01, IETF Internet Draft (Work in progress), July 2005

[9] J. Deissner, J. Huebner, D. Hunold, and J. Voigt: *RPS Radiowave Propagation Simulator*, User Manual, www.actix.com, last visited June 2011

[10] D. B. Johnson, D. A. Maltz, and Y.-C. Hu: *The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4*, IETF RFC 4728, February 2007

[11] S. Hamma, E. Cizeron, H. Issaka, and J.-P. Guedon: *Performance evaluation of reactive and proactive routing protocol in IEEE 802.11 ad hoc network*, In Proceedings of ITCom 06, Boston, MA, USA, October 2006

[12] M. Benzaid, P. Minet, and K. Alagha : *Integrating fast mobility in the OLSR routing protocol*, 4th IEEE Conference on Mobile and Wireless Communications Networks , Stockholm, Sweden, September 2002

[13] C. Gomez, P. Salvatella, O. Alonso, and J. Paradells: *Adapting AODV for IEEE802.15.4 mesh sensor networks: theoretical discussion and performance evaluation in a real environment*, 2006 International Symposium on World of Wireless, Mobile and Multimedia Networks (WoWMoM 2006), pp. 159 - 170, Buffalo, NY, USA, June 2006

[14] Y.-B. Ko and N.H. Vaidya: *GeoTORA: a protocol for geocasting in mobile ad hoc networks*, In Proceedings of the International Conference on Network Protocols, pp. 240-250, Osaka, Japan, November 2000

[15] W.-H. Liao, Y.-C. Tseng, K.-L. Lo, and J.-P. Sheu : *GeoGrid: A Geocasting protocol for mobile ad hoc networks based on grid*, Journal of Internet Technology, pp. 23-32, 2000

[16] Y.-B. Ko and N.H. Vaidya: *Geocasting in Mobile Ad Hoc Networks: Location-Based Multicast Algorithms*, In Proceedings of the Second IEEE Workshop on Mobile Computer Systems and Applications (WMCSA 1999), Washington, DC, USA, 1999

[17] D. Camara and A. Loureiro: *A Novel Routing Algorithm for Ad Hoc Networks*, Baltzer Journal of Telecommunications Systems, Kluwer Academic Publishers, vol. 18:1-3, pp. 85-100, 2001

[18] B. Zhou, F. De Rango, M.Gerla, and S. Marano: *GeoLANMAR: geo assisted landmark routing for scalable, group motion wireless ad hoc networks*, IEEE 61st Semiannual Vehicular Technology Conference (VTC), Stockholm, Sweden, 2005

[19] A. Varga: The OMNeT++ Discrete Event Simulation System, In Proceedings of the European Simulation Multiconference, 2001

[20] Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs), IEEE802.15.4 (Standard), 2006

[21] A. Kiryushin, A. Sadkov, and A. Mainwaring: *Real-World Performance of Clear Channel Assessment in 802.15.4 Wireless Sensor Networks*, 2008 Second International Conference on Sensor Technologies and Applications (SENSORCOMM 2008), pp. 625-630, Cap Esterel, France, August 2008

[22] A. Lewandowski, M. Putzke, V. Koester, and C. Wietfeld: *Coexistence of 802.11b and 802.15.4a-CSS: Measurements, Analytical Model and Simulation*, 71st IEEE Vehicular Technology Conference (VTC), Taipei, Taiwan, May 2010

[23] Y. Huang, S. N. Bhatti, and D. Parker; *Tuning OLSR*, 2006 IEEE 17th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC 2006), pp. 1-5, September 2006

[24] I. D. Chakeres and L. Klein-Berndt: *AODVjr, AODV simplified*, ACM, SIGMOBILE Mob. Comput. Commun. Rev., vol. 6, no.3, pp. 100-101, 2002 doi: http://doi.acm.org/10.1145/581291.581309, last visited June 2011

[25] Texas Instruments CC2420 Single-Chip 2.4 GHz IEEE 802.15.4 RF Transceiver, Datasheet, http://focus.ti.com/docs/prod/folders/print/cc2420.html, last visited June 2011

[26] Maxim MAX2822 2.4 GHz IEEE 802.11b RF Transceiver, Datasheet, http://www.maxim-ic.com/datasheet/index.mvp/id/3938, last visited June 2011

A Traffic Monitoring and Queue Detection System Based on an Acoustic Sensor Network

Barbara Barbagli, Luca Bencini, Iacopo Magrini and Gianfranco Manes
Dept. of Electronics and Telecommunications
University of Florence
Via di Santa Marta 3, 50139 Florence, Italy,
Email: barbara.barbagli@unifi.it, luca.bencini@unifi.it,
iacopo.magrini@unifi.it, gianfranco.manes@unifi.it

Antonio Manes
Netsens S.r.l.
Via Tevere 70, 50019 Florence, Italy
Email: antonio.manes@netsens.it

***Abstract*—Wireless Sensor Networks for real time traffic monitoring in Intelligent Transportation Systems is currently considered one of the challenging application area for this emerging technology. The promise of an unmanaged infrastructure, with a continuously decreasing cost per unit, attracts the attention of both final users and system integrator, opening new business opportunities. This paper describes a Traffic Monitoring Wireless Sensor Network system, based on acoustic arrays and powered by an effective post-processing detection. A practical case study is presented starting from a real problem and reaching the best architectural solution with particular focus on hardware implementation and communication protocol design. Finally, real experience results are shown to highlight the reliability of the developed system.**

***Keywords*-wireless sensor network; acoustic sensors; traffic monitoring; cross-layer routing protocol.**

I. Introduction

Real-time traffic monitoring and early queue detection is of paramount importance in Intelligent Transportation Systems (ITS). Distributed traffic monitoring on a large scale based on non intrusive/obtrusive solutions are highly desirable for traffic monitoring.

Conventional traffic surveillance systems make use of intrusive sensors such as inductive loop detectors or pressure sensors, for high accuracy in vehicle detection. However, these sensors disrupt traffic during installation and repair, and therefore have high installation and maintenance costs. These limitations have pushed towards the development of non-intrusive traffic monitoring technologies, including laser radars, passive infrareds, ultrasonics, passive acoustic arrays and video cameras. These systems have high equipment costs and their accuracy depends on environmental conditions. Moreover video cameras have involved huge data volume demands for a dedicated wired connection in order to communicate with the central server. As a result these solutions are not suitable for large-scale deployment and hence are restricted to small scale applications, where isolated monitoring points are located many kilometers apart.

Passive acoustic transducer-based surveillance systems have also been developed, featuring vehicle classification and multi-lane resolution capability; this is based on processing the characteristic sounds emitted by vehicles [3] [4]. Acoustic sensors are attractive especially for their low cost and simple and non-intrusive installation, however they require a sophisticated post-processing algorithm for extracting useful information.

Another relevant requirement pushing towards the design of an effective traffic monitoring system is to provide high spatial density measurements. A viable solution for achieving that purpose is a system based on a Wireless Sensor Network (WSN) infrastructure offering advantages in terms of flexibility and a significant reduction of installation costs; therefore making large scale deployment possible.

Several solutions based on wireless sensors have been investigated, including wireless magnetic sensors [5] [6], and coherent cross-correlated acoustic transducer [8]. Ding et al. [5] demonstrated wireless magnetic sensors embedded in the road sampling the magnetic field at its front and back ends, and internally processing this data in order to count vehicles for computing the average speed of passing vehicles.

The requirements that adopting a WSN are expected to satisfy in effective traffic monitoring monitoring concern both system level issues (i.e., unattended operation, maximum network life time, adaptability or even self-reconfigurability of functionalities and protocols) and final user needs (i.e., communication reliability and robustness, user friendly, versatile and powerful graphical user interfaces). The most relevant mainly concerns the supply of stand-alone operations. To this end, the system must be able to run unattended for a long period also in the absence of electricity. This calls for an optimal energy management ensuring that the energy spent is directly related to the amount of traffic handled and not to the overall working time.

An additional requirement is robust operative conditions, which needs fault management, since a node may fail for several reasons. Other important properties are scalability and adaptability of the network's topology, in terms of the number of nodes and their density in unexpected events with a higher degree of responsiveness and reconfigurability. Finally, several user-oriented attributes, including fairness, latency, throughput and enhanced data querying schemes [10] need to be taken into account even if they could be considered secondary with respect to our application purposes because the WSN's cost/performance trade-off.

In this paper, a WSN based on an array of acoustic sensors that detect and process the sound waves generated by the traffic flow using a low-cost microprocessor is proposed.

The paper is organized as follows: Section II provides an outlook on the system's composition and operation. Section III and Section IV describe the hardware and the basic operation of each constitutive element. Section V discusses the communication protocol. Finally, Section VI gives the experimental results of a continuous long-term operation.

II. TM-WSN Description

This paper describes a novel traffic monitoring (TM) system based on a Wireless Sensor Network (WSN) infrastructure; the TM-WSN system allows traffic monitoring and queue detection to be performed in real-time at an unprecedented space scale with an extremely low investment in installation and maintenance costs.

A significant characteristic of the system is the WSN infrastructure that combines, in its *basic module*, two kinds of nodes which are both based on *acoustic sensors*, yet they employ different operation techniques and hence, have different hardware characteristics.

The basic module of the system is composed of a Master Node (MN), which has superior computational and energy resources and is connected to a remote database via TCP/IP over UMTS. The MN is wirelessly connected to a number of regularly spaced Sensor Nodes (SNs) operating on a low duty-cycle and woken-up on demand. A basic module infrastructure deployed along the motorway is shown in Figure 1 . This module can be spatially replicated on both sides of the motorway to cover a wide area.

The sound signal is detected and processed by the embedded resource of the MN using an original algorithm that allow to automatically extract *traffic parameters* on site. The information is transmitted to a central server and made available to a remote user.

When a queue or traffic jam is detected at the MN location the SNs are activated by the MN in order to locate the position of the queue or traffic jam, thus providing a real-time picture of the traffic flow sampled at the same space interval as the SN deployed on the motorway. The communication between the devices is performed by a cross-layer MAC Routing protocol which will be described in Section V.

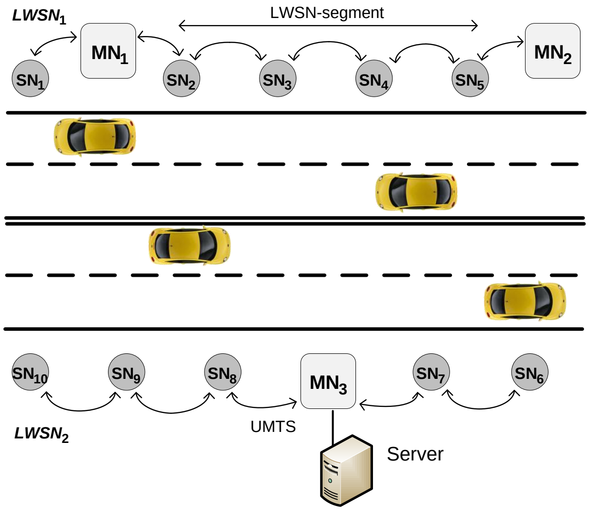


Figure 1. Basic system infrastucture.

III. MN Design and Operation

In this section is first illustrated the MN hardware solution adopted and then, the procedure allowing automatic extraction on-site of traffic parameters.

A. MN Hardware Design

The MN block diagram is shown in Figure 2. It is composed of a Sensor Unit which detects the audio signal coming from the road and a Computational Unit which performs the signal processing and vehicle detection while it simultaneously supports communication with both the associated SNs by the RF Unit and with the central server by the UMTS modem.

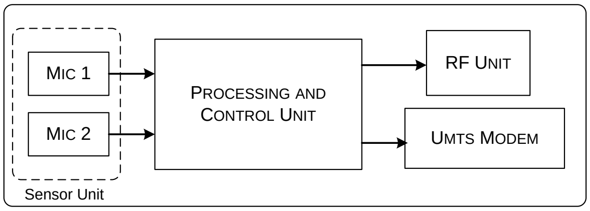


Figure 2. MN hardware block diagrams.

The computational unit consists of a commercial computer on module GUMSTIX VERDEX PRO XM-4 with a Marvell PXA270 400 MHz processor equipped with 64 MB RAM and 16 MB flash memory, and operating with a Linux OS. The audio signals detected by the acoustic sensors are sampled at 16 KHz and quantized at 16 bit, then processed with a real-time algorithm based on FFT routines for estimating the time delay via a coherent cross-correlation method.

A TCP-IP over UMTS Modem provides a bidirectional connectivity to the central server thus enabling a remote control of the MN operative parameters and creating an upgrade of the systems. The RF unit is based on Texas Instrument CC1000 low power transceiver operating in the UHF ISM band, implementing an FSK Manchester coding.

The setup of the MN is packaged into a compact lightweight panel which can be easy installed on the motorway's guardrail.

B. MN Operation and Parameters Extraction Procedure

All vehicles emit characteristic sounds when moving on the road. The sound signal is connected to the source's position therefore, in reference to [4] [8] traffic sensing and vehicle detection can be achieved by processing the signal detected by the acoustic sensor.

The *sensor unit* consists in a pair of microphones (MIC1 and MIC2) arranged in a characteristic setup and deployed along the roadside, with the baseline parallel to the moving direction of the source, as shown in Figure 3.

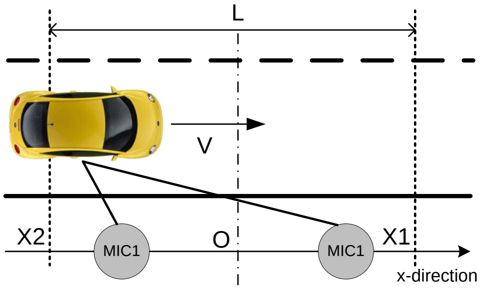


Figure 3. MN sensor unit setup.

The sound wave generated by a passing vehicle reaches the two microphones at slightly different times due to the difference in the air path; on the two signals a *cross-correlation method* is applied to estimate the time delay according to [4] [7].

The signal detected by the MIC1 and the MIC2 are respectively:

$$s_1(t) = s(t) \tag{1}$$

and

$$s_2(t) = s(t - \Delta t) \tag{2}$$

where $s(t)$ is the sound wave generated by the source (vehicle) and Δt represents the time delay between the two signals arriving at the two microphones.

The cross-correlation function of the two signals, $R_{12}(\tau)$ is formulated by:

$$R_{12}(\tau) = s_1 * s_2(\tau) = s * s(\tau - \Delta t) = R(\tau - \Delta t) \tag{3}$$

where $*$ denotes the convolution and $R(\tau)$ the autocorrelation function of $s(t)$.

According to [9], the signal produced by vehicles is a broad band, random-noise signal providing a cross-correlation function with a distinct peak at $|t - \Delta t|$.

In the cross-correlation domain, the position of the peak represents the source's time delay and changes with its position. Mapping the position of the peak in a time interval results in a digital *Sound Map*, which represents the source motion along a predefined track. A typical sound map is shown in Figure 4(a); the x-axis represents the observation time and the y-axis represents the time delay τ.

If further traveling speed is assumed as constant, the detected sound trace could be described by an analytic solution, accordingly with [], and could derive from the sound path difference in the air and expressed by:

$$\tau(t) = \frac{1}{V_s}\left[\sqrt{\left(x + \frac{d}{2}\right)^2 + y_0^2 + z_0^2} - \sqrt{\left(x - \frac{d}{2}\right)^2 + y_0^2 + z_0^2}\right] \tag{4}$$

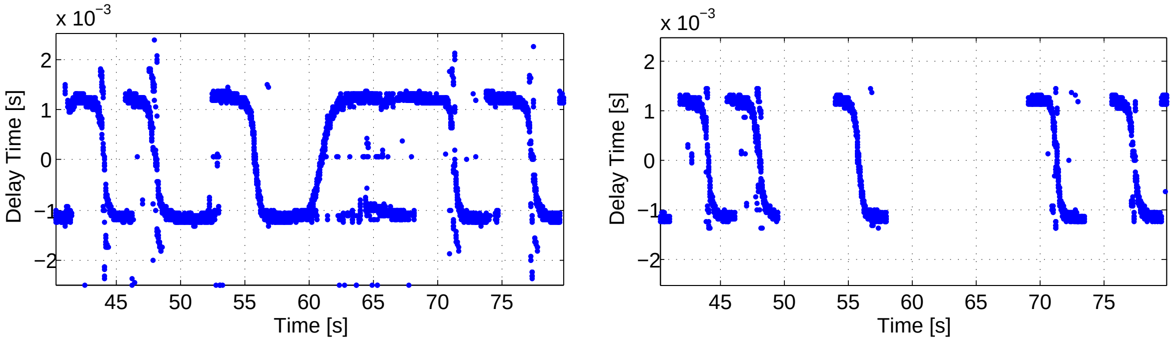
where, V_s is the speed of the sound in air, d is the microphones spacing, x is the vehicle position in the x-direction, y_0 is the distance between mics and vehicle, z_0 is the height of the mics above the ground.

Owing to the linearity of the cross-correlation, *multiple sound sources* can be processed, and thus they could appear in the Sound Map. Unwanted sound sources derive from background noises, typical of each complex outdoor environment, and vehicles moving in the opposite carriageway. As Figure 4(a) shows, the vehicles in the Sound Map appear as traces with an opposite slope whereas background noises appear as isolated points.

To enable an automatic reading of the map in an embedded environment, it is necessary to develop an effective post-processing algorithm. For that purpose very clear sound maps are required, where unwanted traces and surrender noises are both removed. Therefore a "cleaning" procedure has been implemented during the making of the sound map. As the energy associated with vehicles traveling in the opposite carriageway suffers from attenuation due to propagation effects, the related correlation peaks exhibit a much lower amplitude with respect to those coming from the nearest carriageway. Therefore a dynamic threshold, estimated on the average energy value, is applied at the front-end of the process. Furthermore, a reduction of the background noise has been attained, by limiting the time delay range of the correlation peak, consistently with the set up geometry.

As result of the procedure described, a Sound Map is given in Figure 4(b) and can be compared with the sound map shown in Figure 4(a). An automatic traffic parameters extraction can now be performed.

A Sound Map corresponding to a single vehicle transit is shown in Figure 5. We can observe that when the vehicle crosses the orthogonal axis of the setup, corresponding to the point "O" in Figure 3, the time difference is zero.



(a) Sound Map.

(b) Sound Map after the post-processing.

Figure 4. Sound Map before and after the post-processing.

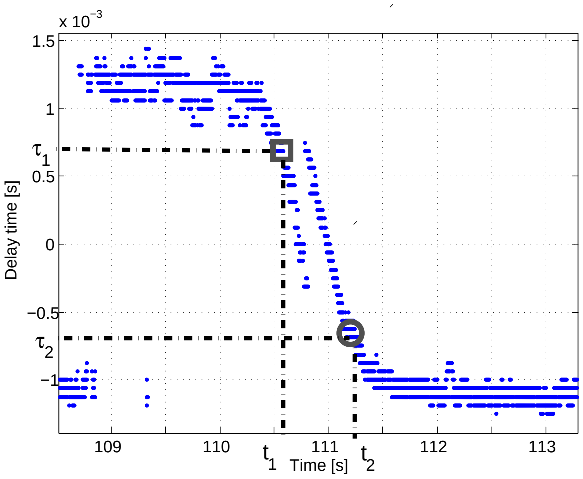


Figure 5. Vehicle detection.

As demonstrated in [8], the trace slope in this point is proportional to the vehicle speed.

As mentioned before, multiple sound sources could appear in the sound map. As in a vehicle the main acoustic source is represented by vehicle tyres; each sound map for a single vehicle, would consist of a two or more traces, each corresponding to a vehicle axle. This phenomenon can be observed in Figure 5.

To detect a *vehicle transit*, two symmetrical points corresponding to the positive time delay τ_1 and the negative time delay $\tau_2 = -\tau_1$ are positioned on the y-axis of the Sound Map (see Figure 5). Those time delays correspond to two symmetrical position, X1 and X2, of the vehicle along the traveling path, whose spacing is L (see Figure 3 for reference). A vehicle transit is detected if the sound trace intercepts the values τ_1 and τ_2 in a sequence that occurs when a vehicle pass through the two virtual positions X1 and X2. Therefore as τ_1 and τ_2 are selected in the linear portion of the trace, the vehicle traveling speed, V_v, can be easy calculated, according to the following expression:

$$V_v = \frac{L * k}{t_2 - t_1} \tag{5}$$

where k is a scale factor taking into account the vehicle axle tracks, D_{axle}, and $t_2 - t_1$ is the time interval taken by the vehicle to cover the distance L, as shown in Figure 5. The scale factor k is used to compensate the asymmetrical behavior of a generic vehicle trace, due to the above mentioned fact that there are two sound sources in the vehicle's front and back axle. Accordingly k is expressed by $L + D_{axle}/L$ and was estimated on a statistical basis. The resulting approximation in estimating vehicle traveling speed is consistent in our case, however, the system aims at evaluating the average traffic parameters, rather than the vehicle parameters. Square and circle markers in Figure 5, thus, represent the sequence associated with the transit of a vehicle, while the vehicle speed is evaluated according to (5).

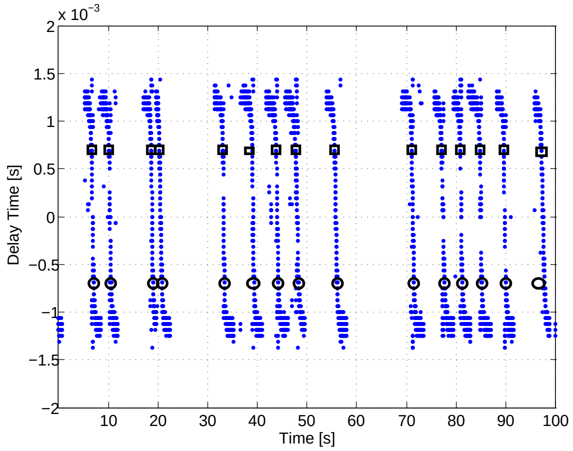


Figure 6. Multiple detection transit.

In Figure 6, a Sound Map is reported, showing the sequence of square and circle markers associated to multiple vehicle detection obtained from the previously described automatic procedure. As it can be observed all the passing vehicles are successfully detected in this case.

The output of the traffic parameters extraction routines is represented by some traffic parameters which indicate the traffic conditions at the MN location. These parameters are included in a summary report and sent to the central server.

The previously described method for traffic parameter extraction was extensively tested during a long period of continuous operation. In Section VI, we present the results of the long-term system operation.

IV. SN Design and Operation

In this section is illustrated the hardware design of the SN along with the SN operation.

A. SN Hardware Design

According to the proposed architecture, the sensor network also includes SN with limited computational capability, only relying on autonomous energy resources. The block diagram of the SNs is represented in Figure 7.

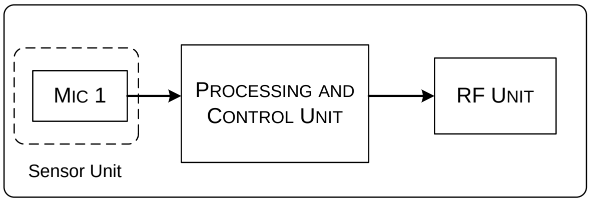


Figure 7. SN hardware block.

The main components of a SN are: the Sensor Unit, which consists of a single microphone, the Processing Unit ARM CORTEX M3 72 MHz, the RF Unit which has a Texas Instrument CC1000 low power transceiver operating in the UHF ISM band, implementing an FSK Manchester code.

The SN primary energy source consists of a 3 V Li-Ion rechargeable battery assisted by a 5 W solar panel as a secondary source. To preserve the battery life, the SNs are duty-cycled at an appropriate low rate.

B. SN Operation and Queue Detection Algorithm

As previously mentioned, the main job of the SNs is to produce traffic reports on-demand for dynamically locating the position of a queue or traffic jam. When a traffic queue or jam is detected at the MN location, the SNs associated with the MN are switched to *operative mode*, the detection of traffic conditions (fluid flow or queue) is performed through an analysis of the energy distribution features. As long as the SNs stay in the operative mode, they regularly produce a traffic report containing traffic conditions information that is passed to the MN according to a scheduling time interval. Communication between the devices is performed by a cross-layer MAC Routing protocol as described in Section V.

The MN reports the information to the central server about the traffic conditions at each individual SN; as a consequence, the *traffic flow distribution* is sampled at the same spacing interval as the SNs deployed on the motorway, thus a complete real-time picture of traffic flow is provided to the user/customer.

The SNs are able to estimate the acoustic energy generated by the traffic. The acoustic signal sensed by the microphone is first high-pass filtered at 1 KHz to cut-off background and wind noise components present in the environment. High-pass filtering is also useful to remove unwanted low-frequency contributions generated by vehicles traveling in the opposite carriageway, for instance, the sound generated by the air-stream of trucks; high frequency (above 1 KHz) energy components, mainly generated by the tyre noise of vehicles traveling in the opposite carriageway, are greatly attenuated by propagation effects. The energy detected by the SN is dominated by the sound sources in the nearer carriageway result in a well defined and space-limited acoustic footprint.

In Figure 8, an energy distribution associated with a traffic flow is presented. In the left side is a distinct peak corresponding to a *passing vehicle*; in the center of the trace a smoother energy distribution can be observed, representing the *standing vehicles*.

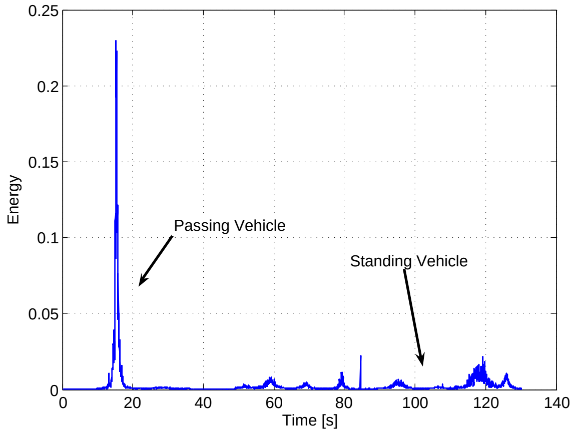


Figure 8. Energy distribution.

In fact, at vehicle speed in excess to some 30 Km/h, the dominant acoustic energy source is represented by *tyres*, featuring well defined energy peaks in the time domain; for standing vehicles, however, the dominant acoustic energy source is represented by *motor noise*, featuring a smoother energy distribution, with a much lower associated energy average. The conditions of regular traffic flow and queues/traffic jams in the nearer carriageway can be classified accordingly.

In fact, a fluid traffic condition is associated to the presence of isolated energy peaks, whereas a queue or traffic jam condition is associated to an energy floor, with a much lower associated energy average.

The processing unit computes the energy distribution in the time domain and an algorithm based on a state machine detects the passing vehicle. An adaptive threshold estimated on the energy value's moving average is on the basis of the state machine. A block diagram of the processing is shown in Figure 9.

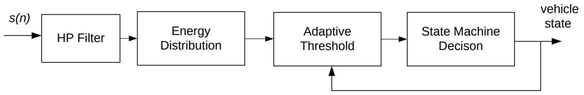


Figure 9. Block diagram of SN operation.

Figure 10 shows the result of this process compared with the sound map generated by the MN. It can be seen that in this case the implemented algorithm is capable of detecting all the peaks in the energy's distribution, as the vehicles are spaced a good distance apart. In heavier traffic conditions, however, the vehicle counting could be underestimated but, in any case, the energy distribution represents a useful indicator for estimating the traffic flow in the carriageway.

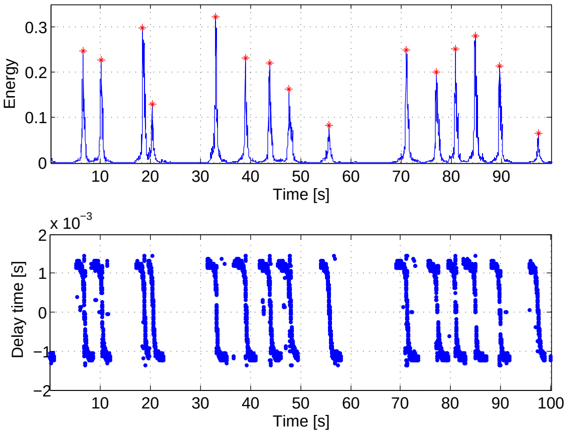


Figure 10. Correlation between energy distribution and sound map.

V. Protocol Design

The most relevant system requirements, which lead in the design of an efficient Medium Access Control (MAC) and Routing protocol for WSNs, mainly concern power consumption issues and the possibility of a quick set-up and end-to-end communication infrastructure. This calls for optimal energy management since a limited resources and node failure may compromise WSN connectivity. Therefore, the MAC and the network layer must be perfected ensuring that the energy used is directly related to the amount of handled traffic and not to the overall working time.

Other important properties are the *scalability* and *adaptability* of network topology, in terms of number of nodes and their density. As a matter of fact, some nodes may either be turned off or may join the network afterward.

Taking these requirements into account, a MAC protocol and a multi-hop routing protocol were implemented. A multi-hop approach was preferred as opposed to a star topology because it also helps to realize an end-to-end communication in the presence of obstacles (i.e., flyovers, trees, curves) that would otherwise prevent the establishment of a direct link between the SNs and the MNs.

Let us start our analysis by considering the wireless network architecture shown in Figure 1. It is comprised of two opposite Line Wireless Sensor Networks (LWSNs) deployed opposite each other (i.e., along the opposite carriageway of a motorway). Each LWSN is composed by at least one MN and a variable number of SNs. Let a *segment* be an array of regularly spaced adjacent SNs of the same LWSN. Each segment is associated with one or at most two MNs, the right and the left one.

The proposed MAC and routing protocol are described and the performance are presented in the following sections.

A. MAC Layer Protocol

The proposed MAC protocol is characterized by the state diagram shown in Figure 11(a).

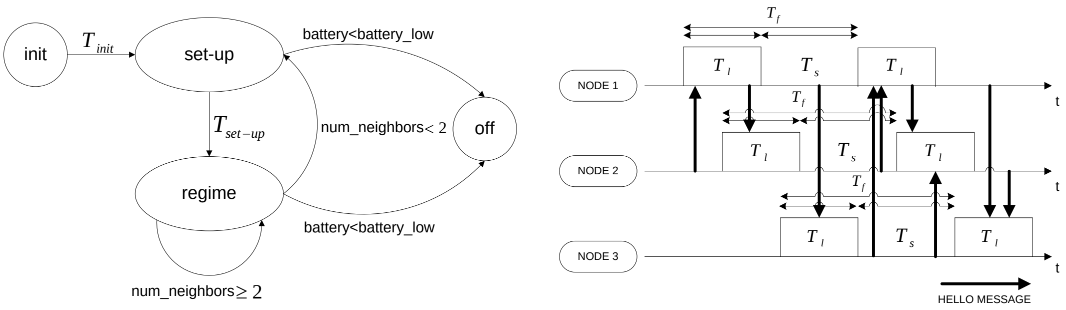
According to it, each node (master/sensor node) wakes up independently, entering an initial idle state (*init state*) in which it remains for the time interval necessary for performing the elementary CPU operations and to be completely switched on (T_{init}). Moreover, before entering the set-up state, each node starts to organize the time into frames whose durations are T_f.

In the *set-up state* each node tries to identify its neighbors and to establish a time synchronization with them. To this purpose it remains in a *listening mode* for a time interval equal to $T_{set-up} \geq 2T_f$ and begins to periodically broadcast a HELLO message sending its *ID* and its *phase*. The phase is the time interval after which the sender exits from the set-up state and enters the regime state. A node that receives a HELLO message adds the source node to the list of its own active neighbors and transmits an acknowledgement.

Once the set-up state has expired, each node enters the regime state. Within this state the operation mode is duty cycled with a periodic alternation of listening and sleeping sub-periods whose time intervals are T_l and T_s respectively. The duty cycle function is given by the following formula:

$$d = \frac{T_l}{T_f} = \frac{T_l}{T_l + T_s} \quad (6)$$

In the *regime state* each node is updated and tries to preserve the synchronization with its neighbors. To this purpose, as Figure 11(b) shows, it sends a frame-by-frame HELLO message in a unicast way to the active nodes in its list according to the phase transmitted by them in previous HELLO messages. As in the set-up state, the HELLO



(a) Finite state machine description of the proposed MAC protocol, involving the transitions occurring among *init*, *set-up*, *regime* and *off* states.

(b) MAC protocol HELLO messages exchange.

Figure 11. MAC protocol description

message contains the *ID* and the *phase* that, in this case, is the time interval after which the sender claims to be again in the listening status waiting for the HELLO message. The phase ϕ is evaluated according to the following rule:

$$\phi_1 = \tau - T_l \tag{7}$$

if the node is in the sleeping mode, where τ is the time remaining to the beginning of the next frame. Conversely, if the node is in the listening status, ϕ is computed as:

$$\phi_2 = \tau + T_s \tag{8}$$

The channel access is managed using the Carrier Sense Multiple Access with the Collision Avoidance (CSMA/CA) scheme, as specified in [11]. This mechanism is very effective in reducing collisions, while the problem of hidden nodes [12] is still partially unsolved.

Each node remains in the regime state until there are at least two neighbors, otherwise it reenters the set-up state in search of connectivity.

To complete the protocol characterization, whenever a node battery is depleted, this node turns off entering the *off state*.

In order to fully characterize the proposed MAC approach, the energy cost per frame interval of a single node (master/sensor node) can be evaluated as follows:

$$C = c_{rx}dT_f + c_{sleep}[T_f(1-d) - NT_{pkt}] + NC_{tx} \quad [mAs] \tag{9}$$

where c_{sleep} and c_{rx} represent the sleeping and the receiving costs $[mA]$ and C_{tx} is the single packet transmission costs $[mAs]$, T_{pkt} is the HELLO packet time length [s] and finally N is the number of neighbors.

In Figure 12, the energy cost per frame interval is shown of each single node as a function of the number of its neighbors. The considered parameters, summarized in Table I, are those relative to the real hardware platform. Moreover, in Figure 12 the accuracy of (9) is highlighted: the analytical results are similar to those obtained with the simulation model developed through the network protocol simulator *NePSing* [14].

Table I
DATA SHEET PARAMETERS OF THE CONSIDERED HARDWARE PLATFORM

Parameter	Symbol	Value
Frame interval	T_f	30 s
Listening interval	T_l	500 ms
Duty cycle	d	1.6 %
Sleeping cost	c_{sleep}	100 μA
Receiving cost	c_{rx}	25 mA
Packet transmission cost	C_{tx}	0.148 mAs

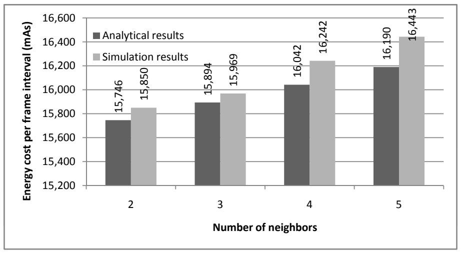


Figure 12. Single node energy cost per frame interval

Finally the protocol is compared with S-MAC and Wise MAC in terms of current consumption as the number of neighbour nodes changes. The Figure 13 highlights the computed performance.

B. Routing Layer Protocol

In order to evaluate the capability of the proposed MAC scheme in establishing effective end-to-end communications within each LWSN, a routing protocol was introduced and integrated according to the *cross layer* design principle [13]. Periodical information is sent which is needed for building and maintaining the local routing tables depicted in Table II.

It resorts to the signaling introduced by the MAC layer with the aim of minimizing the overhead and making the system more adaptable in a cross layer fashion. In particular,

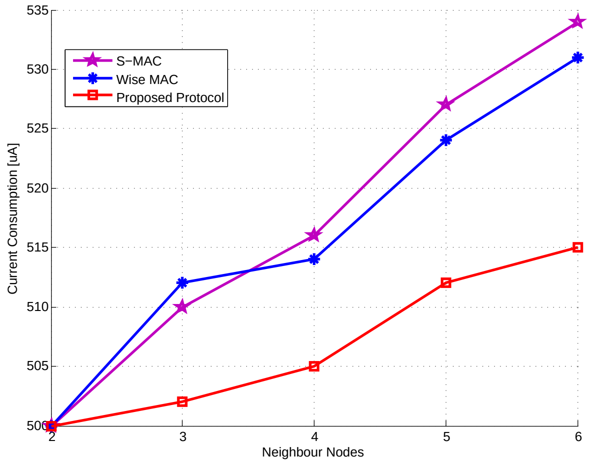


Figure 13. Current Consumption vs. Neighbour Nodes

Table II
ROUTING TABLE GENERAL STRUCTURE

Master Node	Next Hop	Hop Count	Loop Flag
MN_1	SN_1	η_A	false
MN_1	SN_2	η_B	true
MN_2	SN_3	η_C	true

the parameters transmitted along a MAC HELLO message, with period T_f, are the following:

- *ID destination*. If the sender node is in the set-up state, the ID destination will be the broadcast one; otherwise, it will be the ID of the receiver node.
- *ID source*. The ID source is the ID of the sender node.
- *phase*. If the sender node is in the set-up state, the phase will be the time interval after which the sender node exits from the set-up state and enters the regime state; otherwise, it will be the schedule time at which the sender node enters in listening mode according to (7) and (8).
- $ID\ MN_1$. If the sender node is a SN, the $ID\ MN_1$ will be the ID of the first MN which the SN is associated with; otherwise if the sender node is a MN, it will be set at 0.
- *Next Hop 1 (NH_1)*. If the sender node is a SN, the NH_1 will be the ID of the next hop neighbor used by the SN to reach the MN_1 with the minimum number of hops; otherwise if the sender node is a MN, it will be set at 0.
- *Hop Count 1 (HC_1)*. If the sender is a SN, the HC_1 will be the distance from the MN_1 in terms of minimum number of needed hop; otherwise if the sender node is a MN, it will be set at 0.
- $ID\ MN_2$. If the sender node is a SN, the $ID\ MN_2$ will be the ID of second MN which the SN is associated with; otherwise if the sender node is a MN, it will be set at 0.
- *Next Hop 2 (NH_2)*. If the sender node is a SN, the NH_2 will be the ID of the next hop neighbor used by the SN to reach the MN_2 with the minimum number of hops; otherwise if the sender node is a MN, it will be set at 0.
- *Hop Count 2 (HC_2)*. If the sender is a SN, the HC_2 will be the distance from the MN_2 in terms of minimum number of needed hop; otherwise if the sender node is a MN, it will be set at 0.

In the *init state*, each SN sets NH_1 and NH_2 at 0 and HC_1 and HC_2 at a high value chosen a priori.

In the *set-up state*, each LWNS begins to self-organize. Starting from the MNs, the routing information is flooded through the network by each SN.

The routing table is filled up or updated by each node according to the following rules.

As an SN receives a HELLO message from an MN it is associated with, it inserts a row in its routing table (or updates an existing one) assigning the ID source to the ”Master Node” and the ”Next Hop” fields, a value equal to 1 to the ”Hop Count” field and the false value to the ”Loop Flag” field.

As an SN receives a HELLO message from an SN belonging to the same segment.

1) it inserts a row in its routing table (or updates an existing one) assigning $ID\ MN_1$ to the ”Master Node” field, the ID source to the ”Next Hop” field and HC_1 to the ”Count Hop” field;
2) if NH_1 parameter contains its own ID, it will assign the true value to the ”Loop Flag” field; otherwise it increments by 1 the ”Hop Count” field and assigns the false value to the ”Loop Flag” field.

The same procedure is ran for the $ID\ MN_2$, NH_2 and HC_2 parameters.

As an SN receives a HELLO message from an SN belonging to an adjacent segment, it discards the information.

As an MN receives a HELLO message from an SN belonging to the left or the right segment, it discards the information concerning itself and stores the other ones in its routing table.

As a node receives a HELLO message from a node belonging to the opposite LWSN, it stores the information in its routing table without any control or preprocessing.

Finally, in the *regime state* each node updates its routing table frame by frame thanks to the reception of HELLO messages from its neighbors. A node deletes an active neighbor from its routing table if it does not receive any acknowledgment for three consecutive frames after the transmission of HELLO messages.

Figure 14 graphically shows the procedure described above for a simple LWSN composed by one segment. In this example it is suppose that each node communicates only with the two adjacent nodes. Let us consider the node labeled SN_1. During the first frame it receives two HELLO messages, one from MN_1 and the other from SN_2 (see Figure 15). Then it updates its routing table. Only the first row contains useful information. According to the network topology taken into the account, SN_1 has to wait the second frame to complete its routing table and know the next hop to reach the MN_2. During the third frame a loop is verified.

From MN1

ID dest	ID source	phase	ID MN1	NH1	HC1	ID MN2	NH2	HC2
0	MN1	5	0	0	0	0	0	0

From SN2

ID dest	ID source	phase	ID MN1	NH1	HC1	ID MN2	NH2	HC2
0	SN2	2	MN1	SN2	infty	MN2	SN2	infty

Figure 15. HELLO messages sent by MN_1 and SN_2 during the first frame

Once the routing table has been filled, each node may derive the proper metric depending on the type of the application message that it has to manage. The application messages are subdivided into two categories: *query messages*, sent by the MNs to query the SNs, and *response messages*, sent by the SNs in response to a query message. If an SN has to send a response message, it will select from its routing table the next hop neighbor that has the ”Master Node” field equal to the ID of the destination MN, the minimum ”Hop Count” value and the ”Loop Flag” field set at false. Then it forwards the message to it. This procedure is performed by every SN received. If an SN receives a query message sent by an MN it is associated with, it will send the message to every neighbor that belongs to the same segment and has the ”Master Node” field equal to the ID of the sender MN, the ”Loop Flag” set to true and the ”Hop Count” value higher than its own MN distance value.

A *recovery state* is introduced to provide a fault-tolerant communication. If an SN does not find any neighbor of the same segment for establishing an end-to-end communication with one of the MNs it is associated with, it will send a HELP message to all the active neighbors of the opposite LWSN. Therefore they look for an alternative path to reach the involved MN, trying to establish a link with a node located in the same segment of the calling-for-help node.

VI. Experimental Results

A prototype, composed of a basic unit of the system has been deployed near Florence, along the A11 highway operated by Autostrade per l’Italia SpA (ASPI) in order to obtain on-field testing and evaluation (Figure 16). The system has been placed closely to a loop detector to test the MN functionality.

The MN unit was first deployed on May 2009; since then it has undergone extensive operation, regularly collecting and transmitting traffic flow reports for the central server at a 60 second rate. Various data typologies were collected by the system to monitor the traffic flow, jams or queues. Some results are provided in the following figures.

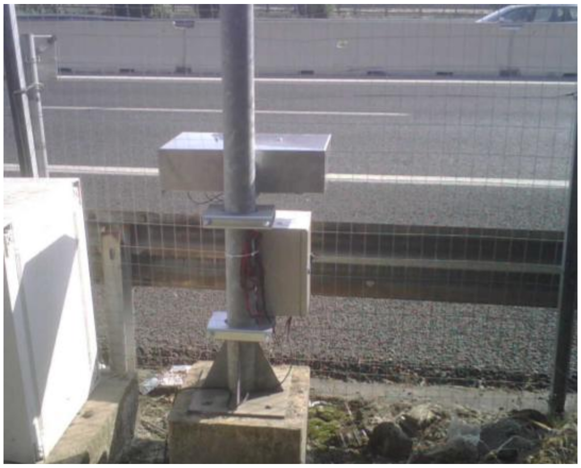


Figure 16. Prototype system photograph.

In Figure 17, a weekly data collection of vehicle transit and average speeds is shown, highlighting the periodicity of the traffic flow with different behavior depending on the day and hours. The MN yield is validated by the comparison with the loop detector (Figure 18). As it is shown, the two systems have collected the same results in terms of shape and vehicle transit; only slightly differences can be evaluated comparing the two graph. During the night, in fact, there is an overestimation of the vehicle transit due to a preliminary setup of the system related to the noise floor site.

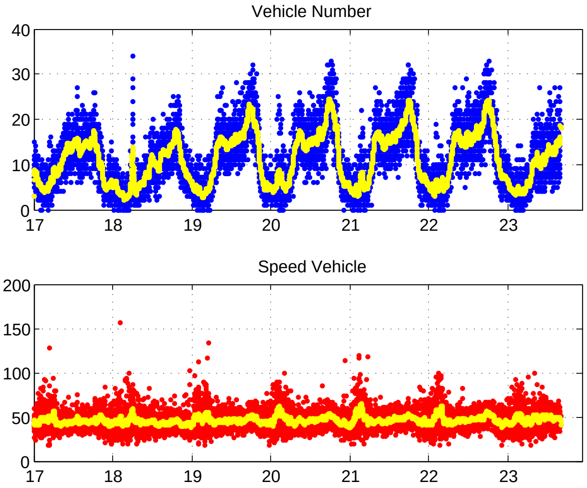
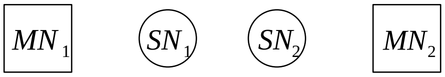


Figure 17. Vehicle transit and average speed for a weekly observation slot.

As a consequence, the MN information is now fully integrated into the ASPI information system; Figure VI



SN1 ROUTING TABLE **1st frame** SN2 ROUTING TABLE

Master Node	Next Hop	Hop Count	Loop Flag
MN1	MN1	1	false
MN1	SN2	infty	false
MN2	SN2	infty	false

Master Node	Next Hop	Hop Count	Loop Flag
MN2	MN2	1	false
MN2	SN1	infty	false
MN1	SN1	infty	false

MN1 ROUTING TABLE MN2 ROUTING TABLE

Master Node	Next Hop	Hop Count	Loop Flag
MN2	SN1	infty	false

Master Node	Next Hop	Hop Count	Loop Flag
MN1	SN2	infty	false

SN1 ROUTING TABLE **2nd frame** SN2 ROUTING TABLE

Master Node	Next Hop	Hop Count	Loop Flag
MN1	MN1	1	false
MN1	SN2	infty	false
MN2	SN2	2	false

Master Node	Next Hop	Hop Count	Loop Flag
MN2	MN2	1	false
MN2	SN1	infty	false
MN1	SN1	2	false

MN1 ROUTING TABLE MN2 ROUTING TABLE

Master Node	Next Hop	Hop Count	Loop Flag
MN2	SN1	infty	false

Master Node	Next Hop	Hop Count	Loop Flag
MN1	SN2	infty	false

SN1 ROUTING TABLE **3rd frame** SN2 ROUTING TABLE

Master Node	Next Hop	Hop Count	Loop Flag
MN1	MN1	1	false
MN1	SN2	2	true
MN2	SN2	2	false

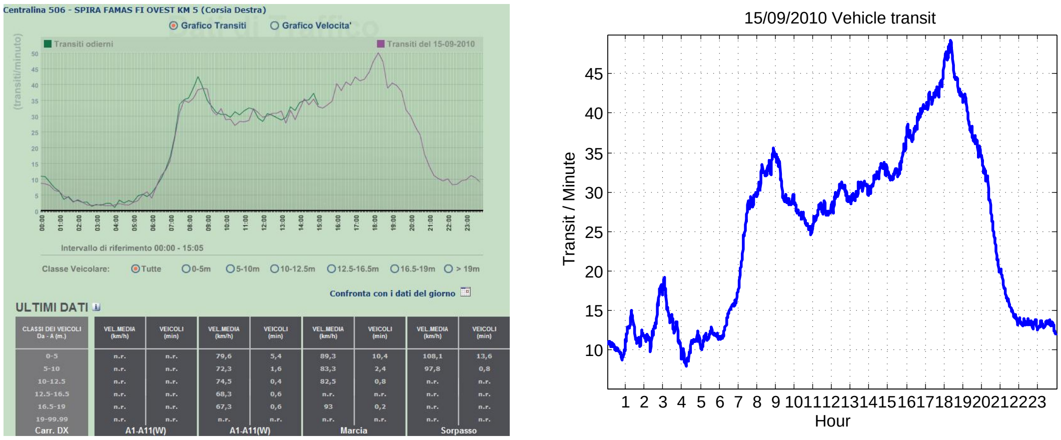
Master Node	Next Hop	Hop Count	Loop Flag
MN2	MN2	1	false
MN2	SN1	2	true
MN1	SN1	2	false

MN1 ROUTING TABLE MN2 ROUTING TABLE

Master Node	Next Hop	Hop Count	Loop Flag
MN2	SN1	3	false

Master Node	Next Hop	Hop Count	Loop Flag
MN1	SN2	3	false

Figure 14. Example of Routing Tables


Centralina 506 - SPIRA FAMAS FI OVEST KM 5 (Corsia Destra)

● Grafico Transiti ○ Grafico Velocita'

Intervallo di riferimento 00:00 - 15:05

Classe Veicolare: ● Tutte ○ 0-5m ○ 5-10m ○ 10-12.5m ○ 12.5-16.5m ○ 16.5-19m ○ > 19m

Confronta con i dati del giorno

ULTIMI DATI

CLASSI DEI VEICOLI Da - A (m.)	VEL.MEDIA (km/h)	VEICOLI (min)	VEL.MEDIA (km/h)	VEICOLI (min)	VEL.MEDIA (km/h)	VEICOLI (min)	VEL.MEDIA (km/h)	VEICOLI (min)
0-5	n.r.	n.r.	79,6	5,4	89,3	10,4	108,1	13,6
5-10	n.r.	n.r.	72,3	1,6	83,3	2,4	97,8	0,8
10-12.5	n.r.	n.r.	74,5	0,4	82,5	0,8	n.r.	n.r.
12.5-16.5	n.r.	n.r.	68,3	0,6	n.r.	n.r.	n.r.	n.r.
16.5-19	n.r.	n.r.	67,3	0,6	93	0,2	n.r.	n.r.
19-99.99	n.r.	n.r.	n.r.	n.r.	n.r.	n.r.	n.r.	n.r.
Carr. DX	A1-A11(W)		A1-A11(W)		Marcia		Sorpasso	

(a) Loop Detector.

(b) Audio System.

Figure 18. Comparison between audio sensor and loop detector results.

shows the ASPI public user interface where information from various sensors typologies are available. In particular, a summary report obtained by the MN is presented. Currently the ASPI interface is not able to present SN reports in graphic format; this work is still in progress.

Extensive tests during the period of operation have provided the motorway practitioners with a complete report on traffic trends. Due to the yield and easy deployment of the system, a $50\ km$, dual carriageway complete installation is planned by ASPI to fully exploit the potential of the system in the A1 motorway, between Florence and Arezzo.

VII. Conclusion

In this paper, a novel sensor network architecture and communication protocol for traffic surveillance has been proposed, exhibiting the unique feature of providing a complete and immediate traffic flow status in real-time at an unprecedented scale. The main features are low installation

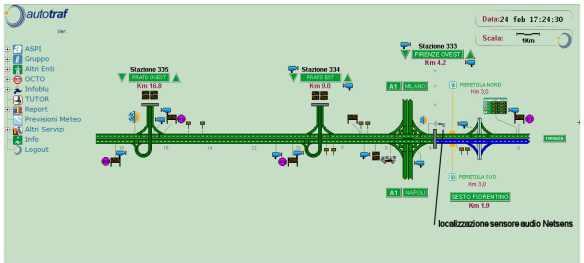


Figure 19. ASPI user interface.

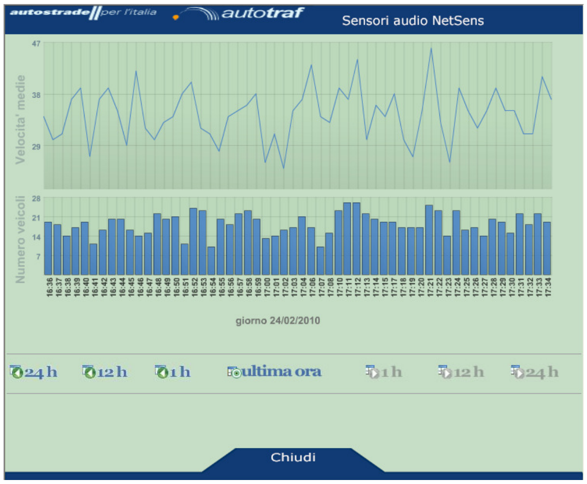


Figure 20. Daily report.

and maintenance costs due to the sensing elements based on the use of passive acoustic transducers. Experimental results, coming from long-term testing performed on a motorway, have demonstrated the effectiveness and yielding of the approach for providing traffic authorities with detailed information about traffic parameters. Thanks to the promising results obtained by the pilot site, ASPI has approved an extensive system installation along the A1 motorway.

REFERENCES

[1] Barbagli, B.; Magrini, I.; Manes, G.; Manes, A.; Langer, G.; Bacchi, M.; , ”A Distributed Sensor Network for Real-Time Acoustic Traffic Monitoring and Early Queue Detection,” Sensor Technologies and Applications (SENSORCOMM), 2010 Fourth International Conference on , vol., no., pp.173-178, 18-25 July 2010 doi: 10.1109/SENSORCOMM.2010.102

[2] Klausner, A.; Rinner, B.; Teng, A.; , ”I-SENSE: Intelligent Embedded Multi-Sensor Fusion,” Intelligent Solutions in Embedded Systems, 2006 International Workshop on , vol., no., pp.1-12, 30-30 June 2006 doi: 10.1109/WISES.2006.329120

[3] Klausner, A.,Erb, S.,Tengg, A.,Rinner, B.:DSP Based Acoustic Vehicle Classification for Multi-Sensor Real-Time Traffic. Graz University of Technology, Graz, Austria.

[4] Forren, J.F.; Jaarsma, D.; , ”Traffic monitoring by tire noise,” Intelligent Transportation System, 1997. ITSC ’97., IEEE Conference on , vol., no., pp.177-182, 9-12 Nov 1997 doi: 10.1109/ITSC.1997.660471

[5] Ding, J.; Cheung, S.-Y.; Tan, C.-W.; Varaiya, P.; , ”Signal processing of sensor node data for vehicle detection,” Intelligent Transportation Systems, 2004. Proceedings. The 7th International IEEE Conference on , vol., no., pp. 70- 75, 3-6 Oct. 2004 doi: 10.1109/ITSC.2004.1398874

[6] Cheung, S. ,Coleri, S. ,Varaiya, P.:Traffic Surveillance with Wireless Magnetic Sensors. University of California, Berkley. USA

[7] Knapp, C.; Carter, G.; , ”The generalized correlation method for estimation of time delay,” Acoustics, Speech and Signal Processing, IEEE Transactions on , vol.24, no.4, pp. 320- 327, Aug 1976 doi: 10.1109/TASSP.1976.1162830

[8] Shiping Chen; Ziping Sun; Bridge, B.; , ”Traffic monitoring using digital sound field mapping,” Vehicular Technology, IEEE Transactions on , vol.50, no.6, pp.1582-1589, Nov 2001 doi: 10.1109/25.966587

[9] Brockmann, E.M.; Kwan, B.W.; Tung, L.J.; , ”Audio detection of moving vehicles,” Systems, Man, and Cybernetics, 1997. ’Computational Cybernetics and Simulation’., 1997 IEEE International Conference on , vol.4, no., pp.3817-3821 vol.4, 12-15 Oct 1997 doi: 10.1109/ICSMC.1997.633265

[10] Al-Karaki, J.N.; Kamal, A.E.; , ”Routing techniques in wireless sensor networks: a survey,” Wireless Communications, IEEE , vol.11, no.6, pp. 6- 28, Dec. 2004 doi: 10.1109/MWC.2004.1368893

[11] 802.15.4-2003: part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs). IEEE Std., October 2003. [Online]. Available: www.ieee.org.

[12] Mohapatra, P., Krishnamurthy, S.V.: Ad Hoc Networks-Technologies and Protocols. Springer Science & Business Media Inc., 2005.

[13] Shakkottai, S.; Rappaport, T.S.; Karlsson, P.C.; , ”Cross-layer design for wireless networks,” Communications Magazine, IEEE , vol.41, no.10, pp. 74- 80, Oct 2003 doi: 10.1109/MCOM.2003.1235598

[14] Pecorella, T.: NePSing - Network Protocol Simulator ng. [Online] Available: http://nepsing.sourceforge.net/.

Enhancing Resiliency Against Routing Layer Attacks in Wireless Sensor Networks: Gradient-based Routing in Focus

Ochirkhand Erdene-Ochir, Apostolos Kountouris
Orange Labs
France Telecom Group
38243 Meylan, France
Email: {ochirkhand.erdeneochir,apostolos.kountouris}
@orange-ftgroup.com

Marine Minier, Fabrice Valois
Universite de Lyon, INRIA
INSA-Lyon, CITI
F-69621 Lyon, France
Email: {marine.minier,fabrice.valois}@insa-lyon.fr

***Abstract*—This paper focuses on the resiliency of wireless sensor network routing protocols against *selective forwarding* attacks by compromised nodes. Informally, resiliency should be understood as the capacity of the routing protocol to endure and mitigate the presence of a certain number of compromised nodes seeking to disturb the routing process. To provide for security when nodes may be compromised, cryptographic solutions must be completed by algorithmic solutions considering "beyond cryptography" approaches. In this article, after discussing the shortcomings of existing routing protocols against packet-dropping malicious nodes we describe some protocol behaviors enhancing routing resiliency under several combined routing attacks. These behaviors are mainly based on traffic redundancy and probabilistic selection for the next hop candidates, which permit to exploit and benefit from the inherent structural redundancy of densely deployed sensor networks. We consider the case that compromised nodes, prior to selective forwarding, and seeking to increase its impact, may perform several well known routing attacks such as Sinkhole, Sybil and Wormhole. Several variants of the well known gradient-based routing protocol were tested and simulation results show that using the proposed techniques resiliency can be improved. Nevertheless, as expected, resiliency comes at a cost and our results also shed some light on the resiliency-energy consumption trade-off. We propose in this paper the behaviors enhancing the resiliency of routing protocols under several combined routing attacks.**

***Keywords*-wireless sensor networks, routing, security, attacks, resiliency, reliability.**

I. Introduction

In typical Wireless Sensor Network (WSN) applications, a large number of resource constrained sensor nodes are deployed over a geographic area in order to collect physical world data and route them towards one or few destinations (data sinks). The rapid deployment capabilities, due to the lack of infrastructure, as well as the self organized and potentially fault-tolerant nature of WSNs make them attractive for multiple applications spanning from environmental monitoring (temperature, pollution, etc.) to building-industrial automation (electricity/gas/water metering, event detection, home automation etc.). In recent years WSNs have emerged as a very active as well as challenging research area in search for solutions to the open problems of scalability, adaptability, low energy consumption and security. In WSNs the difficulty of all these problems is exacerbated by the large numbers and the resource constrained nature of sensor nodes.

Security is particularly challenging in WSNs. Because of their open and unattended deployment, in possibly hostile environments, adversaries can easily launch Denial-of-Service (Dos) attacks, cause physical damage to sensors, or even capture them to extract sensitive information like for instance encryption keys, identities, addresses etc. Consequently node compromise poses severe security and reliability concerns since it allows an adversary to be considered as a legitimate node inside the network. After node compromise, an adversary can seek to disrupt the functionality of routing layer by launching attacks such as node replication, Sybil, Selective forwarding, Sinkhole or Wormhole. To cope with these "insider" attacks, stemming from node compromise, "beyond cryptography" algorithmic solutions must be envisaged to complement the cryptographic solutions for secure routing proposed in [1], [2], [3], [4]. The work presented in this paper is an extension of our first exploratory work [5] in this direction.

In the existing literature, papers often focus on a particular attack proposing ways to detect and to defend against it mainly by excluding malicious nodes [6], [7], [8], [9]. In this paper, we have chosen to follow a different path; we believe the difference is significant enough to justify the need for further study. Our main goal is not to detect attacks and eliminate malicious nodes, but rather to make the routing protocol capable to continue routing packets, at acceptable success rates, in the presence of malicious nodes. This routing protocol capability will be referred to as resiliency. Also, it is worth mentioning that even though routing resiliency is studied using the Selective forwarding attack as basis of our attack model, interestingly this attack is combined with several other well known routing attacks such as Sinkhole, Sybil and Wormhole. Such combinations represent more realistic attack situations than simply considering each attack separately. Finally, since we deal with "insider" attacks, malicious compromised nodes have access to the same information as honest nodes in agreement with Shannon's maxim: "The enemy knows the system". Therefore, malicious nodes, aware of defensive strategies against attacks, are expected to adapt their own strategies to optimize the impact of their attacks. From this standpoint our goal is also to dissuade an adversary from creating adapted attack strategies and just settle for basic (random) Selective forwarding.

It should be noted that we believe that if an attacker has decided to break down the network he will succeed by assuming the necessary cost. The required investment depends on cost-benefit analysis considerations quantifying the adversary's interest in breaking down the network. Under such a worst case scenario protocol resilience will not be effective. However, this is also the case of other approaches, like for instance detecting and isolating malicious nodes. Even if a source node is capable of detecting and isolating malicious neighbors, the packet will not reach the sink if most of its neighbors are compromised. We also show in our simulation results that under Sinkhole attacks where most of the compromised nodes are located around the sink, the sink becomes almost completely disconnected from the rest of the network which in practice is equivalent to the sink being compromised. In what follows we assume that an adversary can compromise only a limited number of sensor nodes, since compromising a node has some cost. In other words, mass attacks, i.e., a large number of both insider and outsider attackers, are out of the scope of this paper. Our main goal is to render a network inherently resilient in the presence of a few malicious nodes, we therefore require that the network performance degrades gracefully as the number of compromised nodes increases. Numerous business applications such as periodic monitoring of electricity, gaz, water metering, and environmental monitoring, manipulate some important but not highly sensitive data. In these non mission critical cases, we assume that an adversary has limited power.

The rest of the paper is organized as follows. Section II, provides an overview of previous work insisting on information, which is relevant in the context of this paper; for instance, routing resiliency and its relationship to other similar notions such as survivability and robustness are discussed and it is explained why classical shortest path routing protocols are not resilient against insider attacks. In Section III, we illustrate our adversary model including network assumptions, adversarial definitions and several routing attacks considered in this paper. We then propose, in Section IV, several probabilistic node selection and packet replication strategies, which improve resiliency by making protocol behavior dynamic and redundant in order to exploit the inherent structural redundancy in the topology of densely deployed WSNs. In Section V we present our approach by mixing and applying these strategies to the well known Gradient-based routing protocol (GBR) [10]; simulations were performed for a basic Selective forwarding attack and for its combination with three other routing attacks, namely Sinkhole, Sybil and Wormhole attacks. Finally, Section VI concludes this paper and outlines future work directions.

II. Scope and related work

In this paper, we focus on the Selective forwarding attack where compromised nodes drop data packets. This attack is not only simple but it can be very effective as well. When multi-hop packet routing is considered even a small number of packet-dropping nodes can significantly deteriorate the packet delivery rate of the network. Furthermore, when several routing attacks such as Sinkhole, Sybil and Wormhole are considered in combination with Selective forwarding, this enables adversary nodes to attract more traffic and so amplify the impact of malicious packet dropping.

In this Section a rather rapid overview of previous work is given with the purpose of introducing relevant terminology, concepts and open issues. The vastness of the literature from one side and space limitation from the other do not permit to be more exhaustive but hopefully this brief discussion will help the reader situate our proposal within this research context.

A. *Routing layer attacks and countermeasures*

Attacks at the network layer were summarized in [11] as follows: (a) spoofed, altered or replayed routing information; (b) Selective forwarding, node replication, Sybil, Sinkhole or Wormhole and `HELLO` packets flooding. `HELLO` packets are special control packets sent by each node for neighborhood discovery. We are mainly interested on the attacks of the second type targeting the routing layer. After node compromise an adversary can extract all sensitive information stored in the node. Other attacks such as radio jamming, exhaustion, collisions, link layer jamming or attacks against data aggregation are out of the scope of this paper since they do not directly target the network layer.

In Selective forwarding, malicious nodes simply drop some packets (Greyhole) or all of them (Blackhole) instead of forwarding them as they are supposed to. The main principle of the Sinkhole attack is exactly the same except that the compromised nodes are, or pretend to be, near the sink to attract most of the traffic. After a successful Sinkhole attack the adversary performs Selective forwarding. One possible solution is to use traffic monitoring to ensure that neighbor nodes forward the messages. In [7], the authors propose to use a Watchdog scheme that identifies selfish nodes and a Pathrater scheme that helps routing protocols avoid such nodes. The Watchdog scheme is further extended by a reputation based scheme, [12], where the neighbors of any single node collectively rate the node according to how well the node executes the functions requested upon it. Alternatively an acknowledgment based scheme was proposed in [6] to detect maliciously packet dropping nodes.

In the Sybil attack, [8], a malicious device illegitimately takes on multiple identities. Doing so the malicious nodes can fill up their neighbors' buffers with non existing neighbors and thus create a false topology. Another way to exploit node capture is the node replication/cloning attack [13], where an adversary replicates several nodes with the same identity at different places in the network. To defend against the Sybil attack, the network needs some mechanism to validate that a particular identity is the only identity being held by a given physical node. In [8] the authors describe resource tests and in [13] two distributed algorithms are proposed: randomized multicast and line-selected multicast exploiting the birthday "paradox" to defend against node replication.

Finally, the Wormhole attack, [9], occurs when an attacker receives packets at one location in the network and tunnels them to another, via an out-of-band connection, in order to replay them at this other location. This attack creates a totally false network topology. A Wormhole detection mechanism, called packet leashes, is introduced in [9] and is based on distance estimation; it consists in two mechanisms: geographical leashes and temporal leashes. Another technique to defend against Wormhole attacks consists in using directional antennas [14].

As a conclusion it can be said that most of the proposed solutions strive to defend against a single attack adopting a two stage approach: first, actively detect malicious nodes and second defend against the attack by avoiding routing traffic through the detected malicious nodes. In contrast our aim is not to detect and defend against a single attack, as previously done, but rather to limit damages when several routing attacks are combined together. We propose to do so by enhancing the resiliency of the routing protocols.

B. Resiliency and related notions

According to Webster [15] in mechanics, resiliency is the capability of a strained body to recover its size and shape after deformation caused especially by compressive stress. Hinging upon the general dictionary definition and after reviewing the multiple definitions of resiliency and other similar notions in networking, we define the resiliency in [16] as the ability of a network to "continue to operate" in presence of k compromised nodes, or in other words, the capacity of a network to endure and overcome internal attacks. Simply put, resiliency is a means to achieve a "graceful degradation" in packet delivery rate with increasing number of compromised nodes.

In the literature, several conceptually similar properties such as survivability [17] and robustness [18], have been discussed but mainly focus on system failures from causes of pure statistical nature contrary to attacks where there is some behind-the-scenes entity with malicious intention. Furthermore, the notion of enduring and overcoming an attack (failure) is not explicitly considered. Finally, resiliency as discussed in [19], [20], [21] is not applied in secure routing but in contexts like robust data aggregation, fault-tolerant routing and key distribution schemes respectively. Nevertheless it should be noted that as [19] compares the resiliency of aggregation functions, our aim is to compare the resiliency of several versions of a given routing protocol.

C. Deterministic routing and its limitations

Insofar minimizing power consumption has been considered a top priority in WSNs research. For increased efficiency, most of the routing protocols use a shortest path criterion to route DATA packets the goal being to reach the sink as quickly as possible. Reactive routing, such as Dynamic Source Routing (DSR) [22], geographical routing, such as Greedy Forwarding (GF) [23] and gradient-based, such as Gradient-Based Routing (GBR) [10], all employ a shortest path principle (with some appropriate definition of "short"). Unfortunately this underlying shortest-path optimization philosophy is responsible for the severe limitations of deterministic routing protocols when attacks involving compromised nodes are considered.

To facilitate discussion lets suppose that some insider attacker has compromised a number of nodes, which are uniformly distributed across the network and which drop all DATA packets they receive. If l denotes the path length in number of hops from source to destination; p_c denotes the probability that a node is compromised and p_n is the probability that a packet is delivered (i.e., all forwarding nodes on the route are legitimate), we have $p_n = (1-p_c)^l$. In this case, the probability to find a "safe" route exponentially decreases with route length; essentially the same applies for Selective forwarding attacks where only part of the traffic is dropped.

In the presence of such attacks, the routing protocols using shortest paths have better overall delivery ratio. However, they are not resilient. First, as the routes are static all DATA packets from a source node take always the same route to reach a sink. Therefore, if at least one intermediate node is compromised along a route, all DATA packets will be lost and the source node will be completely disconnected from the sink. Second, if a source node has at least one malicious neighbor who will try to attract the traffic (best delay, best gradient, geographically closest to the sink etc.), all DATA packets will be engulfed by such a compromised node. Thus, the routing protocol as is will not be able to overcome this situation since the compromised node will always seem the best routing choice to make.

In previous work, a configurable secure routing protocol (SIGF) has been proposed in [24], extending geographical routing (IGF) [25] with the intention of adding security and protection against outsider attacks. It advocates for an incremental approach to security. As a basis SIGF uses nondeterminism in neighbor selection, then it adds a reputation scheme and finally it considers cryptography. In a sense SIGF strives for a layered resistance to attacks. However, reputation schemes cannot defend against colluding malicious nodes and cryptographic primitives cannot defend against node compromise. Our aim is to contribute in a similar way by considering, as in SIGF, nondeterminism as a basis of protocol behavior but in our case striving for resiliency, instead of resistance, to several combined attacks, which is more appropriate when compromised possibly colluding nodes are considered.

III. Network assumptions and Adversary models

In this section we state the network assumptions and several adversarial definitions and we describe the implemented routing attacks.

A. Network assumptions

In the following two types of network device nodes are considered: ordinary sensors and data sinks. Sensor nodes sense and transmit data of the physical world to a single data collector, the sink. Here we deal with WSNs where all sensor nodes are physically identical in terms of transmission range, power, etc. Sensor nodes are densely deployed in a square region of size $N \times N$ and the physical topology of the network is represented by a connected graph. The packets are routed from the source (sensors) to the destination (sink) on this topology.

A common and practical graph model proposed for modeling WSNs is the fixed radius random graph. Let us consider a graph $G(\Omega, E)$ where Ω is a set of nodes wirelessly connected pairwise by a set of E of undirected edges to represent communication links between nodes. In this model, the nodes are randomly placed in a $N \times N$ region according to a uniform distribution. A link exists between two nodes i and j if the Euclidean distance between these two nodes less than the communication range r. We assume that the wireless links in our graph are bi-directional, i.e., if node i hears node j then node j also hears node i.

In addition, from the network security standpoint we use the following, traditionally made, assumptions:

- the "sink" is considered robust, having enough resources in terms of memory, computational power and battery to support the cryptographic and routing requirements of the WSN. Thus, adversaries cannot compromise the sink in limited time.
- the "sensor" has limited resources in terms of memory, computational power and battery. Thus, sensor nodes are non trustworthy since they are vulnerable to physical attacks and an adversary can compromise them.

B. Adversarial definitions

According to [26] an attack is an intentional act by which an entity attempts to evade security services and violate the security policy of a system; that is, an actual assault on system security that derives from an intelligent threat.

According to its capabilities an attacker can be characterized as:

- A *laptop* class attacker: It may have access to powerful devices with more computational resources, such as laptops or their equivalent. A single laptop-class attacker might be able to eavesdrop and/or jam the entire network.
- A *mote* class attacker: It has access to a few motes with the same capabilities as other ordinary sensor nodes. They have no resource advantages over legitimate nodes.

Attacks can also be characterized according to intent as:

- A *passive* attack: In this attack, the adversary attempts to learn or make use of information from a system but does not affect system resources. For example, passive eavesdropping that simply gathers information, can compromise privacy and confidentiality.
- An *active* attack: It attempts to alter system resources or affect system operations. Compared to the passive attack, here the goal of the adversary is to produce DoS attacks to disrupt communication by destroying links or exhaust available resources such as bandwidth or energy.

Finally, attacks can be characterized according to point of initiation as:

- An *outsider* attack: It is initiated from outside the security perimeter by an unauthorized or illegitimate user of the system. Examples are external attacks such as jamming, eavesdropping as well as injecting replayed or fabricated messages.
- An *insider* attack: It is one that is initiated by an entity inside the security perimeter, i.e., an entity that is authorized to access system resources but uses them in a way not approved by the party that granted the authorization. Selective forwarding, Sybil, Sinkhole or Wormhole attacks being notable examples.

With respect to this classification and given our network assumptions our adversary model considers: "mote-class", "active", and, "insider" attackers.

C. Implemented routing attacks

An adversary will try to disrupt communication and cause as much as possible damage to routing protocols. To compare the resiliency of the different protocols, firstly, we modeled the basic Selective forwarding attack, and secondly, we combined it with three other routing attacks; Sybil, Wormhole and Sinkhole. In this sense our attack model is a two-stage combination of simpler attacks. At the first stage, the attacker will launch some attacks in order to enable compromised nodes to attract a lot of traffic. Subsequently at the second stage the compromised nodes will launch the

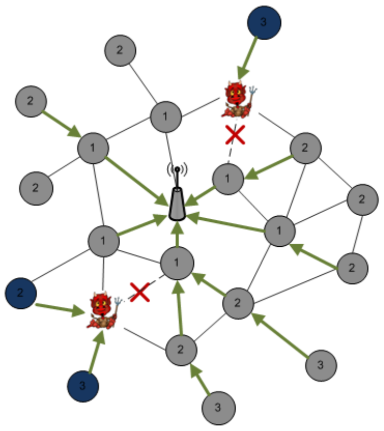


Figure 1. Basic Selective forwarding attack

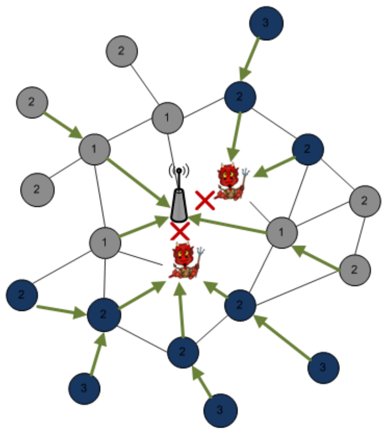


Figure 2. Combined Sinkhole attack

routing attack per se by performing *selective forwarding* on the attracted packets. We have considered the *selective forwarding* attack as a basis of our attack model not only because it is common to all protocols but also because this simple attack has a direct impact on reliable data delivery, which characterizes the success of routing protocols.

In the following the main constituents of our attack model will be described in more detail.

1) Basic attack: In multi-hop routing, messages may cross many hops before reaching their final destination. However, a malicious node in the path of data transmission can refuse to forward messages. Selective forwarding is a simple and basic routing attack easy for an insider adversary to launch. After node compromise, malicious nodes instead of forwarding messages with probability 1 they do so with some lower probability. For instance, they can drop all messages (probability to forward = 0) or they can selectively drop some of them in order to avoid detection of their malicious activity (Fig. 1).

2) Combined attacks: For more efficiency, an adversary can exploit its “insider” knowledge to first try to attract traffic and then drop it. Selective forwarding is effective when malicious nodes are on the routes of packet transfer so it is logical to consider it as the final stage of more complex attack behavior where malicious nodes firstly employ some other attack to advantageously place themselves on the routes of heavy traffic and then effect Selective forwarding. Hence, well known routing attacks such as Sybil, Wormhole, Sinkhole could be combined with basic Selective forwarding. This type of combined attacks is explicitly considered within our model.

For instance, to create a Sinkhole, an adversary will try to compromise nodes closer to the sink, exploiting knowledge of location information, to attract most of the traffic (Fig. 2). After a successful Sinkhole attack, the adversary will perform Selective forwarding. The nature of sensor networks where all the traffic flows towards one (or few) sink node(s) makes this type of attacks highly relevant.

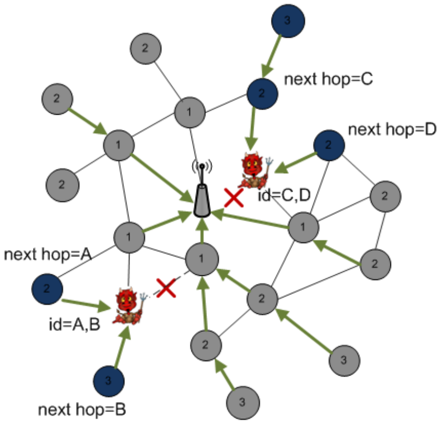


Figure 3. Combined Sybil attack

Sybil attack is defined by malicious nodes illegitimately taking on multiple identities (Fig. 3) thus compromising the neighborhood discovery process. For instance, a malicious node taking two or more identities will increase the probability of being selected by legitimate nodes as their next hop and then produce Selective forwarding to disrupt routing.

In Wormhole, a malicious node receives packets at one point in the network and tunnels them to another point via an out-of-band connection (Fig. 4). Thus, two malicious nodes can make believe that they are neighbors even if they are physically distant. Well placed Wormholes, for instance an adversary closer to the sink, make possible to attract the traffic of the two hop neighborhood. Wormhole attack is particularly dangerous against routing protocols not only because it creates false topologies but also it permits to attract effectively the traffic. It should be noted that Wormholes is also an effective means to create Sybil identities using existing identities in case legitimate nodes can detect fabricated or duplicated identities.

In the remainder of this paper we propose some routing

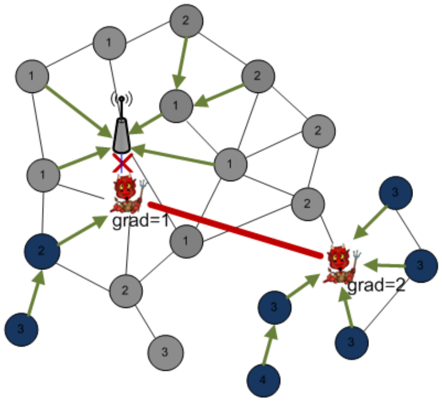


Figure 4. Combined Wormhole attack

behaviors, which could make protocols inherently resilient to such attacks. Our goal is not to detect and to eliminate attacks, but rather to enhance the routing protocols resiliency in order to limit damages.

IV. Protocol behaviors enhancing resiliency

Deterministic protocol behavior forces traffic to flow on a subset of "best" routes, in the quest of optimization (see the discussion in Section II-C). As a result of this, packet delivery success and failure are not fairly distributed among the network nodes; some nodes will have a good delivery ratio and others very bad ones. This is a limitation of the protocol since the network structural (i.e., network topology) redundancy is not exploited to benefit from physically existing alternative routes. In this Section, the techniques that can be employed in order to circumvent this limitation are described.

In this respect resiliency will permit: first, to avoid complete disconnection of nodes; second, graceful degradation of the delivery ratio as the number of compromised nodes increases; and third, obtain packet delivery ratios higher than those achieved by the standard protocols under the same conditions.

The complexity (overhead) of our proposal compared to the deterministic protocol is provided in terms of energy consumption.

Our goal then is to make resiliency emerge through modified protocol behavior. To this end, inspired by previous work, we believe that techniques enabling both dynamic and redundant behavior at the protocol level are needed.

A. *Random selection of the next hop*

A dynamic (random) behavior can be introduced in different ways according to the routing protocol features. In protocols that require a route discovery process, such as DSR, multiple routes can be discovered once and for each `DATA` packet the source node can each time select randomly a different route among the discovered ones. In a protocol without route discovery, such as GF, each node can determine a subset of direct neighbors that are closest to the sink compared to itself and choose the next hop randomly in this subset. Depending on how "greedily" a `DATA` packet should be forwarded, several neighborhood subsets can be constructed. For instance, in a GBR, each node can randomly choose a next hop among those who have a "height" strictly less than itself.

Generally speaking implementing this behavior requires two things. First, the set of selection candidates needs to be defined; it can be of arbitrary size constrained by some maximum allowed distance from the sink. Second, a selection probability law on this set needs to be specified; for instance, it may be desirable that the network node chooses neighbors closer to the sink with higher probabilities. The network node has thus the opportunity to make a random choice for the next hop with a probability to choose the nodes more or less close to the sink.

With this method, the structural redundancy of a physical topology can be effectively exploited in making the protocol fairer in terms of packet loss per node and thus more resilient since the overall packet delivery success can be attributed to a larger population of nodes. Furthermore the energy dissipation at the network is also fairer since the most solicited nodes under a deterministic scheme, i.e., those along the shortest routes, are relieved. Yet another advantage is that attacks targeting state information become less effective since now a single compromised node is not enough to compromise an entire neighborhood. However, this method may decrease packet delivery ratio and increase power consumption due to the lengthening of routes to the sink. There is thus a resiliency-power trade-off that needs to be evaluated. It is possible that by varying the parameters, of candidate set size and selection probability law, this trade-off can be controlled and kept to acceptable levels.

B. *Traffic redundancy*

Another means to effectively exploit the structural redundancy of the network is to enforce some degree of replication of sent packets. Each replica should then follow its own path to reach the sink.

Here two packet replication schemes to achieve redundancy are considered:

- Nodes replicate their own packets a number of times and send them to an equal number of appropriately selected neighbors. The forwarding nodes do not replicate packets and discard duplicates.
- Packets are replicated both at the source and at each intermediate node along the route. Intermediate nodes discard duplicates of already forwarded packets.

By construction deterministic protocols such as DSR, GF, GBR, cannot take advantage of redundant sends to increase their delivery ratio. If at least one node is compromised along the route, all redundant packets are lost, as they take always the same route. Such protocols need to be modified to be able to construct alternative shortest routes for each replica but even then their static nature does not allow them to be resilient. In this respect the discovery, construction and maintenance of alternative routes becomes an important consideration. In the literature, most of the multi-path routing protocols use multiple node (or link) disjoint paths to send redundant packets as shown in [27], [28] for example. A packet delivery rate can be increased significantly by using node disjoint multi-path routing. However, as the protocol gets more complex the energy required to discover and to maintain such multiple node disjoint paths is high.

C. *Probabilistic routing with traffic redundancy*

Finally, we can mix all presented strategies to obtain a random probabilistic routing with traffic redundancy. In this case, the structural redundancy of a physical topology is effectively exploited with some probability to choose longer routes. It will be shown that the random choice of a next hop candidate combined to packet replication naturally implements efficient enough route diversity even though for protocol simplicity node disjoint multiple paths are not guaranteed.

V. Simulations and Results

As a first attempt to better understand routing resiliency as well as the associated cost in terms of power consumption, these techniques were applied on the conventional GBR protocol to analyze through extensive simulation if and how its resiliency to attacks is improved. Simulations were performed using WSNet [29], an event-driven simulator for wireless networks.

A. *Simulation environment*

In our simulations, a unique sink is assumed at the center of the field. The deployed nodes have fixed positions during each simulation. The simulations are averaged over 100 trials for each case with a 95% confidence interval. Table I sums up the simulation parameters.

At this stage we configure WSNet for ideal MAC/PHY layers (e.g., no interference, no path-loss and no collisions) in order to isolate the impact of the defined attacks on routing and conceptually validate our approach before engaging into more resource consuming simulations and pilot deployments, which ultimately will be necessary.

B. *Protocol under study*

GBR [10] is a flooding based routing protocol, which is suitable for routing `DATA` packets from all source nodes to a sink. GBR uses two types of packets: `INTEREST` and `DATA` packets. The sink floods an `INTEREST` packet in order to setup a gradient. The `INTEREST` packet records the number of hops taken from the sink. Then a node can discover its minimum number of hops from the sink, called the node "height". The height difference between a node and each of its neighbors is the gradient on that link. The gradient setup process is executed only once at the beginning of the simulation. The following variants of GBR are considered:

Table I
Summary of the simulation parameters

Parameter	Value
Number of nodes	300
Area size	$100 \times 100m$
Transmission range	$20m$
Topology	uniformly distributed
Traffic generation	Poisson distribution $\lambda = 1\ p/s$
Simulation time	$100s$
Number of packets	30000
Number of runs	100

Table II
Summary of notations

Notation	Description
s	a network node
h_s	height of s
$U(s) = \{u_1, u_2, ..., u_{n_s}\}$	neighbors of s
$V(s) = \{v_1, v_2, ..., v_{m_s}\}$	neighbors of s with height $< h_s$
$W(s) = \{w_1, w_2, ..., w_{l_s}\}$	neighbors of s with height $= h_s$

1) Deterministic GBR: A given network node s sends `DATA` packets to a forwarding candidate with the minimum "height" in order to make maximum progress toward the sink. The next hop candidate, v_i, is chosen in $V(s)$, $1 \leqslant i \leqslant m_s$ (Table II). If several neighbors have the same "height", we choose the first one registered.

2) Random GBR: A given network node s sends `DATA` packets to a randomly chosen forwarding candidate with strictly lower "height" than itself. The next hop candidate, v_i, is chosen randomly in $V(s)$, $1 \leqslant i \leqslant m_s$ (Table II). The nondeterminism introduced by a random selection of the next hop is conceptually similar to SIGF [24]. However, we used a gradient value instead of a geographic distance.

3) Random probabilistic GBR: We have considered two cases according to the probability to select the next hop candidate. Let p_t and $\tilde{p_t}$ be real numbers such that $p_t + \tilde{p_t} = 1$. The considered cases are $p_t = \{0.8, 0.6\}$ and $\tilde{p_t} = \{0.2, 0.4\}$. For a network node s, p_t is the probability to choose the next hop candidate $v_i \in V(s)$, $1 \leqslant i \leqslant m_s$ (Table II) in the subset of neighbors closer to the sink and $\tilde{p_t}$ is the probability to choose the next hop candidate $w_j \in W(s)$, $1 \leqslant j \leqslant l_s$ (Table II) in the subset of neighbors with the same height as itself.

4) Random probabilistic GBR with redundancy: Two cases of redundancy are considered; `DATA` packets are replicated twice (i) at the source node and (ii) by each node along a full path. In those two cases, duplicate copies of a packet are dropped by forwarding nodes.

C. *Implemented attacks*

In the following, we assume a unique trustworthy sink. Sensor nodes are assumed untrustworthy since they are

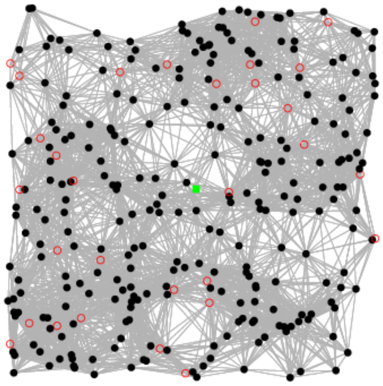


Figure 5. Compromised nodes randomly placed (uniformly)

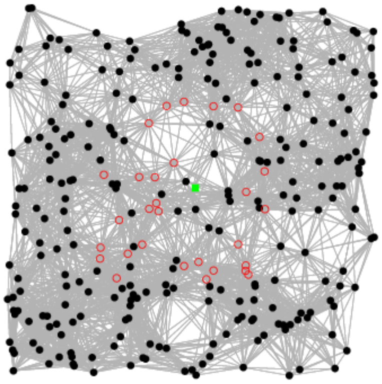


Figure 6. Compromised nodes concentrated around the sink

vulnerable to physical attacks and can be compromised.

With respect to definition described in Section III-B, malicious nodes may belong to one of the following adversarial categories: "mote-class", "active", and, "insider" attackers.

We implemented Selective forwarding as a basic attack and further we considered combining this basic attack with Sybil, Wormhole and Sinkhole attacks.

1) Basic attack: Selective Forwarding. Assuming that the adversary has no information about the location of the sink, the k compromised nodes are randomly and uniformly distributed on a $N \times N$ square field (Fig. 5). For simulations k varies between 10% and 50% of the node population. Malicious nodes do not disturb gradient setup phase and retransmit `INTEREST` packets with correct hop count. They drop all `DATA` packets coming from their neighbors, however, they generate and send their own `DATA` packets to the sink.

2) Combined attack #1: Sinkhole with Selective forwarding. Assuming that the adversary has some information about the location of the sink, the k compromised nodes are randomly distributed on a $M \times M$ (e.g., $M = N/2$) square field around the sink (Fig. 6). For simulations k varies between 10% and 30% of the node population. Malicious nodes simply drop all `DATA` packets coming from their neighbors. However it is assumed that malicious nodes do not disturb the gradient setup phase, retransmit the `INTEREST` packets used for gradient setup with with a correct hop count and finally, they normally generate and send their own `DATA` packets to the sink.

3) Combined attack #2: Sybil with Selective forwarding. The k compromised nodes are randomly and uniformly distributed on a $N \times N$ square field (Fig. 5). For simulations k varies between 10% and 50% of the node population. According to Sybil attack taxonomy [8], our model corresponds to "direct communication" where Sybil nodes communicate directly with legitimate nodes, using "fabricated identities" where an attacker can simply create arbitrary new Sybil identities (not existing in the network) and it is of the "simultaneous" form where an attacker may participate all of his Sybil identities simultaneously in the network. In this adversary model, malicious nodes take two identities. A compromised node disturbs gradient setup phase by duplicating `INTEREST` packets. A malicious node puts a false identity to the duplicated `INTEREST` packet to make believe to their neighbors that there are two nodes, while physically there is only one node. The probability to be chosen for the next hop increases for a malicious node and it can attract more traffic. A malicious node does not lie about its gradient and the two identities take the same true gradient. We choose this particular strategy to separate the impact of Sinkhole attack (which will be the case if the Sybil node lies on its gradient) and of the Sybil attack itself. The false identity is chosen randomly in the large interval of non existing identities to avoid collisions. Once two identities are created, a malicious node drops all `DATA` packets coming from its neighbors for both its own and Sybil identities. We also assume only one Sybil identity to be convinced that a Sybil node will not be detected by simple mechanisms such as node degree comparison even if this strategy limits the impact of Sybil attack.

4) Combined attack #3: Wormhole with Selective forwarding. Two colluding malicious nodes can make believe that they are neighbors even if they are physically distant by tunneling messages via an out-of-band connection. Every pair of malicious nodes $(w1; w2)$ with a distance greater than two hops, creates a Wormhole link. An `INTEREST` packet received by $w1$ is directly transmitted to w2 by using the out of band connection. Thus, tunneled `INTEREST` packets arrive sooner than other packets transmitted over a normal multi-hop route. If $w1$ is placed near the sink, $w2$ obtain a gradient lesser than its neighbors and $w2$ can attract its neighbors' traffic. The k malicious nodes are randomly distributed across the whole network, except in the border. The total number of Wormhole links is $k/2$. For simulations k varies between 10% and 50% of the node population. Once a Wormhole link is created between two malicious nodes $(w1; w2)$, they will drop all DATA packets coming from their neighbors. A given malicious node only belongs to one Wormhole link, the case of several Wormhole links coming from a single Wormhole node is not treated here.The Wormhole malicious nodes use legitimate traffic to perform their activity: falsify neighborhood information and attract traffic; collect node identities and use them as Sybil ones instead of having to fabricate false ones.

D. Evaluation metrics

To gain insight concerning the WSN routing resiliency some metrics are needed in order to meaningfully summarize the information collected by simulations. A single such metric is currently lacking and is an object of ongoing research. As a provisional substitute we have used the following metrics:

- **Average delivery ratio (ADR):**
$$ADR = N_r/N_s, \tag{1}$$
where N_r, N_s are respectively the total number of received and sent packets.

ADR is an important metric to evaluate the overall success of routing functionality, i.e., packet delivery. To refine over the information provided by ADR, we also measured the delivery ratio per node and we grouped the measurements into 5 classes.

- **ADR classes:**
  - Class $c1$: nodes with $ADR = 100\%$
all DATA packets from these nodes are received by the sink
  - Class $c2$: nodes with $ADR \in [66\%; 100\%[$
  - Class $c3$: nodes with $ADR \in [33\%; 66\%[$
  - Class $c4$: nodes with $ADR \in]0\%; 33\%[$
  - Class $c5$: nodes with $ADR = 0\%$
no DATA packet from these nodes is received by the sink and so they are totally disconnected from the sink

This measure allows to determine the distribution of transmission success in the node population and the fracture of the network connectivity. In our point of view, the higher the number of connected source nodes (even if with a low ADR), the more the routing protocol is resilient.

- **ADR per distance:** The delivery ratio per node is measured and grouped according to the distance (in number of hops) of nodes from the sink.

To get the distance in number of hops, we take the geographical distance between the source nodes and the sink, and we divide it by the transmission range. All source nodes have the same transmission range. The routing protocols are more resilient if more distant nodes are able to still reach the sink and thus successfully transmit packets.

- **Average path length (APL):** The number of hops crossed by each received packet.
The end-to-end delay is not explicitly measured in this paper since for our simulations we configure WSNet for ideal MAC/PHY layers which implies no retransmission and no propagation delays. However, the average path length (i.e., hop count) is directly proportional to the average end-to-end delay of the network (see Fig. 11a and Fig. 10) and in this sense it provides an indication of.
- **Normalized power consumption (NPC):**
$$NPC = T_e/\tilde{T_e} \tag{2}$$
where the total energy consumption (T_e) is normalized by the energy consumption of the deterministic GBR without attack and without packet replication sent ($\tilde{T_e}$).

NPC allows to objectively compare energy expenditure under attacks for each case (including redundancy) without having to enter at this time into low level considerations requiring power consumption modeling. The energy model of WSNet as detailed in the WSNet documentation (see in [30]) is linear: the sleep and idle modes of the MAC layer are not taken into account whereas the basic model considers that the cost for one bit sent is 1 and the cost for one bit received is 2. The total energy is thus computed taking into account the energy cost of each bit received or sent.

E. Results and analysis

The focus of our simulations is on comparing the four versions of GBR (Deterministic, Random, Random probabilistic $p_t = 0.8$ and Random probabilistic $p_t = 0.6$) with a single and two types of redundant DATA packets under four implemented attacks discussed in Section V-C, in term of metrics discussed in Section V-D.

An example of the functional flow diagram with traffic redundancy (double sent full path) under a basic Selective forwarding attacks is presented in Fig. 7.

1) Results for the basic Selective Forwarding attack: As expected the average delivery ratio (Fig. 8), the average path length (Fig. 11) and the total energy consumption (Fig. 12) decrease with increasing number of compromised nodes under the basic Selective forwarding attack. When a single DATA packet is considered, Deterministic and Random GBR have a higher delivery ratio than others (Fig. 8(a)). The path length is inversely proportional to the average delivery ratio. With probability p_t decreasing, the average path length (Fig. 11(a)) and the total energy consumption (Fig. 12(a)) increase. However, as the number of the next hop candidates is increased, the structural redundancy of the network is better exploited.

As shown in Fig. 9 in Deterministic GBR only two classes appear. For any source node s either all DATA packets will be successfully delivered ($ADR_s = 100\%$), i.e., no malicious node is along the route, or all DATA packets will be lost ($ADR_s = 0\%$), i.e., at least one forwarding node is compromised along the route. In last case, a source

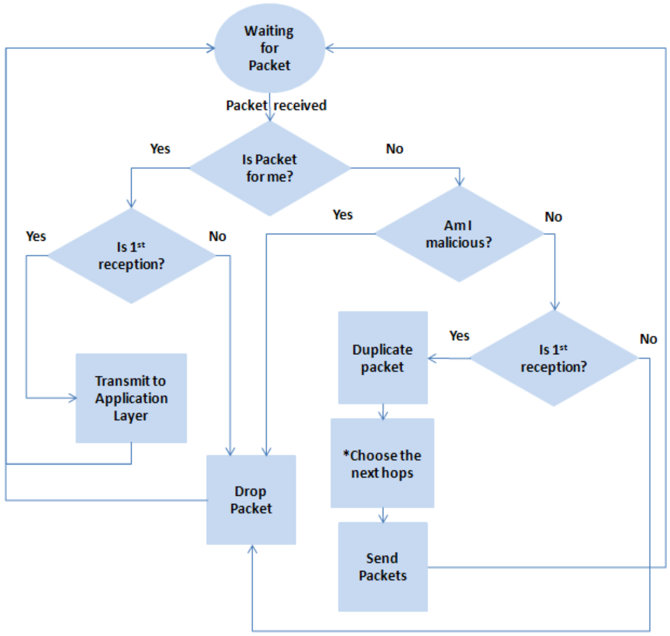
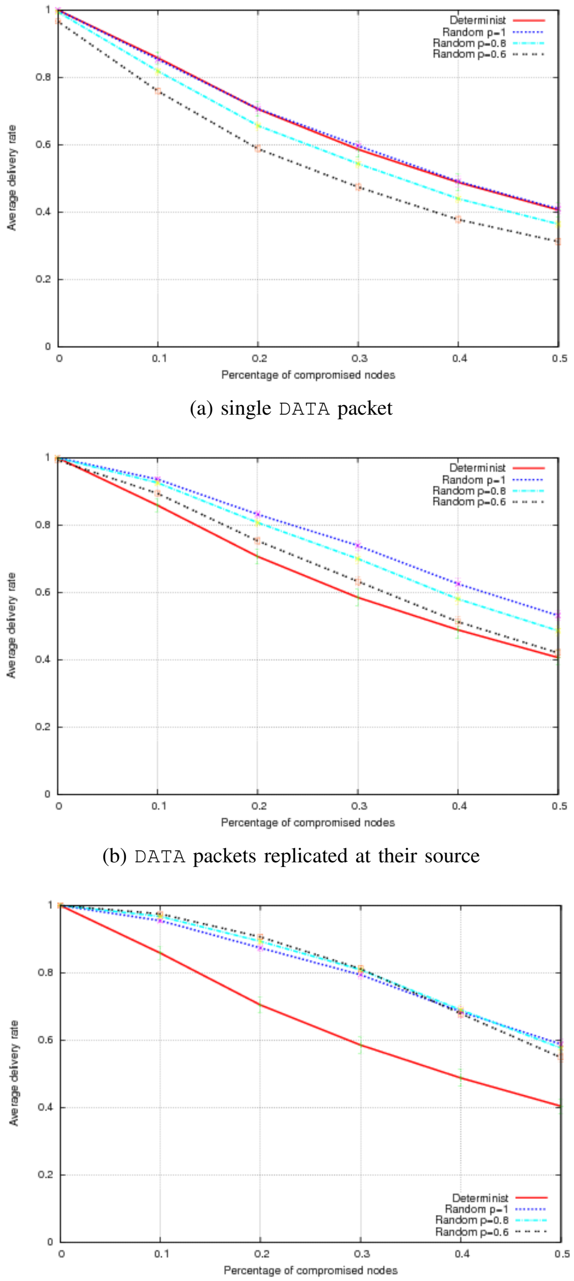


Figure 7. Example of the functional flow diagram with traffic redundancy (double sent full path) under a basic Selective forwarding attack. * The choice of the next hop depends on the dedicated routing protocol as described in Section V-B1 for Deterministic GBR, in Section V-B2 for Random GBR with $p = 1$, in Section V-B3 for Random GBR with $p = 0.8$ and in Section V-B4 for Random GBR with $p = 0.6$.



(a) single DATA packet

(b) DATA packets replicated at their source

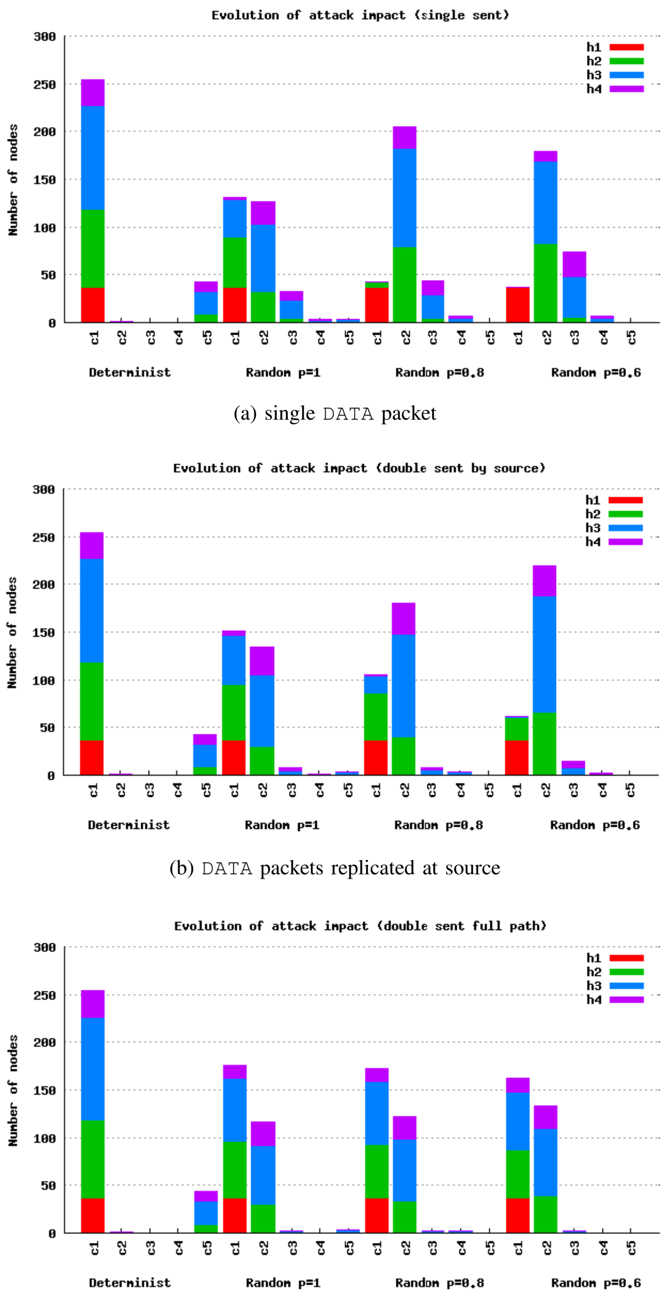
(c) DATA packets replicated by all forw. nodes along the route

Figure 8. Basic Selective forwarding - Average delivery ratio (ADR)

node s is completely disconnected from the sink. Note also that the number of disconnected nodes ($c5$) is significantly important (15%) for Deterministic GBR. On the contrary with all variants of Random GBR four classes $c1$ to $c4$ appear. With Random GBR a low number of nodes are completely disconnected from the sink ($c5$). Note that since the network saves energy due to dropped packets by the compromised nodes, this energy gain can then be exploited by redundant DATA packets to further improve resiliency and ADR. In this way, the source nodes can reach the sink as long as possible, thus, enhancing the network connectivity (Fig. 9).

Resiliency and ADR over Deterministic GBR further improve when probabilistic behaviors are mixed with DATA packet replication at the source because DATA packets may take potentially different routes thanks to the random selection of next-hop neighbors. As shown in Fig. 8(b), all random versions exhibit higher delivery ratio performance, though their average path length is higher (Fig. 11(b)), than the Deterministic GBR whose performance remains unchanged. With traffic redundancy, in Fig. 9(b) and (c), we can observe that the number of nodes with higher delivery ratio ($c1$ and $c2$) is increased and the number of disconnected nodes from the sink ($c5$) is decreased for all Random GBR protocols, while for Deterministic GBR the situation remains unchanged. Network reliability is thus improved since most source nodes remain connected.

As expected (Fig. 9 (a)) with decreasing probability p_t, ADR decreases when the distance from the sink (in number of hops) increases due to the route length effect. However, with traffic redundancy, the ADR of distant nodes is increased for all random versions, while for Deterministic GBR it remains unchanged (Fig. 9 (b) and (c)). Resiliency is thus improved since distant nodes have better delivery ratio. Nevertheless this has a price, as shown in Fig. 12(b)



(a) single DATA packet

(b) DATA packets replicated at source

(c) DATA packets replicated by all forw. nodes along the route

Figure 9. Basic Selective forwarding - ADR classes (c1 to c5) with $k = 10\%$ of compromised nodes; distribution of distances from the sink in number of hops (h1 to h4) within each class is shown

all random versions have a higher energy consumption than Deterministic GBR.

In the last case, where DATA packets are replicated at each intermediate node along a full path, a significant improvement on delivery ratio is observed (Fig. 8(c)). Sending redundant DATA packets by each intermediate node on a full path mixed with a random behavior significantly enhances the resiliency. It appears that for uniformly distributed compromised nodes variation of the probability p_t does not influence the delivery ratio. So, we may choose the value of p_t that has lower energy consumption. In this respect Random GBR ($p_t = 1$) remains the better trade-off in term of energy-resiliency (Fig. 12(c)). However, it remains to be confirmed if for more realistic spatially distributed compromised nodes, the lower probability p_t may allow better delivery rates as it increases the number of next hop candidates.

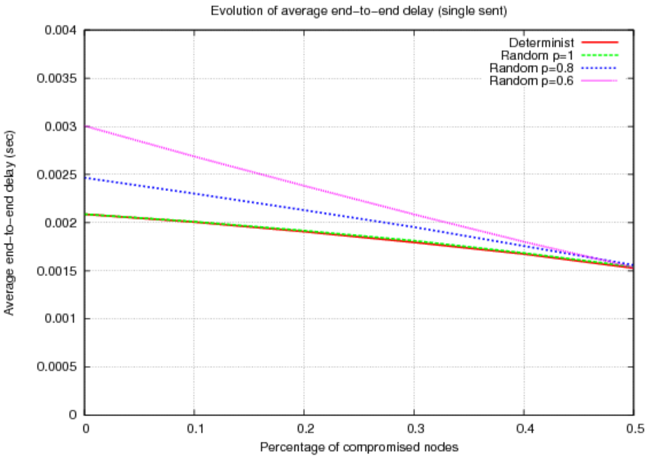
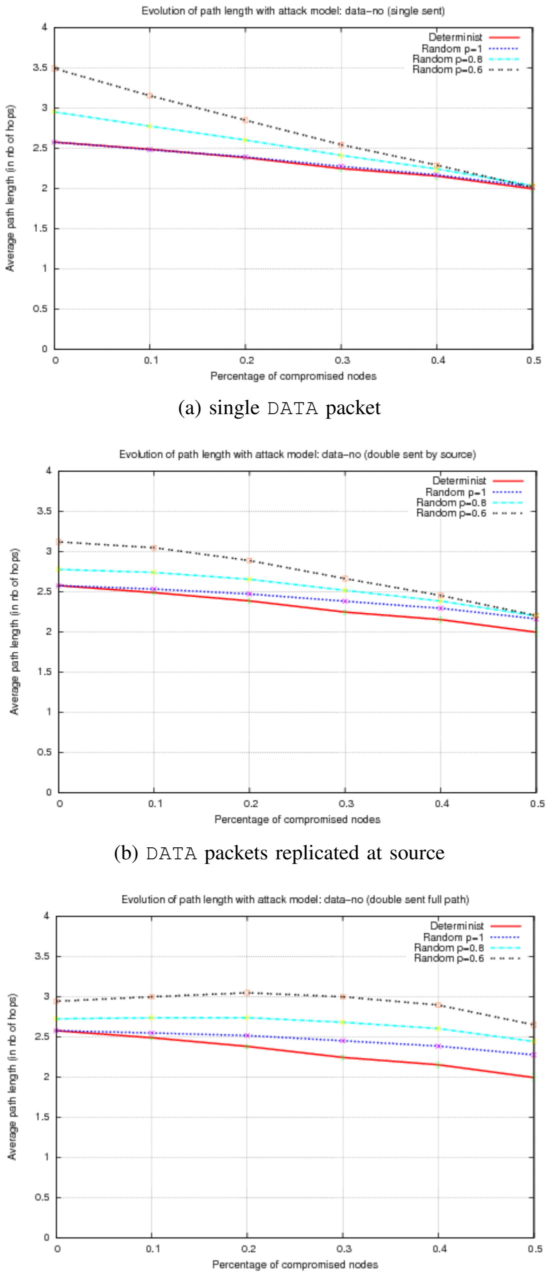


Figure 10. Basic Selective forwarding (single DATA packet) - Average end-to-end delay (sec)

2) Results for the combined attacks: In this Section we illustrate results of four versions of GBR with combined attacks; Sybil, Wormhole and Sinkhole with traffic redundancy, where DATA packets are replicated at each intermediate node along a full path.

Sybil attack results: In Fig. 13(a), we observe that the impact of combined Sybil attack is more important than with basic Selective forwarding. When malicious nodes create two identities, they increase the probability to be chosen as the next hop by their neighbors, if they have smallest gradient. Once chosen as the next hop, they receive more packets for retransmission. With traffic redundancy all Random GBR variants have better delivery ratio than Deterministic GBR. The number of nodes in classes $c1$ and $c2$ is higher than in other classes for all Random GBR variants (Fig. 14(a)). As a result, with all Random GBRs, most of source nodes have ADR greater than 66% with 10% of compromised nodes and very few nodes are disconnected ($c5$). In Deterministic GBR, 20% of source nodes are disconnected from the sink with 10% of compromised nodes, while with Random GBR



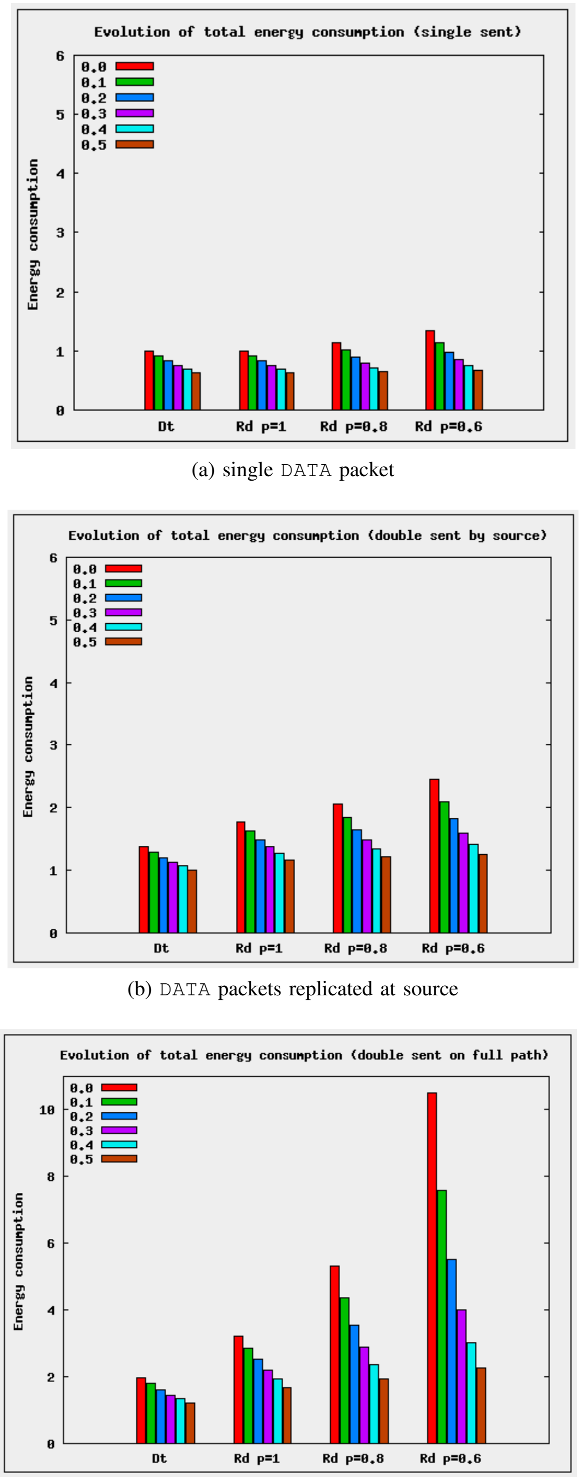
(a) single DATA packet

(b) DATA packets replicated at source

(c) DATA packets replicated by all forw. nodes along the route

Figure 11. Basic Selective forwarding - Average path length (APL)

$(p_t = 1)$ only $0,01\%$ are completely disconnected. Network reliability and resiliency are improved again with Random GBR, since most of the source nodes remain connected. The ADR of distant nodes is increased for all random versions, whereas for Deterministic GBR ADR remains unchanged (Fig. 14 (a)). Resiliency is improved with Random GBR under combined Sybil attack, since distant nodes have better delivery ratio. However, the energy consumption with traffic redundancy (Fig. 16(a)) is increased about 3 times.



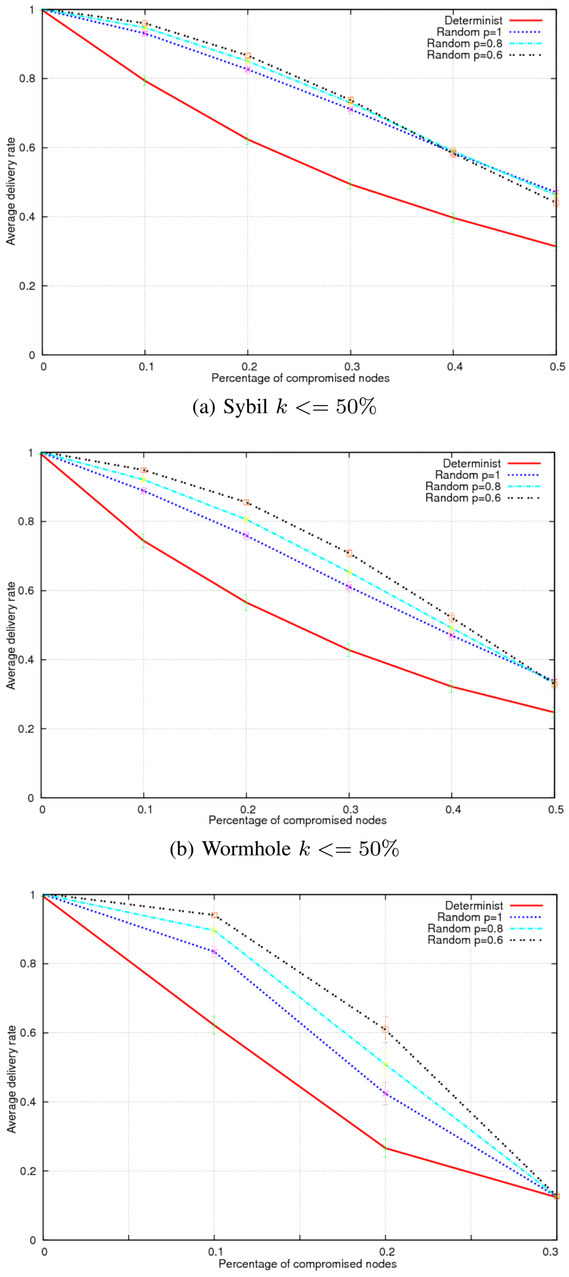
(a) single DATA packet

(b) DATA packets replicated at source

(c) DATA packets replicated by all forwarding nodes along the route

Figure 12. Basic Selective forwarding - Norm. Power Consumption (NPC)

Wormhole attack results: Fig. 13(b) shows that the impact of combined Wormhole attack is more important than both basic Selective forwarding and combined Sybil attacks. If we consider a pair $(w1; w2)$ of malicious nodes

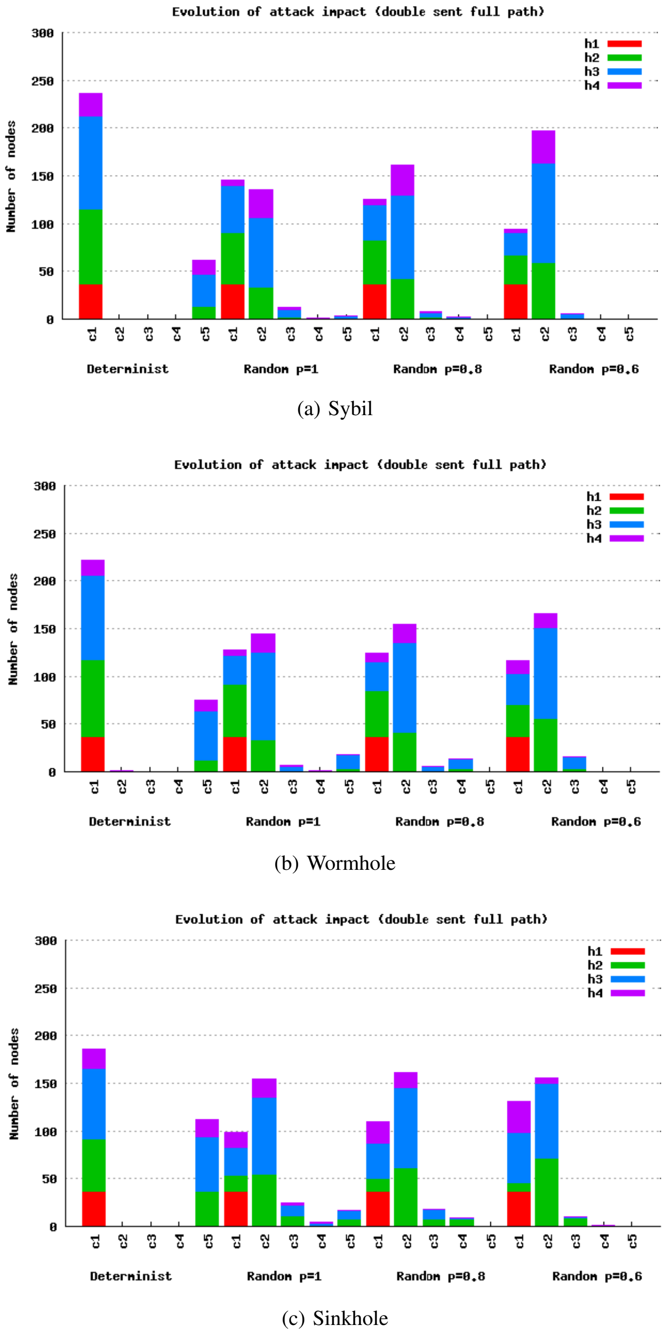


(a) Sybil $k <= 50\%$

(b) Wormhole $k <= 50\%$

(c) Sinkhole $k <= 30\%$

Figure 13. Combined attacks with DATA packets replicated by all forwarding nodes along the route - Average delivery ratio (ADR)



(a) Sybil

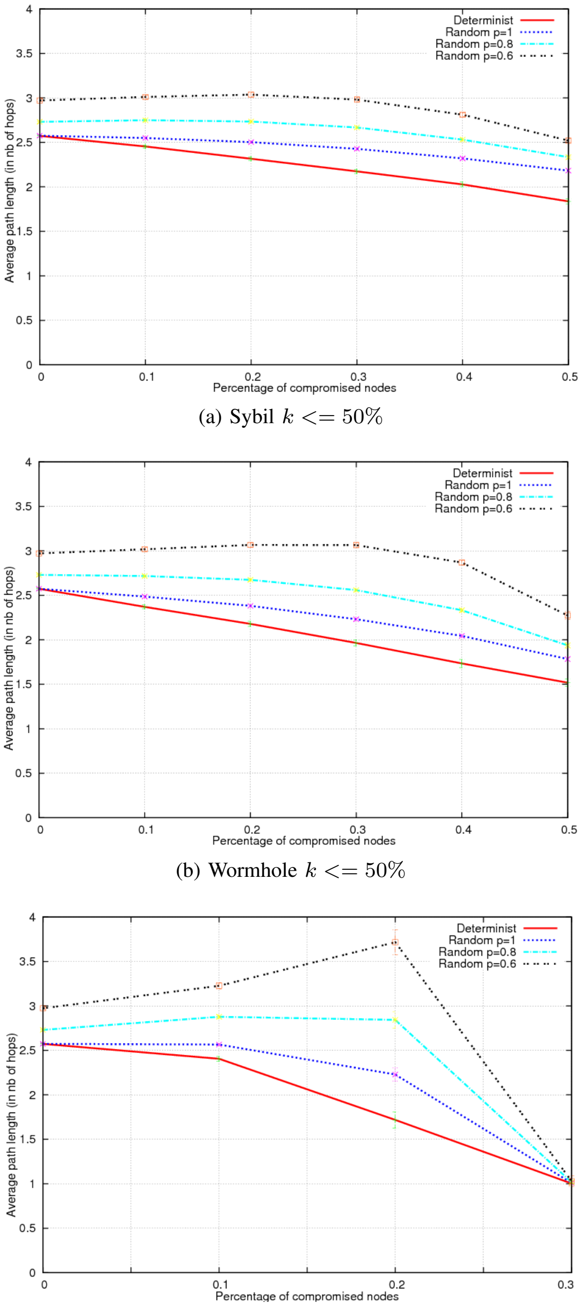
(b) Wormhole

(c) Sinkhole

Figure 14. Combined attacks with DATA packets replicated by all forwarding nodes along the route - ADR classes (c1 to c5) with $k = 10\%$ of compromised nodes; distribution of distances from the sink in number of hops (h1 to h4) within each class is shown

and if $w1$ is placed near the sink, $w2$ obtains a gradient lesser than its neighbors and the Wormhole can attract the traffic. Here again, all Random GBR protocols have better delivery ratio than Deterministic GBR. In Deterministic GBR, 25% of source nodes are disconnected from the sink with 10% of compromised nodes and with Random GBR ($p_t = 1$) it is $0,06\%$ (Fig. 14(b)). Network reliability and resiliency are also improved with all Random GBR variants, since the majority of source nodes remain connected (Fig. 14(b)) and the ADR of distant nodes is increased (Fig. 14(b)). Resiliency is improved with Random GBR under combined Wormhole attack and the energy consumption (Fig. 16(b)) due to traffic redundancy remains almost the same as combined Sybil attack.

Sinkhole attack results: In Fig. 13(c), we observe that the impact of combined Sinkhole attack is the most important



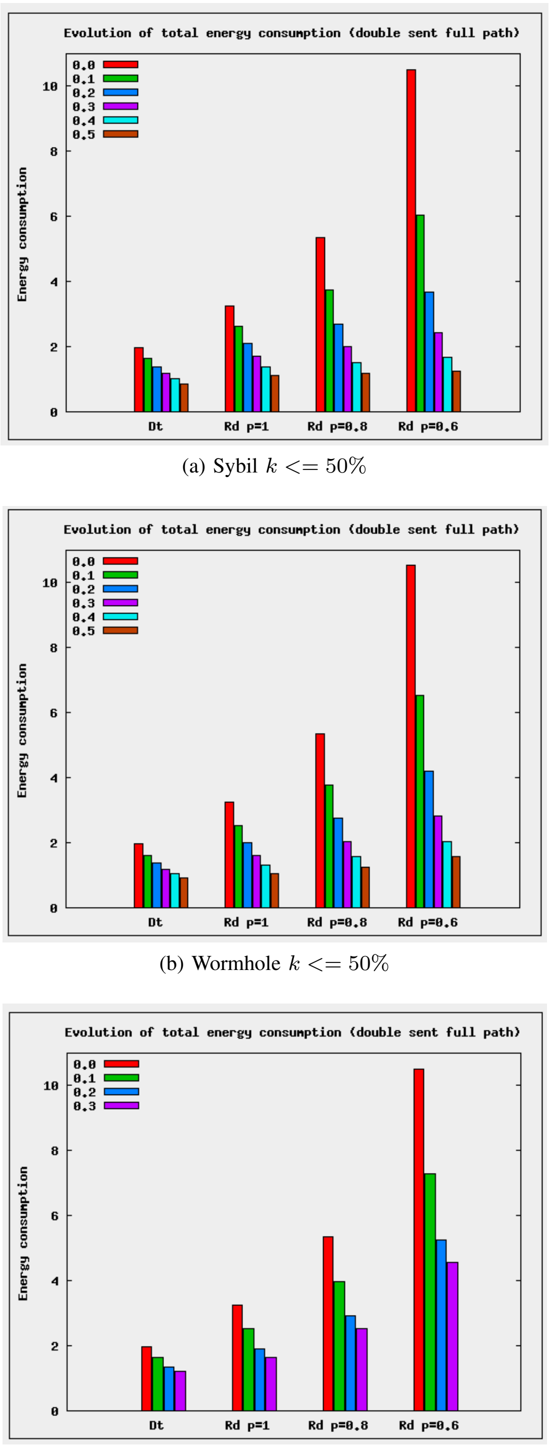
(a) Sybil $k <= 50\%$

(b) Wormhole $k <= 50\%$

(c) Sinkhole $k <= 30\%$

Figure 15. Combined attacks with `DATA` packets replicated by all forwarding nodes along the route - Average path length (APL)

compared to all other attacks. When the compromised nodes are close to the sink, they receive for retransmission more packets than other nodes: they naturally attract most of the traffic. It is worth noting the significant differences in terms of delivery ratio for all random versions compared to Deterministic GBR as well as among the different versions of Random GBR with traffic redundancy. As packets can take longer routes with Random GBR $p_t = 0.6$ (Fig. 15(c)), messages can find "unaffected" routes around the sink if exist. Hence, distant nodes have more chance to find those "healthy" routes near the sink. The source nodes close to the sink have lower ADR because of the important number



(a) Sybil $k <= 50\%$

(b) Wormhole $k <= 50\%$

(c) Sinkhole $k <= 30\%$

Figure 16. Combined attacks with `DATA` packets replicated by all forwarding nodes along the route - Normalized Power Consumption (NPC)

of compromised nodes in their neighborhood (Fig. 14(c)). When all nodes around the sink are compromised, the sink receives packets only from these malicious nodes and no DATA packets are received from the legitimate nodes. That is why we observe on Fig. 15(c) a path length that tends to 1. Resiliency is improved with a Random GBR under the combined Sinkhole attack and the energy consumption (Fig. 16(b)) due to traffic redundancy remains almost the same as with other attacks.

VI. Conclusion

In this article, we have considered the case of mote-class/active/insider attacks against WSN multi-hop routing protocols. In this specific context of node compromise cryptography needs to be complemented by algorithmic approaches. We have proposed WSN routing strategies enhancing the protocol resiliency in the presence of maliciously packet-dropping compromised nodes. The basic Selective forwarding attack as well as its combination with Sinkhole, Sybil and Wormhole attacks was thoroughly investigated in the context of the well established GBR.

We have started by analyzing the conditions required for resiliency at the routing layer. The two main findings were that, first, the shortest-path optimization principles though good for energy efficiency are not adapted at all from the routing layer security (i.e., resiliency to insider attacks) standpoint and, second, that the structural redundancy in the network topology should be effectively exploited by employing some form of redundant protocol behavior.

In accordance with these findings our proposal consists in combining random next-hop selection and packet replication; both are needed. A random and probabilistic choice of the next hop candidates allows a dynamic behavior in route selection exploiting thus the structural redundancy of the network. However, the packet delivery ratio may suffer since packets may take longer routes.

With increasing path length (in terms of packet hop count), the overall delay across the network increases as well. The overall delay is directly proportional to the average path length (ideal MAC/PHY layers). However, we observed that under worst attack scenario such as Sinkhole attacks, the average path length of successfully delivered packets tends to one. This can be explained by the fact that with increasing number of compromised nodes, the sink ends up receiving packets only from its direct neighbors. Similarly, in the worst case mass attack scenario (a large number of both insider and outsider attackers), the observed overall delay across the network will also decrease since most of the packets from distant nodes will be lost.

To counterbalance the longer route effect such dynamic (probabilistic) behavior needs to be combined with some form of packet replication. To validate our ideas we have extensively simulated the proposed techniques by modifying in various ways the well-known routing protocol GBR. The results show that the resiliency of routing protocols can be effectively enhanced.

The main merits of our proposal compared to the classical deterministic protocols are:

- the delivery ratio is improved; “graceful” degradation of the delivery ratio with increasing number of compromised nodes.
- the delivery success is fairly distributed; more sources transmit with a high delivery ratio and distant nodes have better delivery success.
- the connectivity is improved; more sources are remain connected to the sink with increasing number of compromised nodes.
- the structural redundancy of the physical topology is better exploited and the energy consumption is fairly distributed; more nodes participate to the routing.

From simulations, we found that traffic redundancy is extremely energy consuming when no attack, but energy efficiency of the protocol is improved when under attack. Hinging on this observation a future work perspective is the search of a mechanism to dynamically adapt the degree of dynamic/redundant behavior to equalize energy cost and so keep the energy consumption-resiliency trade-off at acceptable levels. It also seems that keeping the routes short (in terms of hop count) should be sought but there are some particular cases (e.g., combined Sinkhole attack) where longer routes should be permitted in order to get around obstacles. It is worth mentioning that in our simulation study we have gone beyond the simple Selective forwarding attack to consider combined attacks (such as Wormhole and Selective forwarding) concluding that these attacks have extreme impact on routing especially when Sinkhole and selected forwarding are combined together.

From our simulation analysis we conclude that an operational definition resiliency, in the context of network routing, should incorporate the notions of fairness, preservation of connectivity and graceful degradation of delivery ratio. Thus, our ongoing research especially concerns the definition of a metric of resiliency that includes all those notions. Such a metric will be a valuable tool in analyzing protocol resilience and will greatly simplify the process of protocol comparison.

Finally, in a near future, we also need to relax the ideal MAC/PHY assumption to validate the performance of resilient routing techniques when channel imperfections and medium access limitations are taken into account; to this end it would be interesting to consider modeling packet loss due to MAC/PHY limitations as a form of unintentional Selective forwarding.

Acknowledgment

Ochirkhand Erdene-Ochir is partially supported by a PhD graduate scholarship through a CIFRE contract between France Telecom and the French Ministry of Industry.

References

[1] P. Papadimitratos and Z. Haas, "Secure rotuing for mobile ad hoc networks," in *Communication Networks and Distributed Systems Modeling and Simulation Conference*, San Antonio, Texas, 2002, pp. 27–31.

[2] Y. C. Hu, A. Perrig, and D. B. Johnson, "Ariadne: a secure on-demand routing protocol for ad hoc networks," *Wireless Networks*, vol. 11, no. 1-2, pp. 21–38, January 2005.

[3] K. Sanzgiri, B.Dahill, B. N. Levine, C. Shields, and E. M. Belding-Royer, "A secure routing protocol for ad hoc networks," in *IEEE International Conference on Network Protocols*. Paris, France: IEEE Computer Society, November 2002, pp. 78–89.

[4] A. Perrig, R. Szewczyk, V. Wen, D. E. Culler, and J. D. Tygar, "Spins: security protocols for sensor netowrks," in *Seventh Annual International Conference on Mobile Computing and Networks*, Rome, Italy, July 2001, pp. 189–199.

[5] O.Erdene-Ochir, M.Minier, F. Valois, and A. Kountouris, "Toward resilient routing in wireless sensor networks: Gradient-based routing in focus," in *4th International Conference on Sensor Technologies and Applications (Sensorcomm)*, Venice, Italy, July 2010.

[6] B. Xiao, B. Yu, and C. Gao, "Chemas: Identify suspect nodes in selective forwarding attacks," *Journal of Parallel Distributed Computing*, vol. 67, no. 11, pp. 1218–1230, June 2007.

[7] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," in *6th annual international conference on Mobile computing and networking*, Boston, USA, August 2000, pp. 255–265.

[8] J. Newsome, E. Shi, D. Song, and A. Perrig, "The sybil attack in sensor networks: analysis & defenses," in *Information Processing in Sensor Networks*, K. Ramchandran, J. Sztipanovits, J. Hou, and T. Pappas, Eds. Berkeley, USA: ACM, April 2004, pp. 259–268.

[9] Y.-C. Hu, A. Perrig, and D. B. Johnson, "Packet leashes: A defense against wormhole attacks in wireless networks," in *22nd Annual Joint Conference of the IEEE Computer and Communications Societies*, San Fransisco, USA, April 2003, pp. 1976–1986.

[10] C. S. Mani and M. B. Srivastava, "Energy efficient routing in wireless sensor networks," in *Military Communications Conference Proceedings on Communications for Network-Centric Operations: Creating the Information Force*, vol. 1, McLean, USA, October 2001, pp. 357–361.

[11] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: attacks and countermeasures," *Ad Hoc Networks*, vol. 1, no. 2-3, pp. 293–315, August 2003.

[12] P. Michiardi and R. Molva, "Core: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks," in *Communications and Multimedia Security*, ser. IFIP Conference Proceedings, B. Jerman-Blazic and T. Klobucar, Eds., vol. 228. Portoroz, Slovenia: Kluwer, September 2002, pp. 107–121.

[13] B. Parno, A. Perrig, and V. Gligor, "Distributed detection of node replication attacks in sensor networks," in *IEEE Symposium on Security and Privacy*. Oakland, USA: IEEE Computer Society, May 2005, pp. 49–63.

[14] L. Hu and D. Evans, "Using directional antennas to prevent wormhole attacks," in *Network and Distributed System Security Symposium*. San Diego, USA: The Internet Society, February 2004, pp. 1–11.

[15] "http://www.merriam-webster.com/dictionary/resilience," July 2011.

[16] O.Erdene-Ochir, M.Minier, F.Valois, and A.Kountouris, "Resiliency of wireless sensor networks: Definitions and analyses," in *IEEE International Conference on Telecommunications (ICT)*, Doha, Qatar, April 2010.

[17] R. J. Ellison, R. C. Linger, T. Longstaff, and N. R. Mead, "Survivable network system analysis: A case study," *IEEE Software*, vol. 16, no. 4, pp. 70–77, July 1999.

[18] J. P. G. Sterbenz, R. Krishnan, R. Hain, A. Jackson, D. Levin, R. Ramanathan, and J. Zao, "Survivable mobile wireless networks: issues, challenges, and research directions," in *Workshop on Wireless Security*, W. Maughan and N. Vaidya, Eds. Atlanta, USA: ACM, September 2002, pp. 31–40.

[19] D. Wagner, "Resilient aggregation in sensor networks," in *ACM Workshop on Security of Ad Hoc and Sensor Networks*, S. Setia and V. Swarup, Eds. Washington, USA: ACM, October 2004, pp. 78–87.

[20] D. Ganesan, R. Govindan, S. Shenker, and D. Estrin, "Highly-resilient, energy-efficient multipath routing in wireless sensor networks," in *2nd ACM international symposium on Mobile ad hoc networking & computing*. Long Beach, USA: ACM, October 2001, pp. 251–254.

[21] X. Li and D. Yang, "A quantitative survivability evaluation model for wireless sensor networks," in *IEEE International Conference on Networking, Sensing and Control*, Japan, March 2006, pp. 727–732.

[22] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile Computing*, S. US, Ed., vol. 353, 1996, pp. 153–181.

[23] B. Karp and H. T. Kung, "Gpsr: greedy perimeter stateless routing for wireless networks," in *Proceedings of the 6th annual international conference on Mobile computing and networking*, Boston, USA, August 2000, pp. 243–254.

[24] A. D. Wood, L. Fang, J. A. Stankovic, and T. He, "Sigf: a family of configurable, secure routing protocols for wireless sensor networks," in *ACM Workshop on Security of ad hoc and Sensor Networks (SASN)*, ACM, Ed., VA, USA, October 2006, pp. 35–48.

[25] B. Blum, T. He, S. Son, and J. Stankovic, "Igf : A state-free robust communication protocol for wireless sensor networks," Technical report, Univ. of Virginia, Charlottesville, VA, USA, Tech. Rep. CS-2003-11, November 2003.

[26] E. D. L. Andersson and L. Zhang, “Report from the IAB workshop on Unwanted Traffic March 9-10, 2006,” RFC 4948 (Informational), August 2007. [Online]. Available: http://www.ietf.org/rfc/rfc4948.txt

[27] D. Ganesan, R. Govindan, S. Shenker, and D. Estrin, “Highly-resilient, energy-efficient multipath routing in wireless sensor networks,” *Mobile Computing and Communications Review*, vol. 5, no. 4, pp. 11–25, 2001.

[28] Y. M. Lu and V. W. S. Wong, “An energy-efficient multi-path routing protocol for wireless sensor networks,” *Int. J. Communication Systems*, vol. 20, no. 7, pp. 747–766, 2007.

[29] E. Hamida, G. Chelius, and J.-M. Gorce, “Scalable versus accurate physical layer modeling in wireless network simulations,” in *22nd Workshop on Principles of Advanced and Distributed Simulation*, Roma, Italy, June 2008, pp. 127–134.

[30] “http://wsnet.gforge.inria.fr/,” July 2011.

Reliability Estimation of Mobile Agent System in MANET with Dynamic Topological and Environmental Conditions

Chandreyee Chowdhury
Dept. of Computer Sc. and Engg.
Jadavpur University
Kolkata, India
email: chandreyee.chowdhury@gmail.com

Sarmistha Neogy
Dept. of Computer Sc. and Engg.
Jadavpur University
Kolkata, India
email: sarmisthaneogy@gmail.com

***Abstract*—A mobile agent is an agent with the ability to migrate from one host to another where it can resume its execution. Mobile agents can be used in wireless and mobile network applications in order to save bandwidth and time. In this paper we consider reliability issues that need to be addressed before mobile agents can be used in a broad range of applications in Mobile Adhoc Network. We show how a Mobile Agent based System can be made more reliable despite the uncertainties introduced by underlying network environment. Adhoc network brings in new aspects to dependability because the characteristics of such network affect reliability of the services offered by the agent system. Here we propose an algorithm for estimating the task route reliability of a system of agents that is based on the conditions of the underlying network. The system consists of independent agent groups, each group corresponds to a particular application for which these are deployed. The complexity of mobile agent based system combined with the underlying dynamic topology of adhoc network drives us to estimate it using Monte Carlo simulation. Smooth Random Mobility Model is used to estimate node location at a particular time. Environmental factors like multipath propagation that affect the received signal power are also considered. The results achieved demonstrate the robustness of the proposed algorithm. This paper demonstrates a reliability estimation model for mobile agent based system in mobile adhoc network and shows that reliability is heavily dependent on the conditions of the network and on agent heterogeneity.**

Keywords- Mobile Ad hoc network; Monte-Carlo; Reliability; Mobility Model; Fault-tolerance;

I. INTRODUCTION

A mobile agent is a combination of software program and data, which migrates from a site to another site to perform tasks assigned by a user according to a static or dynamic route [1]. It can be viewed as a distributed abstraction layer that provides the concepts and mechanisms for mobility and communication [2]. An agent consists of three components: the program, which implements it, the execution state of the program and the data. A mobile agent may migrate in two ways namely weak migration and strong migration [3]. Weak migration occurs when only the code of the agent migrates to its destination, a strong migration occurs when the mobile agent carries out its migrations between different hosts while conserving its data, state and code. The platform is the environment of execution. The platform makes it possible to create mobile agents; it offers the necessary elements required by them to perform their tasks such as execution, migration towards other platforms and so on.

Typical benefits of using mobile agents include

- Bandwidth conservation: sending a complex query to the database server for processing.
- Reduced latency: a lightweight server can move closer to its clients
- Load balancing: loads may move from one machine to the other within a network etc.

The route of the mobile agent can be decided by its owner or it can decide its next hop destination on the fly.

Here, we assume the underlying network to be a Mobile Ad Hoc Network (MANET) that typically undergoes constant topology changes, which disrupt the flow of information over the existing paths. Mobile agents are nowadays used in MANETs for various purposes like service discovery [4], network discovery, automatic network reconfiguration etc.

Dependability of any computing system may be defined as the trustworthiness of the system, which allows reliance to be justifiably placed on the service it delivers [5]. It is an integrative concept that encompasses attributes like availability (readiness of usage) and reliability (continuity of correct service) [5]. In MANET, like in any other mobile distributed system, mobile nodes access information through wireless data communication at any time and everywhere (motion and location independence) [6]. Therefore, this environment itself introduces new features and aspects to dependability, affecting both availability and reliability of the services of distributed systems.

Hence the reliability of underlying network becomes a factor that may affect the performance, availability, and strategy of mobile agent systems [7] [8].

In this paper, we define a Mobile Agent-based System (MAS) to be a system consisting of a number of different groups of agents where each group accomplishes an independent task.

The connectivity between the nodes is calculated according to the two-ray model [9] for signal propagation reflecting multipath propagation effect of radio signals. The node movements are assumed to be smooth as is the case in most real life scenario. Smooth Random Mobility Model (SRMM) [10] is used for this purpose. We propose a randomized agent planning strategy where an agent selects a destination almost randomly giving preference to a list of nodes over the others and the routes are also updated dynamically, in order to incorporate node mobility, as agents roam in the network. We estimate the reliability of such a mobile agent based system using Monte Carlo simulation technique. This technique is used to avoid the typical computational complexity that may arise.

Some contemporary work in this area is discussed in Section II. Our work in reliability estimation is presented in details in the subsequent section (III). The simulation results of our reliability model are summarized in Section IV. Finally, Section V concludes with an indication of our future endeavor in this area.

II. RELATED WORKS

Reliability analysis of MAS in adhoc network is a complicated problem for which little attention has been paid. Most of the work done in this area is related to distributed systems and distributed applications. But as pointed out in [8], features like scalability and reliability becomes critical in challenging environment with wireless networks. However, the scalability/reliability issue of MAS has been highlighted in [11], although the work does not focus on MANET. We did not see any work that considers transient environmental effects (apart from node mobility) into the reliability calculation for MANET.

A. Reliability of Distributed Systems

Two reliability measures are introduced in [12], distributed program reliability and distributed system reliability. Here graph traversal is used in designing an efficient method to evaluate the proposed measures.

In [13], a unified algorithm is proposed to efficiently generate disjoint file spanning trees by cutting different links, and the distributed program reliability and distributed system reliability are computed based on a simple and consistent union operation on the probability space of the file spanning trees.

In [14], two algorithms are proposed for estimating the reliability of a distributed computing system with imperfect nodes. One is called symbolic method (SM), is based on a symbolic approach that consists of two passes of computation, and the other algorithm, called factoring method (FM), and employs a general factoring technique on both nodes and edges.

B. Mobile Ad Hoc Network

In [15], Toh et al. describes a MANET as a collection of two or more devices equipped with wireless communications and networking capability. This definition is expanded further by explaining the method by which their networking capability is realized. Like point to point radios, ad-hoc devices can communicate directly with other devices within their range. They may also communicate with those outside their range by using intermediate nodes to relay or forward the message to the destination node. This second capability, multi-hop communications without the need for network infrastructure is what makes MANET unique.

Research on ad-hoc networks generally focuses on the modification and creation of protocols in the network and transport layer, such as Transmission Control Protocol/Internet Protocol (TCP/IP) to accommodate the mobility of the nodes and make network performance more robust. In [16], Ye et al. proposed a deployment strategy to increase probability of a ‘reliable path’. The increase in path reliability was accomplished through strategic node placement, limiting the application to instances where node mobility be directed. In [17], a protocol is proposed to accommodate the probabilistic reliability of a MANET but it does not explicitly measure network reliability.

C. Reliability of MANET

Due to the analytical complexity and computational cost of developing a closed-form solution, simulation methods, specifically Monte Carlo (MC) simulation are often used to analyze network reliability. In [18], an approach based on MC method is used to solve network reliability problems. In this case graph evolution models are used to increase the accuracy of the resultant approximation. In [19], a MC method is designed to estimate network reliability in the presence of uncertainty about the reliability of both links and nodes.

But little has been addressed on the reliability estimation of MANETs. In [20], analytical and MC-based methods are presented to determine the two-terminal reliability for the adhoc scenario. Here the existence of links was considered in a probabilistic manner to account for the unique features of the MANET. However, there remains a gap in understanding the exact link between a probability and a specific mobility profile for a node. In [21], MC-based methods are presented to determine the two-terminal reliability for the adhoc scenario. This work is an extension of that in [21], by including directly, mobility models in order to allow mobility parameters, such as maximum velocity, to be varied and therefore analyzed directly. The methods in this paper will now allow for the determination of reliability impacts under specific mobility considerations. As an example, one may consider the different reliability estimate when the same networking radios are used to create a network on two different types of vehicles. Here node mobility is simulated using Random Waypoint mobility model [22]. But this Random Waypoint model of mobility being a very simple one often results in unrealistic conclusions.

D. Reliability of Mobile Agents

Little attention has been given to the reliability analysis of MAS. In [23], two algorithms have been proposed for estimating the task route reliability of MAS depending on the conditions of the underlying computer network. In [24], which is an extension of the previous work, a third algorithm based on random walk generation is proposed. It is used for developing a random static planning strategy for mobile agents. However, in both the works the agents are assumed to be independent and the planning strategy seemed to be static. So this work does not address the scenario where agents can change their routes dynamically. Moreover, it does not address the issue of node mobility in between agent migrations.

In [1], a preliminary work has been done on estimating reliability of independent mobile agents roaming around the nodes of a MANET. The protocol considers independent agents only. Node and link failure due to mobility or other factors is predicted according to NHPP. Explicit node movement according to some mobility model is not considered. An agent may migrate to any node with equal probability. This may not be not realistic as some nodes may provide richer information for a particular agent deployed by some application. In [25], the MAS is assumed to be consisting of a number of agent groups demanding for a minimum link capacity. Thus, each agent group requires different channel capacity. Hence, different groups perceive different views of the network. In this scenario the reliability calculation shows that even with large number of heterogeneous agent groups with differing demands of link capacity, the MAS gradually reached a steady state.

III. Our Work

Though mobile agents are recently used in many applications of MANET, dependability analysis of such applications is not much explored. However, attributes like scalability, reliability and availability are affected by the dynamic network topology of MANET. However the scalability/reliability issue of MAS has been highlighted in [11], although the work does not focus on MANET. However, we have done some work on estimating reliability of wireless networks (in [26]), where nodes move according to some mobility model like Smooth Random Mobility Model [10]. But mobile agents are not considered in [26].

Moreover, we have done some preliminary work [1] [25] on agent reliability but it does not consider several issues that are considered in the present work.

A. Terminologies used in this paper

(V,E)	the graph (G) representation of our network;
N	no. of mobile nodes;
S	our mobile agent based system;
M	no. of mobile agents that constitutes S and are deployed in the network; thus, S= $\{m_1, ..m_i\dots m_M\}$
R_s	reliability of S;
n	no. of nodes successfully visited by an agent;
$\lambda_i(t)$	task route reliability of i^{th} agent in a step of simulation;
$\lambda(t)$	average reliability of all the agents;
L(t)	an array of length NxN
$r_i(t)$	the probability that m_i is working correctly at time t that is the individual software reliability of m_i;
Gt,Gr	transmitter and receiver gain respectively;
ht,hr	height of the transmitting and receiving antenna;
d_{ij}	the distance between nodes i and j
Q	no. of simulation steps;

B. Problem Definition

In this paper, we assume that our mobile agent-based system (S) consists of M independent agents deployed by k owners that may move in the underlying MANET. The reliability of (S) is defined as the probability that (S) is operational during a period of time [2]. Consequently S is said to be fully operational if all its currently existing mobile agents are functional or operational [3], whereas it is fully down if all its currently existing mobile agents are fully non-operational. Moreover, (S) is said to be partially operational if some of its currently existing mobile agents are operational. Later, in Section III.C we define reliability of an individual agent in this context.

1) Modeling MANET: We model the underlying network as an undirected graph G= (V,E) where V is the set of mobile nodes and E is the set of edges among them. Let the network consist of N nodes, thus, |V|=N that may or may not be connected via bidirectional links (e). The following assumptions are made ([27] [28]):

1) The network graph has no parallel (or redundant) links or nodes.
2) The network graph has bi-directional links.
3) There are no self-loops or edges of the type (vj, vj).
4) The states of vertices and links are mutually statistically independent and can only take one of the two states: working or failed.

Initial locations of the nodes (v_is) are assumed to be provided. The mobility of nodes in MANET can be simulated using SRMM [10]. This model is like Random Waypoint Mobility Model [10] but more realistic as it prevents the nodes from taking sharp turns or making sudden stops.

To incorporate SRMM [10] a Poisson event determines the time instant of change in speed. A new speed is chosen from the interval $[0,V_{max}]$ where 0 and V_{max} are given higher preference and rest of the values are uniformly distributed. Once a target speed is chosen the current speed is changed according to the acceleration a(t), which is once again uniformly distributed in $[0, a_{max}]$. The values of V_{max} and a_{max} may be different for different users. For example, for vehicular traffic, these will have higher values than pedestrians. Thus, as in [10],

$$v_i(t) := v_i(t-\Delta t)+a_i(t)*\Delta t \tag{1}$$

A new target direction is chosen only when $v_i(t)=0$. We simulate here the *stop turn and go* [10] behavior. The target direction is uniformly distributed between $[-\pi/2, \pi/2]$ with $\pi/2$ and $-\pi/2$ having higher priorities [10]. At every time instant direction $(\Delta\varphi_i(t))$ changes incrementally $(\Delta\varphi_i(t))$ unless it attains the target direction. Thus, as in [10],

$$\varphi_i(t)= \varphi_i(t-\Delta t) + \Delta\varphi_i(t) \tag{2}$$

Now, using this speed at previous time instant, acceleration, and direction, we can estimate the position (x_i,y_i) of the node at $(t+\Delta t)$ as

$$x_i(t+\Delta t)=x_i(t)+\Delta t*v_i(t)*\cos\varphi_i(t)+0.5*a_i(t)*\cos\varphi_i(t)*\Delta t^2 \tag{3}$$

$$y_i(t+\Delta t)=y_i(t)+\Delta t*v_i(t)*\sin\varphi_i(t)+0.5*a_i(t)*\sin\varphi_i(t)*\Delta t^2 \quad (4)$$

The movement of the nodes is assumed to be bounded within a specified simulation area as in [10].The distance between a pair of nodes (d_{ij}) can be calculated as follows

$$d_{ij}(t) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (5)$$

The probability of link existence (P_{link}) not only depends on the distance between the nodes but is also very much dependent on the environmental factors. So, even when two nodes remain within the transmission range of each other, but due to factors like signal fading, shadowing, diffraction etc., the quality of transmission can degrade appreciably [29]. The average received power (p_r) is a function of the distance between the transmitter and the receiver. Here we take the two-ray model for radio propagation in order to show how the transmitted signal with power (p_t) suffers from multipath propagation while reaching the receiving end. Thus, $p_r(d)$ can be stated as mentioned below [9]:

$$p_r(d) = p_t G_t G_r \frac{h_t^2 h_r^2}{d^4} \quad (6)$$

In free space, the received power varies inversely to the square of the distance but here we have assumed the exponent to be 4 to indicate the presence of a medium.

2) Modeling Mobile Agent Based System: In this scenario we can think of a mobile agent as a token visiting one node to another in the network (if the nodes are connected) based on some strategy as needed by the underlying applications to accomplish its task.

An agent starts its journey from a given owner and moves from one node to another at its will. The owner provides a priority list to the agent, which contains a list of node ids that are most beneficial migration sites (for the application that deployed that particular agent). So, an agent will always try to visit those nodes from the priority list as its first preference. But this movement is successful if the two nodes are connected and there is no simultaneous transmission in the neighborhood of the intended destination. We assume that cases of collisions (if any) are taken care of by the underlying MAC protocol. So, we associate a probability with the movement to indicate transient characteristics of the environment, since, for example, the routing table may not be updated properly or the link quality may have degraded so much (due to increased noise level) that the agents are unable to migrate. Thus, if an agent residing at node A decides to move to node B (connected to A) then the agent successfully moves to B with probability p_{tr}. Here p_{tr} denotes the problem of unpredictable background noise level mentioned above. For example, noise level may increase due to heavy rainfall.

Let us suppose that at an instance t, the MANET consists of five nodes namely MN_A, MN_B, MN_C, MN_D and MN_E and their connectivity is as shown in Figure 1.The dotted line represents an erroneous link. We assume that all the nodes have appropriate host platform for the agents and the agents may update their migration policy on the fly. An agent x (say) residing at node A does the following:

1) It chooses its next destination almost randomly giving more preference to the nodes in the priority list. If that destination is not visited before and if there is a path then x moves to its new location with probability p_{tr}.

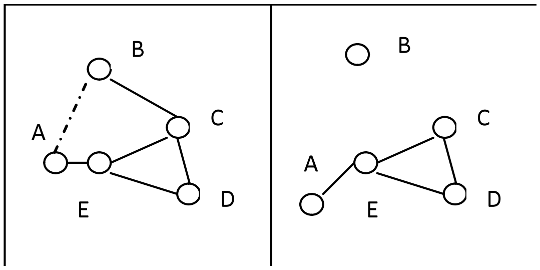


Figure 1. An instance of a network graph at instant t (left) and t+Δt (right) respectively

2) But when x attempts to move to MN_B at (t+Δt) time instant, the network graph changes (Figure 1) and MN_B becomes an isolated node, which is unreachable. It may also happen that the capacity of the link (from MN_A to MN_B) is lower than that needed by x. So for the underlying routing algorithm, a link exists between MN_A and MN_B but for agent x, the capacity of the link is not sufficient. So MN_B is unreachable for x.
3) So x will not be able to move to MN_B.
4) In the next time instant x may retry or try to choose its next destination randomly again.

This helps in the improvement in system performance. This is because of the fact that the agents themselves try to overcome the transient faults.

3) Modeling Agent Reliability:

In this scenario we study the reliability of MAS with respect to the network status and its conditions (for example, connectivity of the links, path loss probability etc.). We start with a dynamic planning strategy where each agent is expected to visit N (<=number of nodes in the network) nodes in the network to accomplish its task. Each group of agents starts its journey from a given node, which acts as its owner. We assume that a node can only own a single group of agents. In other words, a node can only host one application that will deploy a number of agents. Due to the constraints of mobile nodes (MN) such assumption is not absurd at all.

We have taken the failure probability (P) of the mobile nodes (P_{Node}) to be a variable of Weibull distribution [21].

Now reliability of MAS (R_s) can be defined as

$$R_s = \{R_{MAS}|R_{MANET}\} \tag{7}$$

Here reliability of MANET (R_{MANET}) can be treated as an accumulative factor of ($1-P_{Node}$) and P_{Link}. P_{Link} can be treated as a combination of P (p_r is at an acceptable level) and the mobility model. Here p_r denotes the received power at node j after traversing distance d_{ij} from sender node i.

Here we calculate individual agent reliability on the underlying MANET as follows:

If an agent can successfully visit M nodes out of N(desired) then it has accomplished M/N portion of its task. Thus, reliability in this case will be M/N.

But if the application requires all N nodes to be visited in order to fully accomplish the task and in all other cases the task will not be considered to be done, reliability calculation will be modified as:

If an agent can successfully visit all N nodes desired then it has accomplished its task. Thus, reliability in this case will be 1. In all other cases it will be 0.

Above definitions of agent reliability works only if there is no software failure of the agent (assumed to follow Weibull distribution [21]).

Now, the probability that the MAS is operational i.e., reliability of MAS (R_{MAS}) can be calculated as the mean of reliability of all its components, that is, the agents in this system.

$$R_{MAS} = \frac{\sum\{Agent\,\mathrm{Re}\,liabilities\}}{No.ofAgents} \tag{8}$$

Finally to calculate R_s in equation 7 an algorithm is proposed in this paper in the next section.

C. Steps of Reliability calculation of mobile agent with dynamic route

1) SRMM is used to simulate the effect of node mobility.
2) The probability of the existence of a link is calculated according to equation 6 to cover multipath propagation effect of radio signals.
3) Breadth First Search (BFS) is used iteratively to identify the connected components (clusters) of the network and are given unique identifiers (cluster id).
4) A mobile agent prefers to select a destination, which is not visited before, from the priority list. If it finds a route (that is if the source and destination share the same cluster id) then it moves with a certain probability and the process continues otherwise the process halts.
5) Individual node failure is also considered and Weibull distribution [21] is used to simulate the same. Weibull distribution takes two parameters, scale and shape. We have given the values in such a way that as time passes on the probability of failure also increases.
6) Finally, Monte Carlo method of simulation is used to find the overall reliability.

1) Input parameters: M (number of independent mobile agents in the system), The initial state of the network (node position, location, speed of the nodes)

2) Detailed Steps:

1. Initialize n (that is the number of mobile nodes successfully visited by an agent) to 0 and a source for the mobile agent.
2. List of vertices along with their initial positions is given.
3. The priority list for each agent group is also formed and kept with the owners.
4. i. To simulate the effect of node mobility create E', a subset of VXV with the same using SRMM as follows.
    a. The $v_i(t)$ and $\varphi_i(t)$ are calculated using equation (1) and (2) respectively.
    b. The position of each MN is updated for the next time increment by equation (3) and (4).
    c. Distance between each pair of nodes is calculated using equation (5) and E' is populated according to equation (6).

   ii. Some nodes may also fail because of software/hardware failure or become disconnected from the network according to NHPP distribution. Node failure can be simulated by deleting the edges e from E' further that are incident on the failed node $v \in V$.
5. According to Weibull distribution we find individual software reliability r_i for an agent i.
6. BFS is used unless all connected subgraphs are assigned a proper cluster id. Thus, an isolated node is also a cluster.
7. The agents perform their job on this modified graph.
    a. An agent will prefer to choose a node to be its next destination if it is in its priority list and is not visited already. All other nodes (not there in the priority list) are equally likely destinations.
    b. If that destination falls in the same cluster as it is now residing, the agent moves to the new destination with probability p that represents the instantaneous background noise level in the network. If it succeeds, n is incremented by 1.
    c. Despite several attempts that an agent may make, if an agent fails to move to its next destination (say $node_i$), then,
        i. the agent tries to move to other destinations as needed by the application.
8. Repeat steps 3 to 6 until all nodes are visited or the new destination falls in a different cluster.

```
9.  Calculate $\lambda_i(t) = \frac{n}{N}$   (9)

    Here the value of n depends heavily on the conditions
    of the underlying network.
10.     Reset the value of n.
11. Repeat steps 5-9 for all agents (k) in the system.
12. Calculate $\lambda(t) = \frac{1}{k}\sum_{i=1}^{k}\lambda_i(t)r_i$   (10)

13. Repeat steps 3 to 11 Q (simulation steps) times.
14. Calculate node reliability $\frac{1}{Q}\sum_{q=1}^{Q}\lambda(q,t)$   (11)
```

It is to be mentioned that step 4 is repeated for every move of the mobile agent. Since in a typical adhoc scenario we cannot assume the nodes to be static during the entire tour of the mobile agents so after every single move the entire network configuration (hence the effect of node mobility) is recalculated. Moreover in this case E' does not have to be a subset of E because with time some nodes may also move closer to the other nodes and thus, creating a link between them.

If an agent fails to move because of background noise level, then it may retry depending on the amount of delay that the respective application can tolerate.

Here we have assumed that in order to accomplish a task the agents need to visit all the nodes in the network. So we have N as the denominator in equation 9. But we can change this parameter and our algorithm will still work if lesser number of nodes is needed to be visited. We have also assumed that the agent can always retract to its owner.

It may be seen in practice that in a network some nodes have rich information and the agents tend to move to those nodes as their next destination over the other. That is why, we prioritize the nodes by providing a priority list rather than randomly selecting the next destination in step 7 of the algorithm. Here we feed the priority list from owners but the agents may also learn about such rich nodes from their experience and may share this information also with the others using some multiagent communication scheme like the blackboard model [30]. A mobile agent may leave a message for another agent at one of the N hosts. Whenever the dependent agent comes to that host it will receive that message and act accordingly. So, the node priorities can also be modified on the fly. This is a possible application of learning [31] in this system.

TABLE I. PRIORITY LIST OF THE AGENTS

Agent Id	Priority List
Agent 1	*MN$_2$,MN$_4$*
Agent 2	*MN$_1$, MN$_3$*
Agent 3	*MN$_4$*
Agent 4	*MN$_1$, MN$_2$*

D. An Example

We have taken an instance where there are ten nodes in the network. Four mobile agents are deployed by four different owners and they start their journey from their owners. Agents 1, 2, 3 and 4 start their journey from nodes MN$_1$, MN$_2$, MN$_3$ and MN$_4$ respectively and roam around the network to accomplish its task. Thus, an application (for example, service discovery) running on MN$_1$ deploys agent 1. Our job is to find the number of nodes that are successfully visited by these agents, which indicates the progress of its task (how many services the agents discover for a MANET) and consequently the reliability of the agent group will be calculated. Average reliability of all groups taken over a certain time period for a number of simulations represents the reliability of the MAS despite the uncertainties of MANET. So, for reliability calculation we are giving equal priority to all nodes. However, our migration policy gives some nodes higher weight over the others (step 7a in the algorithm) indicating the fact that all destinations are not equally likely. The agents are fed with a given priority list by their respective owners as shown in TableI. For example, visiting nodes MN$_2$ and MN$_4$ will be most beneficial for agent 1 and so on.

The nodes are taken close enough (Figure 2) so that they form an almost connected network. As shown in Figure 3a, MN$_9$ is isolated from the MANET initially. But eventually it finds MN$_{10}$ within its range and hence can connect itself to the network (Figures 3b, c and d). This strategy of node distribution sounds realistic as the nodes in a MANET may not remain connected to each other always due to individual node movement and environmental characteristics.

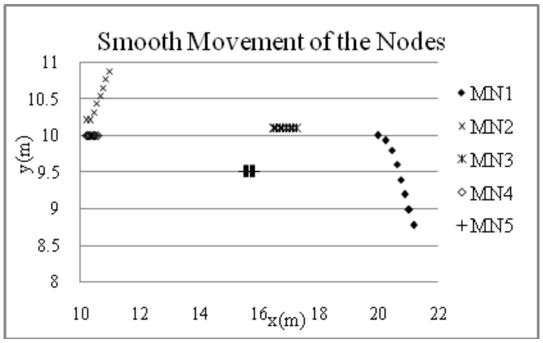
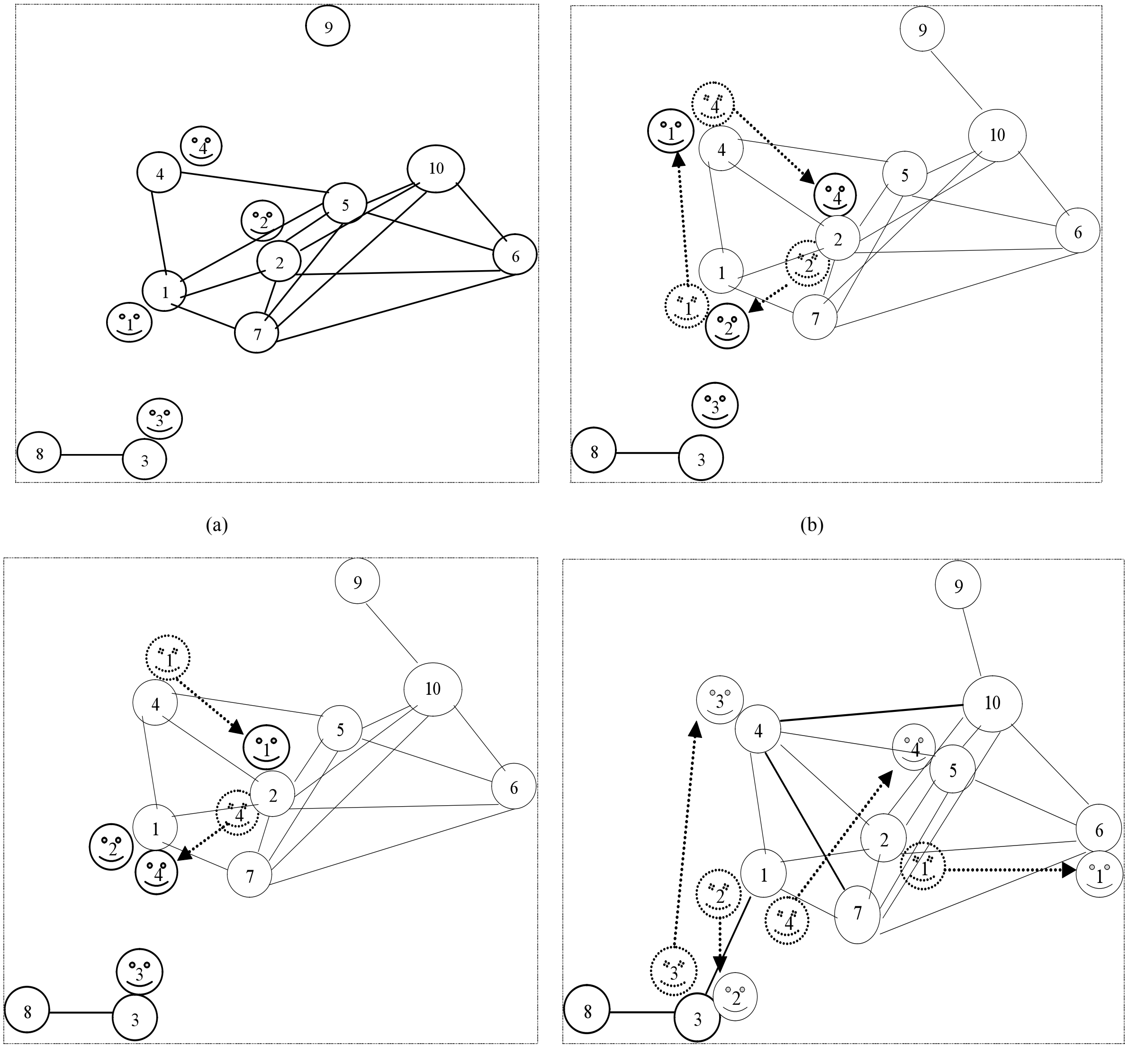


Figure 2. Movement of the nodes according to SRMM

Every 3 seconds the positions of the nodes are updated according to SRMM. The simulation is carried out for 30 seconds and the positions of the different nodes are given in Figure 2. The smooth movement of the nodes is obvious from the figure itself. Connectivity of the nodes is calculated according to the Two-ray model. For convenience we have only shown four nodes to be deploying agents. The network topology at 4 successive time instants is shown in Figure 3(a, b, c and d). Agents are also shown in Figure 3 by callouts along with a numeral to indicate agent ids. The dotted ones (callouts) represent the starting position and the bold ones (callouts) represent end point of their journey at that time instant.

Figure 3(a) indicates a disconnected network graph for the MANET with an isolated node (MN_9) and two components (clusters). Nodes, MN_3 and MN_8 form one



(a)

(b)

(c)

(d)

Figure 3.

a. Network graph at time instant t=t0 and the position of the agents

b. Network graph at time instant t=t0+Δt and the position of the agents

c. Network graph at time instant t=t0+2Δt and the position of the agents

d. Network graph at time instant t=t0+3Δt and the position of the agents cluster and all other nodes (except MN$_3$, MN$_8$ and MN$_9$) fall into a different cluster. Here any agent can move to any destination it wants to within its cluster. The agents start their journey in such a scenario.

While the nodes move and form a network configuration as shown in Figure 3(b), the agents also start migrating in the network. The network connectivity is slightly changed here as MN$_9$ now comes within the transmission range of MN$_{10}$ and hence becomes connected to one of the clusters. So our MANET now contains two clusters, (one containing MN$_3$ and MN$_8$ and the other containing the rest). Since agent1 gives highest priority to MN$_2$ and MN$_4$ over the others so, agent1 first visits MN$_4$. For similar reasons, agent2 visits MN$_1$ (from MN$_2$) and agent4 visits MN$_2$ (from MN$_4$) respectively. But agent3 cannot migrate successfully as node MN$_4$, the highly beneficial migration site for agent3, lies in a different cluster.

Network connectivity changes a little in the next 3 seconds as indicated in Figure 3c. So, agents 1 and 4 make successful migrations to their highly preferred destinations such as MN$_2$ (from MN$_4$) and MN$_1$ (from MN$_2$) respectively. However, MN$_3$, a highly beneficial migration site for agent2 falls in a different cluster than MN$_1$ (where agent2 currently resides). Consequently, agent2 cannot make any migration but stays at MN$_1$. Moreover due to transient characteristics, the link between nodes MN$_3$ and MN$_8$ becomes erroneous. As a result agent3 makes an unsuccessful attempt (step 7b in the algorithm) to migrate to MN$_8$ (from MN$_3$) but stays at MN$_3$. As the agents are sent with a given probability, even if nodes fall in the same cluster, an agent may not be able to make a successful migration. This scenario indicates the notable effect of transient errors on the performance of MAS.

Finally in the next 3 seconds the collection of nodes form a connected graph as MN$_3$ comes within the transmission range of MN$_1$. Now the agents can migrate to any other node with a certain probability (step 7 of our algorithm). Thus, agents 1 and 4 migrate to MN$_6$ (from MN$_2$) and MN$_5$ (from MN$_1$) respectively. Agents 2 and 3 also finally find their most beneficial migration sites (MN$_3$ for agent 2 and MN$_4$ for agent 3) reachable and attempt to make successful migrations.

In this way, the simulation is continued and the nodes in the MANET continued to form different network configurations affecting agent migrations. The value for received power is taken to be 16dBm. In the calculation the antenna gains are taken to be 2.2dBi, the height is taken to be 2m and the transmitting power is taken to be 20dBm [32].

TABLE II. VARIATION OF RELIABILITY WITH NO. OF MONTE CARLO SIMULATION STEPS

Q	Reliability
10	0.539
100	0.5315
500	0.5319
1000	0.5314
2000	0.5312
10000	0.5319

At the end of the 10th second, agents 1, 2 and 4 finish migration to 9 nodes each (including their owners) out of all 10 nodes in the MANET accomplishing (9/10 that is) 90% of their tasks each. However agent3 was only able to cover 7 out of 10 nodes (70%) because its owner MN$_3$ was disconnected from most nodes of MANET for a while, thus, accomplishing only 70% of its task. This scenario shows the effect of MANET configuration on the performance of MAS. Thus, the overall reliability of MAS comes out to be (3*0.9+0.7)/4=0.85 that is 85%. If another simulation run is carried out for the same amount of the time, then the overall reliability comes out to be 0.825. If we use Monte Carlo simulation for a number of times (Q=100 onwards) the overall reliability tends to converge to 0.53 (as shown in Table II). Thus, with a MANET of 10 nodes moving according to SRMM, the MAS where the agents almost randomly choose their neighbor and migrate, will be 53% reliable.

IV. EXPERIMENTAL RESULTS

The simulation is carried out in Java and it can run in any platform. The initial positions of the MNs are given along with their initial speeds and the maximum acceleration that can be attained by them. All agents of the same group start from the same node, that node are designated to be the owner. The maximum allowable speed and acceleration of the MNs are read from a file. These values are needed by SRMM. The simulation time is taken to be 1 hour. For the rest of the experiments, the number of nodes is taken to be 40 unless stated otherwise. The other parameters like received power, antenna gains are kept the same as mentioned in the example (Section IIID). Unless otherwise stated the number of agents is taken to be 30 and the number of groups is taken to be 4. In MANET due to environmental factors like diffraction, fading along with asynchronies in movement pattern, some nodes become isolated from the network. Some of the nodes may rejoin and some remain disconnected from the network. So, to start with we have taken such a scenario (of MANET) in terms of initial node positions and respective speeds.

With four (4) groups and a total of 30 mobile agents, if we increase the MANET size, the reliability is found to drop eventually as shown in Figure 4. This result is in concurrence with the one we get in [1]. Here at every step we add approximately 10 nodes but almost none of them remain within the transmission range of any of the disconnected components of the existing MANET. This is not also possible in a MANET with an appreciable diameter. So the number of successful agent migration reduces as more nodes become unreachable for an agent. Consequently at each step there is no drastic change in network connectivity as can be observed in [1], just the size of some disconnected components increase. This results in the gradual fall in reliability with increasing N.

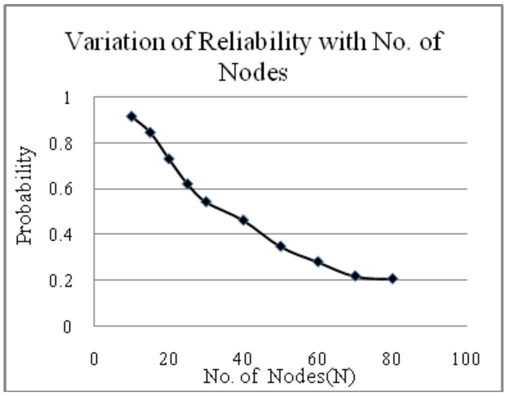


Figure 4. Variation of reliability with increasing network size

Now we look into the matter in more details for MANET with fast moving nodes. The maximum acceleration of the nodes is varied to yield different standard deviations for a given mean. When the average of all the maximum acceleration that a node can attain is 0.75, we plotted the reliability value for standard deviation = 0.1, 0.2, 0.3, 0.4 and 1. Similar things have been done for average value of 1.5 and 3 as shown in Figure 5. In most cases for a given standard deviation, higher mean implies lower reliability. So, this indicates the fact that when all nodes have the same variance in speed, if the overall MANET nodes are slower then obviously, the nodes will remain crowded implying higher reliability. On the contrary, for a given mean, higher the standard deviation, lesser will be the reliability. This indicates that when all the nodes move with comparable speed (lower standard deviation), for example, group movement in disaster relief or military operations, overall reliability improves. But when some nodes lag behind the others, reliability of MAS would get hampered as the MANET breaks into a number of clusters.

The above mentioned conclusion is valid irrespective of the MANET nodes being slower or faster. Thus, in Figure 6, the points at the peak of the curve yield lower standard deviation. But if the mean goes even higher, that is for faster MANET, the reliability of MAS reduces as shown in Figure 6.

The effect of background noise is observed in Figure 7. As the environment becomes noisier, such as urban areas, interference is higher. So the receiver would not be able to decode the signal if the received signal power is low. Thus, a weak signal having signal power of 8dBm could not be decoded in crowded areas. But for environment with lower interference, such as highways or countryside, the transmission range increases, enabling weaker signals (having power of 8-15dBm) to be detected and decoded properly. Hence network connectivity improves making MAS more reliable.

We have seen that if node movements are allowed only at the beginning before the mobile agents start their task route, then performance of the algorithm does not vary appreciably with number of mobile agents deployed in the system. But if the situation is made more realistic by allowing node movements in between agent migration then reliability of MAS varies with its size as shown in Figure 8. As far as the number of agent groups remains fixed (heterogeneity), the increasing size of MAS (in terms of M) does not seem to affect reliability greatly. But if the heterogeneity among agents increases, even for a fixed size of MAS, reliability improves and slowly reaches a stable

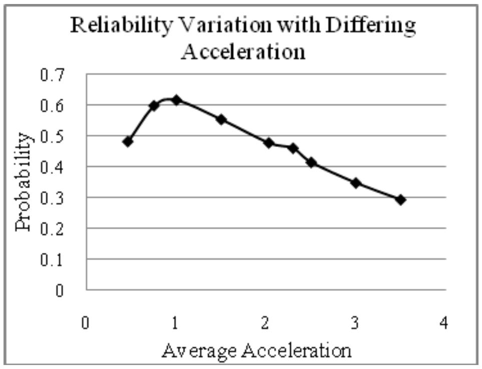


Figure 6. Variation of reliability with faster nodes

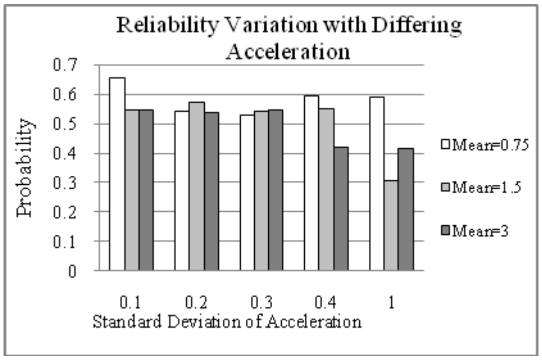


Figure 5. Variation of reliability with greater variation of node accelerations

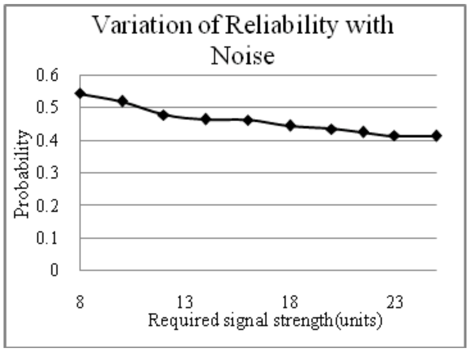


Figure 7. Reliability Variation with Noise

state. This result is significant as it shows that a large number of applications deploying different types of agents (having different migration pattern) does not hamper the reliability of MAS. Rather they cover the different parts of the network in a better manner and can better exploit the denser portion of MANET. So, an increasing number of heterogeneous agents yield better performance than a single group of homogeneous agents of comparable size. This is because the homogeneous agents have similar migration pattern, they start from the same region of MANET and tend to face similar connectivity problems.

Let us now concentrate in the migration pattern of the agents. As we know, every agent is provided with a preferred list of migration sites (priority list of the agents) by their owner. Longer the priority list wider will be the agent's scope to choose its next destination. But still, the probability of successful agent migration remains highly dependent on the position and connectivity of the next destination. Hence as shown in Figure 9, only a little improvement can be observed for longer priority list.

Keeping all parameters fixed if we increase the simulation time, the MANET diameter increases, thus, decreasing the overall reliability of MAS. But after some time the network connectivity somewhat stabilizes, thus, reducing any further the rate of change of reliability with time and the system enters a somewhat stable state (shown in Figure10).

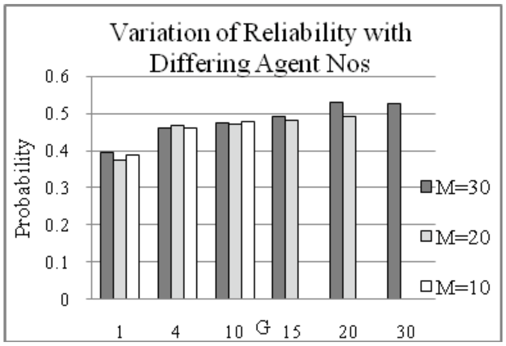


Figure 8. Reliability variation with increasing no. of agents and agent groups

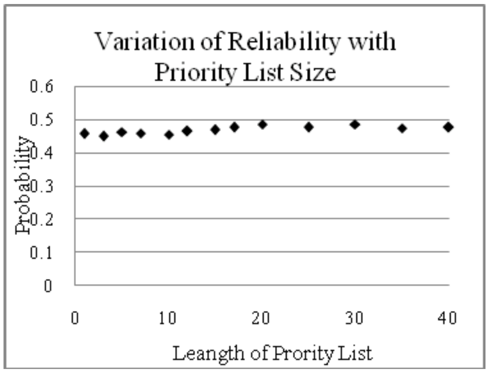


Figure 9. Effect of varying priority list size on MAS reliability

V. Conclusion

In this paper, a scalable approach to estimate the reliability of a mobile agent based system for MANET is presented. The reliability calculation depends heavily on the conditions of MANET, like area covered by a node, size of MANET and of course, node mobility.

A starting point for the agents is provided. However the agents are not fed with a given route, rather a list of preferred migration sites are mentioned, which is quite practical. The agents show slightly better reliability if more nodes are designated as preferred migration sites, that is, the agent's scope becomes a little wider.

SRMM is used to simulate the movement of the nodes. The protocol is validated and results are shown in Section IV. It can be observed that for a faster MANET only if all the nodes move with comparable speeds then MAS is found to be appreciably reliable. Higher background noise is also found to hamper the reliability of MAS.

As can be seen, reliability improves heavily if the agent set is sufficiently heterogeneous, despite the dynamics and uncertainties associated with MANET. This work does not consider agents with differing QoS requirements for migration.

We are planning to include (i) mobile agents designed specifically for an application like service discovery and (ii) security characteristics to design a fully dependable system.

Reference

[1] C. Chowdhury and S. Neogy, "Estimating Reliability of Mobile Agent System for Mobile Ad hoc Networks", Proc. 3rd International Conference on Dependability, DEPEND 2010, pp. 45-50, 2010.

[2] J. Cao, X. Feng , J. Lu, and S. K. Das, "Mailbox-Based Scheme for Designing Mobile Agent Communications", Computer, Vol. 35 n.9, pp. 54-60, September 2002.

[3] N. Migas, W.J. Buchanan, and K. McArtney, "Migration of mobile agents in ad-hoc, Wireless Networks", Proc. 11th IEEE International Conference and Workshop on the Engineering of Computer-Based Systems, pp. 530 – 535, 2004.

[4] R.T. Meier, J. Dunkel, Y. Kakuda, and T. Ohta, "Mobile agents for service discovery in ad hoc networks", Proc. 22nd International Conference on Advanced Information Networking and Applications, pp. 114-121, 2008.

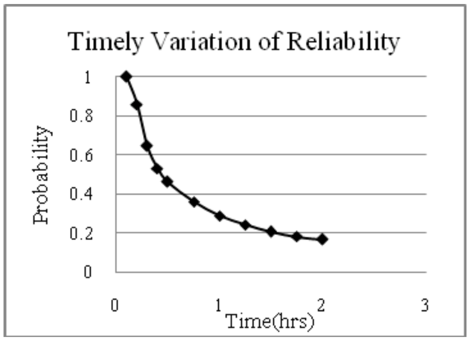


Figure 10. Timely variation of reliability

[5] R. H. Laamanen, T. Alonko, and K. Raatikainen, “Dependability issues in mobile distributed system”, Proc. of the Pacific Rim International Symposium on Dependable Computing, pp. 7-14, 1999.

[6] A. Avizienis, J. Laprie, and B. Randell, “Fundamental concepts of dependability”, LAAS-CNRS, Technical Report N01145, Apr. 2001.

[7] A.L. Murphy and G. P. Picco, “Reliable communication for highly mobile agents”, Autonomous Agents and Multi-Agent Systems, Vol. 5, Issue 1, pp. 81-100, 2002.

[8] O. Urra, S. Ilarri, and E. Mena, “Agents jumping in the air:dream or reality”, Proc. 10th International Work-Conference on Artificial Neural Networks, IWANN’09, Special Session on Practical Applications of Agents and Multi-Agent Systems, pp. 627–634, 2009.

[9] M. Rooryck, “Modelling multiple path propagation-Application to a two ray model”, in the journal of L'Onde Electrique , Vol. 63, pp. 30-34, Aug.-Sept. 1983.

[10] C. Bettstetter, “Smooth is better than sharp: a random mobility model for simulation of wireless networks”, Proc. 4th ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems, pp. 19-25, 2001.

[11] S. Ilarri, R. Trillo, and E. Mena, “SPRINGS: A scalable platform for highly mobile agents in distributed computing environments”, Proc. 4th International WoWMoM 2006, Workshop on Mobile Distributed Computing (MDC 2006), pp. 633–637, 2006.

[12] C. S. Raghavendra, V.K.P. Kumar, and S. Hariri, “Reliability analysis in distributed systems”, in the IEEE Transactions on Computing, Vol. 37, Issue 3, pp.352–358, 1988.

[13] D.J. Chen, and T. H. Huang, “Reliability analysis of distributed systems based on a fast reliability algorithm”, in the IEEE Transactions on Parallel Distributed Systems, Vol. 3, Issue 2, pp.139–154, 1992.

[14] M.S. Lin, D.J. Chen, and M.S. Horng, “The reliability analysis of distributed computing systems with imperfect nodes”, The Computer Journal, Vol. 42, Issue 2, pp. 129–141, 1999.

[15] C-K Toh, “Ad Hoc Mobile Wireless Networks”, Prentice Hall PTR © 2002.

[16] Z. Ye, S. V. Krishnamurthy, and S. K. A Tripathi, “Routing Framework For Providing Robustness to Node Failures in Mobile Ad Hoc Networks”,in Ad Hoc Networks, Vol. 2, No. 1, pp. 87-107, 2004.

[17] M. Abolhasan, T. Wysocki, and E. Dutkiewicz, “A Review of Routing Protocols for Mobile Ad Hoc Networks”, in Ad Hoc Networks, Vol. 2, No. 1, pp.1-22, January 2004.

[18] T.Elperin, I. Gertsbakh, and M. Lomonosov, “Estimation of network reliability using graph evolution models”, in the IEEE Transactions on Reliability, Vol. 40. No. 5. pp 572-581, 1991.

[19] M. Marseguerra, E. Zio, P. Luca, and D. W. Coit, “Optimal Design of Reliable Network Systems in Presence of Uncertainty”, in the IEEE Transactions on Reliability, Vol. 54, No. 2, June 2005.

[20] J. L. Cook and J. E Ramirez-Marquez, “Two-terminal reliability analyses for a mobile ad-hoc wireless network”, in Reliability Engineering and System Safety, Vol. 92, Issue 6, pp. 821-829, June 2007.

[21] J. L. Cook and J. E. Ramirez-Marquez, “Mobility and reliability modeling for a mobile ad-hoc network”, IIE Transactions, Vol. 41, Issue 1, pp. 23 – 31, 2009.

[22] C. Bettstetter and C. Wagner, “The spatial node distribution of the random waypoint mobility model”, Proc 1st German Workshop on Mobile Ad hoc Networks (WMAN), pp. 41-58, 2002.

[23] M. Daoud and Q. H. Mahmoud, “Reliability estimation of mobile agent systems using the Monte Carlo approach”, Proc. 19th IEEE AINA Workshop on Information Networking and Applications, pp. 185–188, 2005.

[24] M. Daoud and Q. H. Mahmoud, “Monte Carlo simulation-based algorithms for estimating the reliability of mobile agent-based systems” Journal of Network and Computer Applications, pp. 19–31, 2008.

[25] C. Chowdhury and S. Neogy, “Reliability Estimate of Mobile Agent Based System for QoS MANET Applications”, Proc. Annual Reliability and Availability Symposium 2011, RAMS 2011, pp.1-6, 2011.

[26] C. Chowdhury and S. Neogy, “Reliability estimation of fault-tolerant wireless and mobile networks”, Proc. 3rd International Conference on Communication Theory, Reliability, and Quality of Service, CTRQ 2010, pp. 67-72 2010.

[27] M. L. Shooman, “Reliability of computer systems and networks: fault tolerance, analysis, and design”. New York: Wiley; 2002.

[28] C. Srivaree-ratana, A. Konak, and A. E. Smith, “Estimation of all-terminal network reliability using an artificial neural network”, Computers and Operations Research, Vol. 29, pp. 849–868, 2002.

[29] H. Taub and D. L. Schilling, “Principles of Communication Systems”, McGraw-Hill, 1986.

[30] S. Li and F. Hu, **“**Communication between the RoboCup agents based on the blackboard model and observer pattern”, Proc. 5th international conference on Wireless Communications, Networking and Mobile Computing, pp. 5007-5011, 2009.

[31] R. S. Sutton and A. G. Barto, “Reinforcement Learning:An Introduction”, MIT Press, Cambridge, MA,1998.

[32] W. Stallings, “Wireless Communications and Networks”, Pearson, Second Edition, 2009.

Durable Solar Energy Harvesting from Limited Ambient Energy Income

Sebastian Bader, Bengt Oelmann
Department of Information Technology and Media
Mid Sweden University
Sundsvall, Sweden
{*sebastian.bader, bengt.oelmann*}@*miun.se*

***Abstract*—Typical wireless sensor network applications in the domain of environmental monitoring require or profit from extended system lifetime. However, restrictions in sensor node resources, especially due to the usage of capacity limited batteries, forbid these desired lifetimes to be reached. As opposed to batteries, energy harvesting from ambient energy sources enables for near-perpetual supply of sensor nodes, as the utilized energy source is inexhaustible. Nevertheless, the supply from ambient energy sources is rate-limited, wherein this supply-rate is mainly defined by the system deployment location. On the other hand, the attached sensor node has a consumption-rate, which has to be supplied to guarantee continuous node operation. In this paper, we address the matching of supply-rate and consumption-rate in solar energy harvesting systems at locations with limited insolation. The focus lies on the reduction of harvester energy overhead, which in low-duty cycled system easily reaches similar or higher consumption levels than the load it supplies. We suggest and present two harvester architectures [1], that have their main design consideration on simplicity. The individual modules of the architectures are tested and verified in laboratory measurements and we evaluate the fully implemented systems in an outdoor deployment. Based on the laboratory results, implementation choices for the architecture modules have been made. Whereas both harvesting architectures continuously supplied the attached load during the deployment period, we were able to compare their behavior with each other and present individual advantages and drawbacks.**

***Keywords*-energy harvesting; environmental monitoring; wireless sensor networks; energy sources; node lifetime**

I. Introduction

According to general believe, Wireless Sensor Networks (WSN) have the possibility to revolutionize the way we perceive and interact with our environment [2]–[4]. Combining sensing, control, processing and communication capabilities in relative small and inexpensive measurement systems, allows sampling at large-scale with high temporal and spatial resolution. This in turn offers advantages in a plethora of different application domains, increasing efficiency of existing applications and enabling a set of completely new functions.

In Environmental Monitoring, as one of these application domains, WSNs offer distributed and autonomous measurements with automatic data acquisition possibilities. Large areas of interest can be observed with a scalable number of sensing stations and flexibility in their positioning, while not demanding increased human interaction. Furthermore, connectivity through the network infrastructure allows sample transfer from all sensor nodes to a desired gathering point and remote access and control of these nodes.

Table I
Typical energy harvesting sources and power levels available in outdoor environments [8], [9]

Energy source	Power density	Condition
Solar	$100\,\mathrm{mW\,cm^{-2}}$	direct sunlight, outdoors
Wind	$100\,\mathrm{mW\,cm^{-2}}$	$9\,\mathrm{m\,s^{-1}}$, $10\,\mathrm{m}$ altitude
Ambient RF	$< 1\,\mathrm{\mu W\,cm^{-2}}$	unless close to emission source
Thermo-Electric	$60\,\mathrm{\mu W\,cm^{-2}}$	at 5 °C temperature difference

Nevertheless, application setup and deployment can require considerable amount of time and money [5]–[7], especially when numerous sensor nodes are involved. Therefore, lifetime and maintenance demands become an important issue, defining economical feasibility of this technology. Ideally, system operation should be indefinitely, uninterrupted and without requiring human involvement.

As an active electronic system, one primal requirement for system autonomy is the constant supply of power. Due to the typical inaccessibility of a fixed power infrastructure, energy storage devices - usually in form of batteries - are used as power sources. Though, as energy storage capacity of these devices is limited, autonomous lifetime of the systems these devices power, is inevitably limited as well.

As a result, energy harvesting attracts increasing attention in research involving Wireless Sensor Networks. Harnessing available ambient energy, such as from wind, sun, vibrations or temperature gradients, energy reservoirs in storage devices of limited capacity can be recharged on a regular basis. Because these ambient energy sources are not limited in their energy capacity (i.e., they are inexhaustible), but only in their supply rate, matching their supply rate with the load consumption demand enables perpetual energy supply.

Table I provides an overview of expectable power densities for different ambient energy sources, typically available to outdoor environmental sensor networks. While power densities can be quite high, it is strongly depending on deployment location and environment.

Solar energy harvesting is the most frequently used form of energy harvesting in outdoor Wireless Sensor Networks, which might find explanation in several of its properties. (i) Its conversion technology is rather mature and low-cost, because of the use in macro-scale energy production. (ii) Power densities are often sufficiently high. (iii) Available energy spreads over a wide area and conversion rate is easily scalable, and (iv) conversion does not require mechanical parts, leading to higher maintenance requirements.

Nonetheless, also for solar energy harvesting, achievable conversion rates are highly location specific. This means while there are locations providing good harvesting possibilities, there are also those where solar radiation is limited and insolation unequally distributed over the year. Research targeting micro-solar energy harvesting systems for the former case is documented plentiful in literature. Opposed to that, systems addressing low solar radiation environments are strictly limited.

In this paper, we address the issue of limited irradiation conditions for solar energy harvesting based outdoor sensor networks. The general architecture of solar energy harvesting power supplies is presented and analyzed towards limitations for use in low irradiance situations. Design considerations are made to allow the sufficient conversion of light into electricity for powering sensor sample-and-send sensor nodes in Environmental Wireless Sensor Networks, while at the same time avoiding lifetime limitations due to battery storage devices. Herein, the focus of the system lies on providing sufficient energy levels, guaranteeing uninterrupted operation at all times, opposed to optimization towards efficient energy conversion at times of strong irradiation.

The remainder of the paper is organized as follows. The next section summarizes a subset of existing related work in the area of solar energy harvesting systems. After that, Section III provides theory on general solar energy harvesting system architecture, location influence and application requirements, leading to the design considerations of solar power supplies for the intended scenario. Section IV will present measurement and evaluation setups, followed by measurement results, resulting system architecture and its evaluation in Section V. Finally, Section VI will conclude the results obtained.

II. Related Work

Plenty of work has been done in Wireless Sensor Networks for Environmental Monitoring, as typical applications in this domain gain from measurement capabilities this technology can provide. To mention only a small subset of the work presented in this area, applications include monitoring of bird nesting behavior [2], observations of glacier movement [10], monitoring of volcano activity [5] and analysis of rainforest environments [11].

Nonetheless, while usually large-scale deployments with numerous sensor nodes are expected, most deployments are at a proof-of-concept stage with limited coverage and amount of sensing stations. Likely reasons for this are high cost for system setup and maintenance. In turn, a major part in maintenance is the replacement of depleting energy storage devices, limiting the period of unattended sensor operation.

Energy harvesting has gained more attention, as it can replenish energy reservoirs from ambient energy sources. Types of energy sources cover a broad area, including solar, wind, water flow, vibration, temperature difference and even pH differences in trees [12]–[15]. While the availability of the energy source is highly application dependent, in outdoor environments (i.e., the typical deployment location for environmental monitoring applications) solar energy is almost ubiquitous.

Existing solar energy harvesting systems are presented amongst others in [16]–[19]. Distinction exists between implemented storage devices, charge circuitry and system management (i.e., mainly hardware vs. software control). Typical energy storage devices include Nickel-Metal Hydride (NiMH) batteries, Lithium-Ion (Li-Ion) batteries and Electrochemical Double Layer Capacitors (EDLC), coming each with their advantages and disadvantages.

Systems embedding NiMH batteries include the ones introduced in [17], [20], [21], while Li-Ion based systems are presented in [19], [22]. Due to the limitation of charge cycles, several solutions resulted, combining rechargeable batteries with EDLCs (also known as supercapacitors or ultracapacitors), leading to extended system lifetime [16], [23]. A set of systems, purely relying on Electrochemical Double Layer Capacitors as their storage type, is demonstrated in [1], [18].

Relating to energy extraction, a topic of discussion is Maximum Power Point Tracking (MPPT) for the solar panel. Reference [24] provides a broad overview of different MPPT techniques, whereas not all of them are applicable in micro-solar energy harvesting systems due to energy overhead concerns. Most often used are methods such as *perturb-and-observe*, *hill-climbing*, as well as *fractional open-circuit voltage* and *fractional short-circuit current*.

Micro-solar energy harvesting systems comprehending MPPT techniques are those in [18], [21], [25]. However, some of the methods used yield only limited performance. Arguments against the use of Maximum Power Point Trackers, especially in applications with low power output, are raised in [26].

Work concerning low irradiation conditions is very limited. Reference [11] mentions problems regarding restricted solar availability due to shading in their deployment. Moreover, in [26] limited power income is a major design consideration, but the application addressed is indoors and therefore energy availability is more predictable and constant.

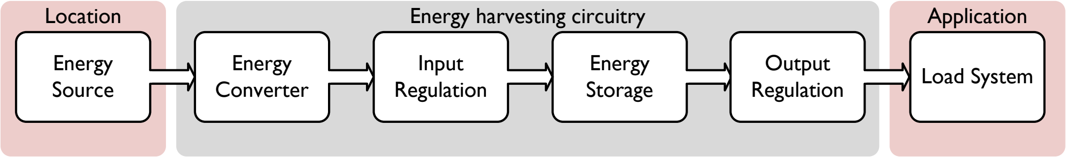


Figure 1. Modular structure of a general energy harvesting system

III. Theory

The underlying architecture of micro solar energy harvesting systems contains modules for energy conversion, energy storage, as well as energy management. While the existence of these modules is conventional, the way of implementation and necessity of additional circuitry varies between systems. Typical differentiation is made between storage types, charge circuitry, energy conditioning, as well as general system complexity. These different design considerations are mainly based on application and location constraints, i.e., expected conversion and consumption rates.

In this section, we will analyze the general architecture of micro solar energy harvesting systems and introduce application and location limitations in our system scenario. Based on these design constraints, we will suggest two harvester architectures.

A. Solar Energy Harvesting

Solar energy harvesting is one of the most common ways of employing ambient energy sources, supporting or replacing battery power supplies in distributed sensor networks. Figure 1 depicts the typical modular structure of an energy harvesting system. While the ambient energy source itself and the load system can be considered as external modules, both have a strong influence on system operation. In turn, energy source availability is depending highly on the system location, as well as load system demands are based on the application. As these factors have considerable impact on the system performance, a more detailed analysis will follow.

The energy harvesting circuitry itself acts as an intermediate module between energy source and load. It contains a conversion module, an energy buffer, as well as typically some sort of input and output regulation.

1) Solar Energy Conversion: Solar cells are used to convert sunlight into direct electrical current, using the photovoltaic effect. In micro solar energy harvesting for distributed sensor systems, size and cost are typical constraining factors. Depending on load system consumption, number of nodes and deployment location, typical solar panels in use rate between hundreds of milliwatt and a few watt.

The output current of a photovoltaic cell is mainly dependent on its terminal voltage and the light intensity, irradiating the cell. This relation is typically described with a solar panel's IV-curve, such as depicted in Figure 2a. The higher the quality of the solar panel, the more its IV-curve will match a rectangle. This is described in the fill-factor of a solar panel, describing its maximum performance in relation to its theoretical maximum performance. The fill-factor is defined as

$$FF = \frac{P_{mpp}}{V_{oc} \cdot I_{sc}}\,, \tag{1}$$

with P_{mpp} being the maximum extractable power, V_{oc} the solar panel's open-circuit voltage and I_{sc} its short-circuit current. The operating point of maximum extracted power is the solar panel's maximum power point (MPP). However, the maximum power point will change with varying irradiance levels, thus being an irradiance dependent maximum power point function. The maximum power points of the solar

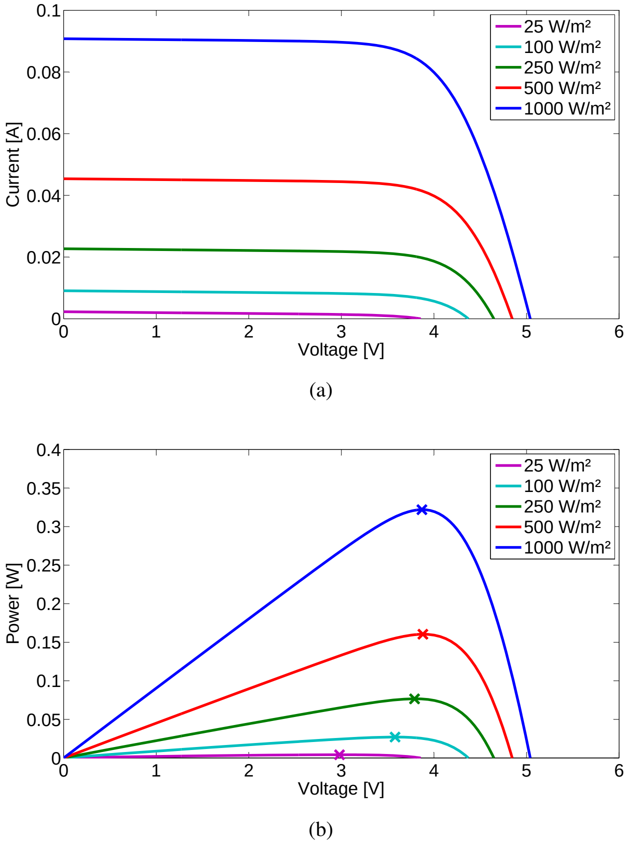


Figure 2. Typical relationships of (a) – current and voltage (IV-curve) and (b) – power and voltage (PV-curve) of a small scale solar panel under different irradiance levels

Table II
OVERVIEW OF MAIN CHARACTERISTICS FOR DIFFERENT STORAGE TYPES; BASED ON [28]

Type	Voltage [V]	Energy Density [W h kg^{-1}]	Self-discharge [%/month]	Cycles [#]	Toxicity
Lead-Acid	2	30-50	5	200-300	high
NiCd	1.2	40-80	20	1500	high
NiMH	1.2	60-120	30	500	low
Li-Ion	3.6	100-150	< 10	1000	low
DLC[1]	2.5-5.5	1-5	several/day	>500000	low

panel underlying Figure 2a are marked in its PV-curves in Figure 2b.

2) Energy Storage: In systems, where the load should be supplied continuously, an energy buffer is necessary. This is, to supply the load from a reservoir at times of insufficient energy income from the ambient energy source. For solar energy harvesting, this typically occurs in a daily cycle. However, impact of the daily cycle itself can be depending on an additional seasonal cycle. The desire is, what in [27] is called energy-neutral operation. At any moment in time, available energy should be greater or equal to the required energy for supplying the load. That is,

$$P_{solar}(t) + P_{store}(t) \geq P_{load}(t) + P_{loss}(t)\,, \qquad (2)$$

where P_{solar} is the power extracted from the solar panel, P_{store} the extractable power from the energy storage, P_{load} the load power consumption and P_{loss} represents storage and conversion losses.

Different types of storage elements have been implemented, with the most common choice being rechargeable batteries. However, alternatively also electrochemical double layer capacitors are used. An overview of properties of typically used technologies is provided in Table II. While rechargeable batteries offer higher energy densities and lower self-discharge rates, leading to be more suitable for long-term storage applications, their overall lifetime and number of recharge cycles is strictly limited. On the other hand, DLCs have long lifetimes and can be charged easily and fast, though their low energy density and high leakage circumvent long-term storage.

3) Input Regulation: The input regulation module usually fulfills two tasks in solar energy harvesting. On the one hand, it adjusts the energy input to meet requirements for further use. On the other hand, it allows to alter the operating point of the solar panel, to extract maximum power.

While input adjustment mainly depends on output levels of the solar panel and the respective storage technology in use, it can be found to some extend in almost all system architectures. Typically implemented functions include reverse-current protection, charge management for the storage device, as well as voltage level adjustments. Opposed to this, Maximum Power Point Tracking (MPPT) is an optional function and its effectiveness in micro solar energy harvesters is not always clear. This is, due to the rather high energy consumption of the tracking solution itself compared to the efficiency gain it will lead to. As the energy consumption of the tracker does not scale down well, systems with low energy harvesting might reduce efficiency when using MPPT [25].

4) Output Regulation: As opposed to the input regulation, the output regulation usually only provides one function which is the adjustment of the harvester's output voltage to an appropriate level for the attached load system. The necessity and the form of implementation purely depends on the energy storage device used in the harvesting system. While systems based on Li-Ion batteries typically do not require voltage adjustments, most of the other storage implementations do. In the majority of cases a step-up regulator which boosts the voltage of the energy buffer is involved. In some cases, implementation of these regulators can be avoided by raising the storage voltage due to the series connection of several storage cells. However, this, in turn, will lead to an increase in both cost and size of the harvester.

The boost operation of the step-up regulator can be formulated as

$$V_{out} = V_{bat} \cdot \frac{I_{bat}}{I_{out}} \cdot \eta\,, \qquad (3)$$

where V_{out}, V_{bat}, I_{out} and I_{bat} are the voltages and currents of the battery and the regulator respectively, and η is the conversion efficiency which typically is a function of the previous parameters. Important dimensioning factors for the use in these applications are, on the one hand, the power consumption of the regulator itself and, on the other, the conversion efficiency. As the latter can vary significantly, a regulator with high efficiencies for the expected input and output parameters should be selected.

B. Application Considerations

As mentioned previously, the application parameters have a considerable influence on the energy harvester design. This is, because energy supply from the ambient source which is rate-limited and the consumption of the load have to match to guarantee continuous operation. As the application determines the sensor nodes tasks and thus their energy demands, the application has to be considered in the design of the harvesting system.

The typical applications that are in focus of this work involve Environmental Monitoring Wireless Sensor Networks. In particular, we consider applications which gather data from a large-scale area in a time-driven manner. As a result of this, the sensor nodes in the network follow a periodical work scheme which is determined by the desired sampling rate of the gathering application. Therefore, the workload of the sensor nodes is predictable which allows for relatively accurate estimation of their power consumption.

Figure 3 depicts a typical network organization in these types of applications. The architecture is organized hierarchically in a cluster-star topology. This leads to a great number of simple sensor nodes, while only a smaller subset of sensor nodes is involved in a multi-hop backbone network. These clusterheads, in turn, are usually equipped with more resources to balance their increased workload. In addition, the backbone network is connected via a gateway node to a server which stores the collected data and allows for remote access to the network. As there is usually a distance between the deployment site and operator, the communication link between the gateway and server involves remote communication technology (i.e., typically long-range RF, satellite or GSM/GPRS).

Because of the large number of sensor nodes that are expected in these types of applications, maintenance of each individual sensor node in the network is not feasible. The sensor node lifetime should thus be as long as possible to allow for extended data collection periods. Based on the periodical workload of the sensor nodes, duty-cycling is an efficient way to reduce their overall power consumption. The sensor nodes thus follow a defined schedule of active and inactive periods which enables for estimation of their average power consumption according to

$$P_{avg} = \delta \cdot P_{active} + (1 - \delta) \cdot P_{inactive} \,, \qquad (4)$$

$$0 < \delta < 1 \,. \qquad (5)$$

In this case δ is the duty-cycle rate and P_{avg}, P_{active} and $P_{inactive}$ are the respective power levels in average, in active state and in inactive state.

Additionally, equation 4 can be broken down to power levels and time intervals which are typically involved, so that

$$P_{avg} = \frac{P_p \cdot t_p + P_c \cdot t_c + P_s \cdot t_s + P_i \cdot t_i}{T_{sample}} \,, \qquad (6)$$

where P_p, P_c, P_s, P_i, t_p, t_c, t_s and t_i are power levels and time intervals for processing, communicating, sensing and when inactive respectively, and T_{sample} is the sample interval of the sensor nodes.

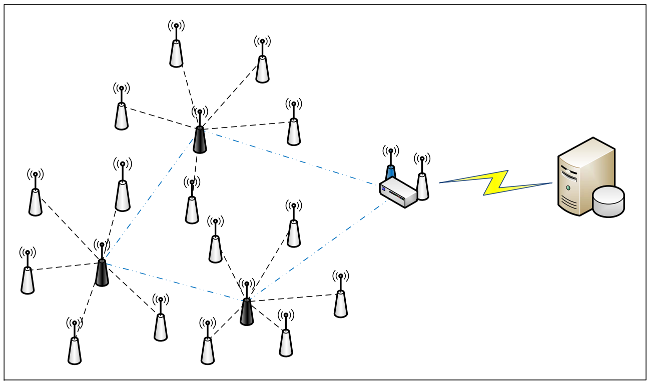


Figure 3. Possible network architecture of data gathering applications in environmental wireless sensor networks under scope

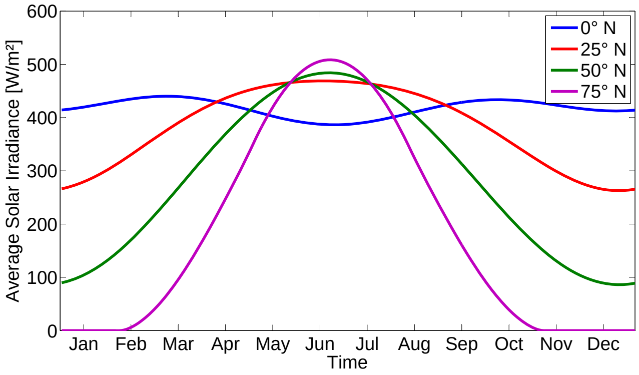


Figure 4. Overview on estimated solar irradiation on top of earth atmosphere at different latitudes (data obtained from [29])

Based on the, usually in environmental monitoring found, low sampling rate and the low power consumption in the idle state of the system, the resulting average power consumption is also low.

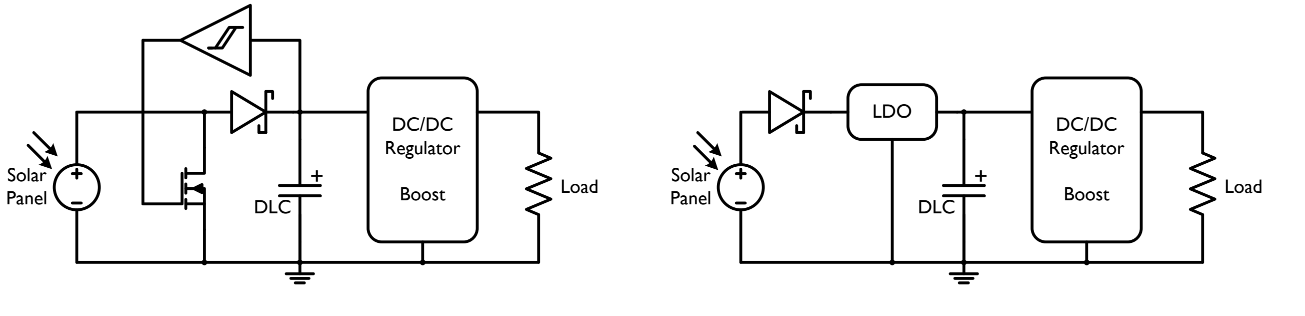
C. Location Dependency

The second external parameter which influences the harvester operation is the location the final system is deployed in. This is mainly because of varying availability constraints of incoming solar irradiation with changing deployment location. Figure 4 shows changes in estimated solar irradiation over the year at different latitudes in the northern hemisphere. Moving north from the equator, two observations can be made which have to be considered in the usage of ambient energy sources at different locations. These observations are

1) With increasing latitude, one typically has to deal with a decrease in solar intensity (e.g., a decrease in average yearly solar radiation).
2) With increasing latitude, variation of solar irradiation during the year increases which leads to periods of high and low solar radiation.

While the first constraint alone does not pose such a big problem, the combination with the latter constraint is what requires an additional design consideration. Certainly, a reduced average solar radiation leads to a lower supply rate from the ambient energy source, which limits the permitted energy demands of the load system. However, for low power systems, such as Wireless Sensor Nodes, this often is not an issue. Furthermore, the reduction in the supply rate could easily be compensated with an increased solar panel.

As opposed to that, the variable solar irradiation over the year forms some limitations the harvesting system has to deal with. As there are time intervals with high solar radiation, as well as intervals with low solar radiation, ideally an energy balancing is desired. Because of the rather long timescales involved, this balancing requires long-term



(a)

(b)

Figure 5. Simplified circuit diagrams of the suggested solar energy harvesting architectures - (a) direct input coupling and (b) LDO input regulation

storage of energy and requires, thus, typically battery storage technology. In contrary, if energy balancing is not an option, the system design is determined by the period of lowest energy income. This means, the harvesting unit is designed for the worst case scenario of the deployment location.

In addition to this *global* location dependency, a *local* location dependency can influence the harvesting behavior and outcome. Typical influences are due to obstacles which change the intensity and direction of incoming solar irradiation. The amount of influence in these situations is difficult to predict, but the influence might be classified into generally open or generally shaded locations. Nonetheless, the effect of these obstacles usually is an overall reduction of solar income or short-term variations, as opposed to long-term variations of global location dependency.

D. Suggested Architectures

In this work, we mainly target systems that should be capable of operating from solar energy even at locations with limited solar radiation, as described previously. Furthermore, the one main design goal is the durable and continuous energy supply to the load system. Thus, the two solar energy harvesting architectures presented, are built upon a Double Layer Capacitor (DLC) energy buffer which allows long system lifetime, but provides only short-term energy storage in the order of days.

The two suggested architectures are presented in simplified form in Figure 5. While the basic structure of the two systems is the same, there is a difference in the implementation of the input regulation. Because of the usage of DLCs as the harvesters' energy buffer and the resulting low energy storage capacity, these harvesting systems are vulnerable to short-term variations in irradiation conditions. This means that the systems have reduced capability of balancing changes in available ambient energy, compared to systems using battery buffer technologies. Thus, these architectures must work sufficiently with the available energy income at any time, except of short bridging periods covering e.g. nights.

Based on this, the system is designed for the worst-case scenario of solar irradiance during the year. As the available energy at these irradiance levels is very limited, harvester simplicity is the key to the successful operation of the system. The smaller the amount of available energy is, the more important becomes the own power consumption of the harvesting module (further referred to as the harvester's energy overhead). For continuous operation, the available energy E_{in} has to be large enough to supply the load system at any moment in time, such as

$$E_{in} \geq E_{oh} + E_{load}\,, \tag{7}$$

where E_{oh} is the energy overhead of the harvester and E_{load} the energy demand of the load. Keeping E_{oh} low allows to supply the load with less E_{in}. It should be noted, that while the load consumption in this application typically is an average consumption resulting from duty-cycled sensor node operation, the overhead consumption occurs continuously.

In addition to the common storage technology, the two presented systems use the same output regulation module. This module consists of a DC-DC regulator of boost topology. As DLCs typically have a rather low nominal voltage and this voltage further decreases tremendously with the discharge of the capacitor, for most sensor nodes a voltage level adjustment is required. For implementation a Texas Instruments TPS61070 was chosen, because it has a low power consumption and offers high conversion efficiencies.

The difference between the two architectures lies in their input regulation. Figure 5a depicts an architecture with direct coupling of solar panel and energy buffer, while in the architecture shown in Figure 5b a Linear Dropout Regulator (LDO) is integrated for input regulation.

Direct coupling of solar panel and storage element means that, on the one hand, the solar panel operating point is determined by the charge state of the DLC, on the other, the charge voltage of the energy buffer is only limited by the open-circuit voltage of the solar panel at any time. In result, the power output of the solar panel depends on the current charge state of the storage element. With a double layer capacitor of the type implemented [30], these operating points will typically be between 1 V and 2.5 V according to Figure 2b. Additionally, due to risk of performance reduction or damage, the DLC has to be protected from over-charging. Over-charge occurs when the capacitor voltage exceeds its nominal voltage which can result in reduced lifetime and eventual destruction [31]. A typical way of over-voltage protection is the introduction of a Zener diode. However, as Zener diodes do not have ideal behavior, losses around the breakdown voltage are immense. The typical behavior of a Zener diode is depicted against the ideal behavior in Figure 6. As it can be seen, with this choice, harvesting losses of tens of milliampere close to the breakdown voltage have to be accepted. As this is an intolerable level in most situations, the over-voltage protection in the suggested architecture consists of a combination of hysteresis comparator and MOSFET. This combination replaces the Zener diode by implementing an almost ideal Zener diode behavior. The comparator observes the DLC voltage and triggers the MOSFET to disconnect the solar panel from the energy storage device once the nominal voltage is reached. In this way losses below the breakdown voltage are limited to the operating consumption of the comparator, while all energy is diverted from the DLC as soon as the breakdown voltage is reached. Implementing this protection circuit with a low-power hysteresis comparator, such as a Maxim MAX9017, these losses are limited to a few microampere.

In the second architecture, an input regulation based on an LDO regulator is implemented. As opposed to direct coupling, this means that solar panel and double layer capacitor are only indirectly connected with each other. The implementation of this regulator comes with mainly two advantages for the harvester. Firstly, the regulator makes an over-voltage protection mechanism obsolete, and secondly, ambient energy availability periods are used more efficiently. The former originates in the regulated output of the LDO regulator which, as long as its input is high enough, provides a constant, predefined voltage at the output. In this case, this constant voltage should be chosen in accordance to the nominal voltage of the double layer capacitor. This means that, the DLC will be charged to its nominal voltage only, but never higher. Additionally, the regulator will hold the capacitor at its nominal voltage as long as the input to the LDO allows this. This results in the second advantage of this architecture, because the DLC just begins discharging when ambient energy availability decreases to an insufficient level. As opposed to that, the directly coupled architecture involves a second charge/discharge condition, which is caused by the hysteresis band of the comparator. However, the implementation choice of regulators in this architecture are limited, based on the internal structure of LDO regulators. The challenge is, that most LDO regulators do not permit the voltage level at the output to be considerably higher than at the input. Because of the energy storage element at the output and the intermittent energy source at the input, however, this is a common situation in energy harvesting applications. This restriction limits the choice of appropriate regulators tremendously, especially when it comes to power consumption constraints. We decided on a parallel structure of two Texas Instruments TPS71525.

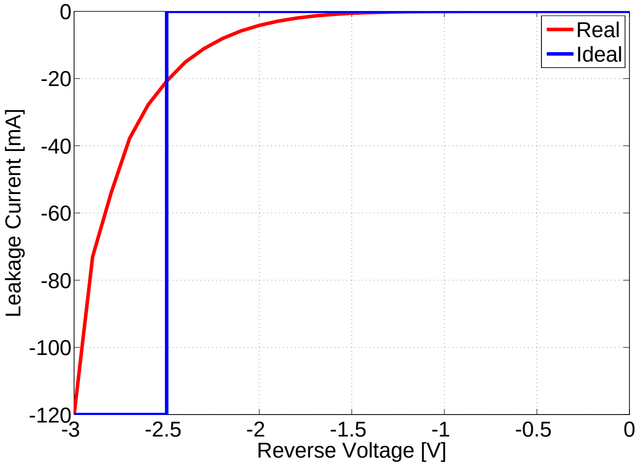


Figure 6. Comparison of ideal and measured current-voltage characteristic of a Zener Diode in reverse connection

In both systems maximum-power-point-tracking is avoided, as the implied additional energy overhead is too high for the relatively low amounts of additionally gained energy extraction in low-irradiance conditions, such as presented in [26].

IV. Experimental Setup

The evaluation of the architectures is divided into two parts. Firstly, single components and modules of the architectures are analyzed and evaluated in a laboratory environment which describes their behavior and supports the implementation process. Additionally to these measurements, an outdoor deployment of the final architecture implementations is conducted to verify the system behavior in its real application environment.

A. *Laboratory Measurements*

In the laboratory measurements, the single components and also combinations of modules are analyzed to determine the behavior of these modules in the final system. Most of the measurements have been performed on the double layer capacitor (i.e., the storage module), as this is the module with most implementation flexibility. The experiments cover analysis of the capacitor's energy storage capability, the influence of its ESR, and the behavior in serial connection of two DLCs.

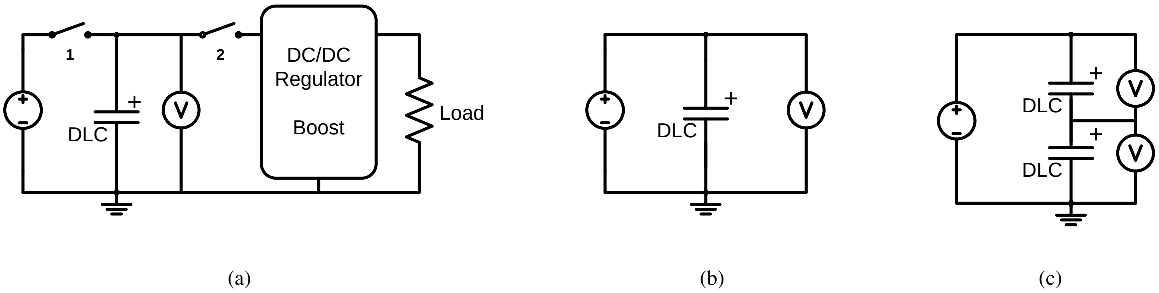


Figure 7. Measurement setups of double layer capacitor experiments - (a) Energy storage time analysis; (b) Charge cycle measurement; (c) Evaluation of series connection

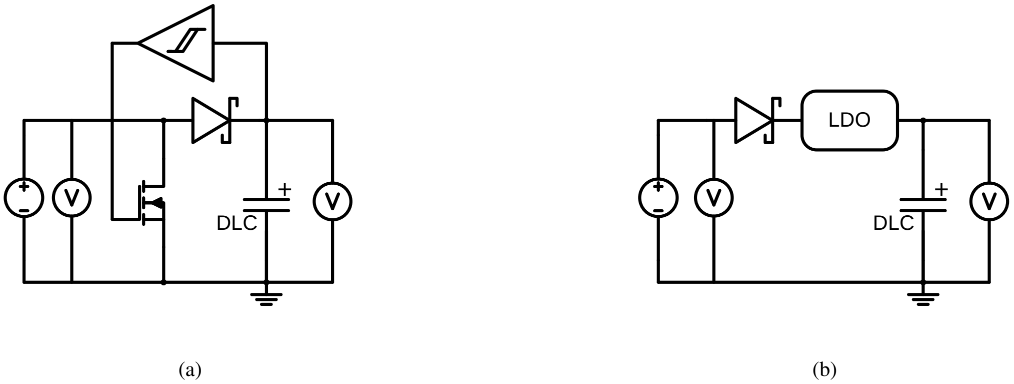


Figure 8. Measurement setups for charge mechanisms - (a) Over-voltage protection measurement; (b) Charge behavior of LDO-based architecture

Additionally to these double layer capacitor tests, evaluation of the charging mechanisms for both architectures have been conducted. This means, for the directly coupled architecture, the over-voltage protection mechanism has been validated, whereas for the LDO-based architecture the whole charging process is evaluated.

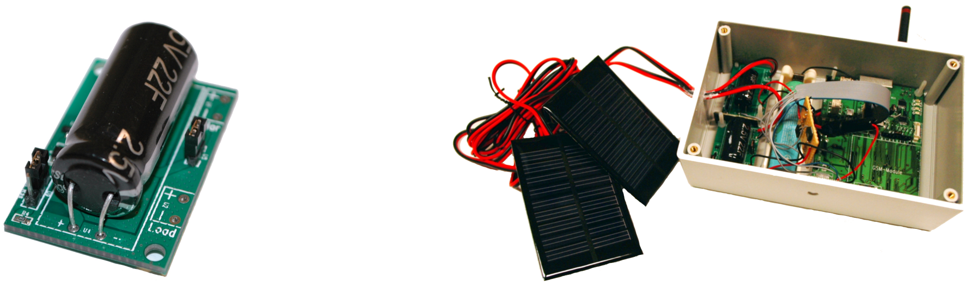
The measurement setups are depicted for the capacitor and charge management experiments in Figures 7 and 8 respectively. As DC source in all setups a Hameg HM8143 Programmable Power Supply is used. Furthermore, for voltage measurements in setups 7a and 7b we used an Agilent 34410A Digital Multimeter, whereas a National Instruments NI USB-6008 data acquisition tool is integrated for measurements in 7c, 8a and 8b.

In order to compare the energy storage capabilities of the double layer capacitors, we use a setup according to Figure 7a. The boost-regulator in this setup is a Texas Instruments TPS61070 and DLCs with different capacities from Cooper-Bussmann [30] are implemented as storage device. As system load we employ a typical bi-modal consumer, which is programmed on a Sentio-e^2 node [32]. Changes in the consumers duty-cycle allows for analysis of different load consumptions. In the beginning of this experiment, the switch number one is closed to charge the capacitor to its nominal voltage, while the load is disconnected. Once the DLC is fully charged, the source is disconnected, whereas the load is connected. The voltage level during discharge and the discharge time is logged.

With the measurement setup depicted in Figure 7b, the charging cycle of the double layer capacitor is analyzed. In particular we evaluate the impact of different equivalent series resistances (ESR) in this experiment. Thus, we implement two DLCs of same capacity, but different voltage and ESR ratings. The Hameg HM8143 is used for charging at constant rate and its integrated electronic dummy load (EDL) allows for controlled discharge.

The last DLC experiment conducted, evaluates the behavior of double layer capacitors in serial connection and its setup is shown in Figure 7c. For the measurement two double layer capacitors of same type and capacity are connected in series and collectively charged to the double nominal voltage. Once charged, they are discharged over the EDL of the Hameg HM8143. This procedure is repeated several times, while the individual voltages of the capacitors are logged.

Figure 8a illustrates the experimental setup of verifying the over-voltage protection mechanism. The protection circuit consists of a MAX9017 hysteresis comparator with internal voltage reference and a ON Semiconductor N-channel MOSFET [33]. In the measurement, the double layer



(a)

(b)

Figure 9. Implementation of the deployment system - (a) Direct coupling harvester circuit board; (b) Complete system integration

capacitor is charged with a DC source of higher voltage than the capacitors nominal voltage, while source voltage and DLC voltage are monitored.

The LDO-based charge mechanism is analyzed with the setup depicted in Figure 8b. The LDO used in this setup is a Texas Instruments TPS71525. Furthermore we use a Fairshild Semiconductor Shottky Diode [34], which protects the source from reverse currents. During charging in this experiment voltages at the input and output are logged.

B. Deployment Evaluation

After the evaluation of single modules, the full architectures have been implemented and deployed in an outdoor environment. The architecture implementation occurred according to Figure 5. For the solar panels a commercially available 4.5 V-100 mA type was chosen, because this panel will provide voltages, high enough to fully charge a 2.5 V DLC, even under low irradiation conditions. The physical size of this panel is 94 × 61 mm^2, thus comparable to dimensions chosen in other systems. As storage element DLCs with capacities of 10 F and 22 F have been chosen.

The load system is implemented with a Sentio-e^2 node platform [32]. This node is designed especially for environmental monitoring applications in mind. It is based on a Texas Instruments MSP430 microcontroller and a CC1101 low-power radio transceiver operating in the 433 MHz ISM-band. The platform consumes less than 7 µA (19.6 µW) in low-power mode with operating timers, which makes it highly suitable for low duty-cycling operations, such as in environmental monitoring. The node operates on a synchronous TDMA communication protocol, which efficiently reduces active time to a minimum, thus, further reducing energy consumption. The load is set to a bi-modal consumption with an average current draw of 20 µA, which approximately equals one packet transmission per minute.

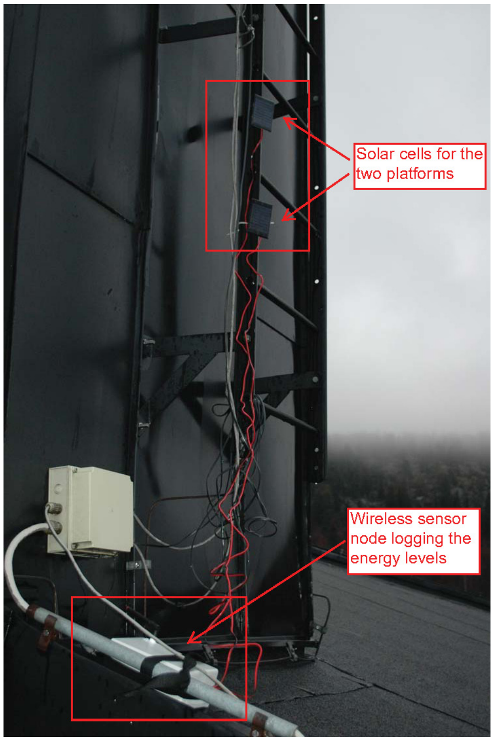

Solar cells for the two platforms
Wireless sensor node logging the energy levels

Figure 10. Picture of the deployment setup of a solar harvesting sensor node at the Mid Sweden University campus in Sundsvall, Sweden

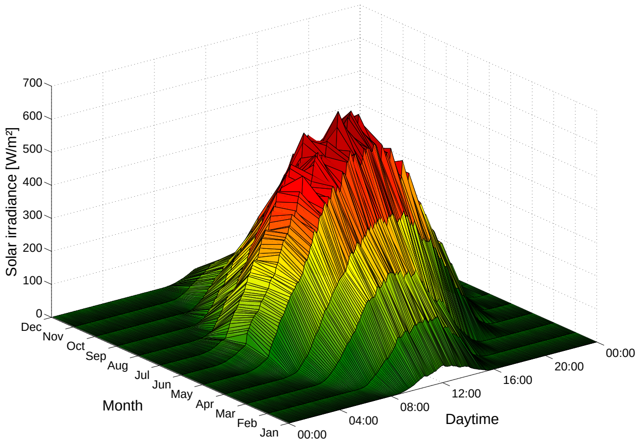


Figure 11. A typical annual insolation profile in Sundsvall, Sweden

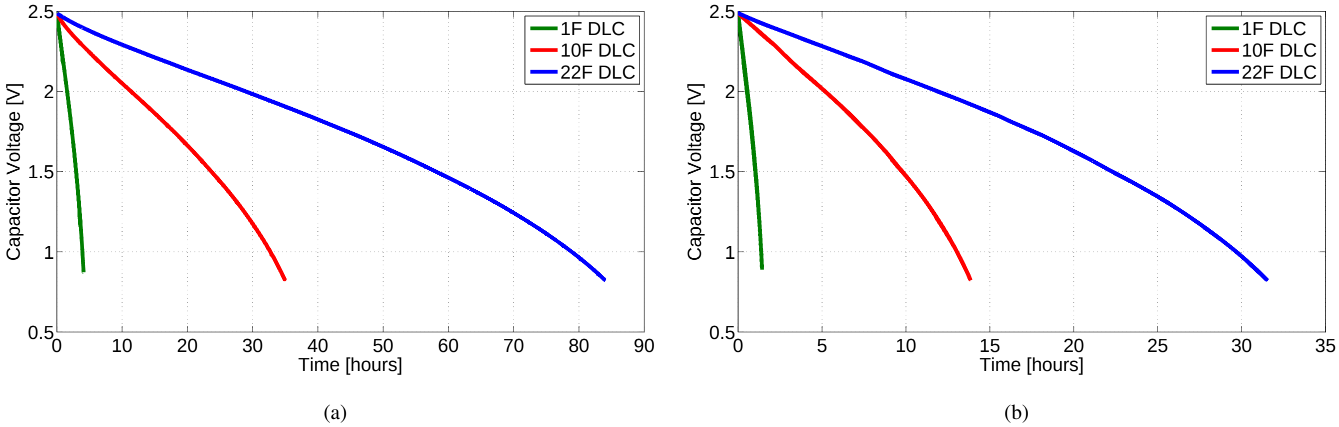


Figure 12. Maximal supply time of fully charged double layer capacitor under bi-modal load - (a) for 20uA average load current; (b) for 100uA average load current

During the deployment, the node system has the function of monitoring the implemented architectures, and additionally measures environmental parameters. While the voltage measurement of the double layer capacitor is performed by the microcontroller's internal ADC, the platform is further equipped with temperature, humidity and solar radiation sensors. For temperature and humidity a Sensirion SHT15 is chosen, whereas a Davis 7821 carries out the solar radiation measurements. The sampling rate of the sensors are 5 min and a Li-Ion battery is provided to allow data collection during times of harvester malfunction.

Figure 9 shows a picture of the implemented system. In 9a the harvesting circuit of the directly coupled architecture is shown, while 9b depicts the complete, deployable system.

The systems have been deployed at the Mid Sweden University campus in Sundsvall, Sweden $(62°24'N, 17°19'E)$. A picture of the deployment setup of one of the deployed sensor nodes is given in Figure 10. The deployment location was chosen to be on a building's roof, to avoid obstacles blocking the insolation to the solar panels. The biggest challenge at this location is the great distance from the equator, which leads to a lower solar radiation and large variations in insolation over the year. A typical solar insolation profile of this location is given in Figure 11, which clearly indicates the short daylight period and low irradiance levels in the winter month. These worst case energy levels will define the maximal load, the harvester can supply at this location. To investigate the system operation under these conditions, the deployment period lasted from November 2009 to January 2010.

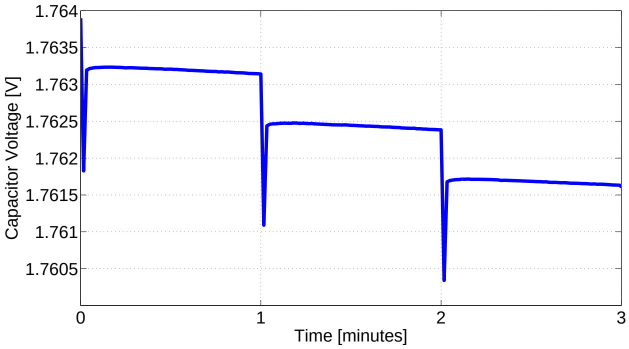


Figure 13. Close-up of the discharge behavior in Figure 12

V. Results

Figure 12 shows the results for the first laboratory measurement, depicted in Figure 7a. The graphs show the discharge from fully charged double layer capacitors of various capacities. The load for this discharge is bi-modal and in Figure 12a the average load current is 20 µA, while the average load current in Figure 12b is 100 µA. The discharge has been continued until the voltage was too low to keep the output regulator operating. The discharge time is a good indicator for how long a dark-period is allowed to be, without the system failing. It is noticeable that the 1 F double layer capacitor lets the output regulator stop at a higher input voltage than the 10 F or 22 F capacitor. This is due to the limited energy stored in the capacitor and the bi-modal load. While in Figure 12 the discharge curves appear constant, the influence of the bi-modal load can be observed in a close-up, shown in Figure 13. The high current peaks, occurring periodically every minute, result in a voltage drop depending on the ESR of the respective DLC. This voltage breakdown relaxes after the current pulse is over. However, the low voltage level for a short moment in time, might stop the output regulator. As the capacitor with smaller capacity has a higher ESR, also the voltage drops are higher.

The ESR of the double layer capacitors also influence the charge-discharge behavior of the device. The measurement of two double layer capacitors of same capacity (i.e., in

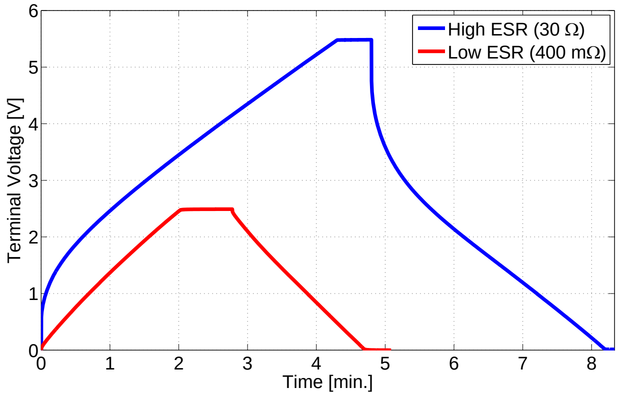


Figure 14. Effect of the ESR on the charge-discharge behavior of DLCs

this case 1 F), but different nominal voltages and ESRs, is depicted in Figure 14.

Figure 15 shows the results of using two double layer capacitors in a serial connection. In the graph it can be observed that charging and discharging does not occur at the exact same rate, even for capacitors of same type. This results into unequal distribution of voltage for the two DLCs. As there is no balancing between the two double layer capacitors, the resulting voltage difference will increase over time. In turn, this means that although the charging voltage might not exceed twice the nominal capacitor voltage, a single capacitor can be charged higher than expected and thus be damaged. Therefore, even with a series connection of capacitors, over-voltage protection is needed.

The results of evaluating the presented over-voltage protection mechanism, presented previously, is depicted in Figure 16. As it can be seen in the graph, the power supply is disconnected from the storage capacitor when the capacitor voltage reaches its nominal voltage. While the power supply is then shorted, the double layer capacitor is discharged until its voltage will cross the lower hysteresis level, connecting the power supply back to the DLC.

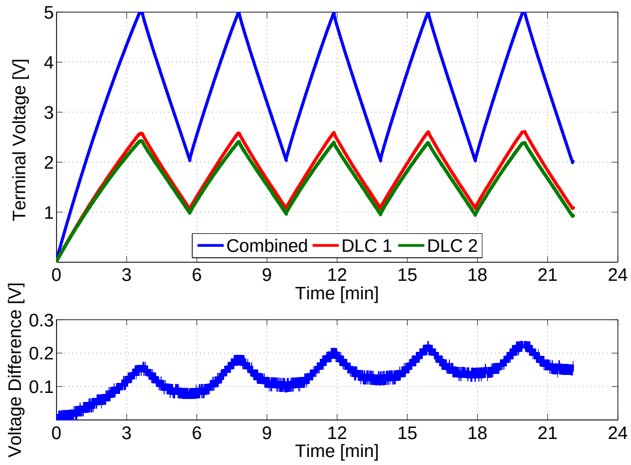


Figure 15. Measurement of charge-discharge behavior of double layer capacitors in a serial connection

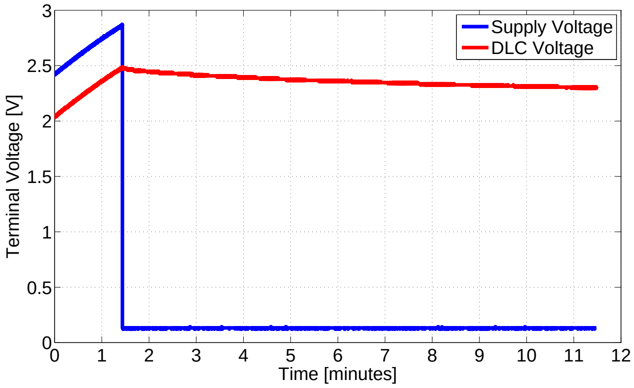


Figure 16. Evaluation of the over-voltage protection mechanism in the directly-coupled harvesting architecture

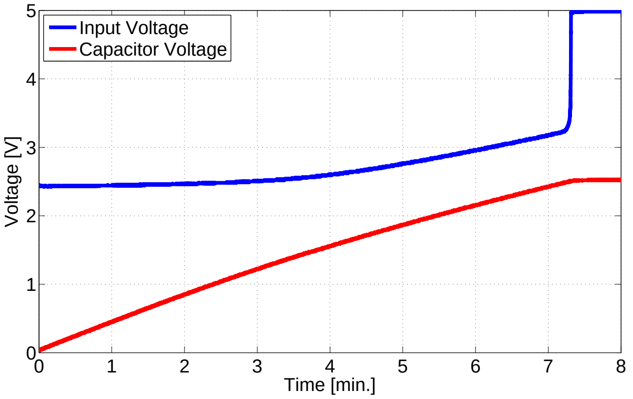
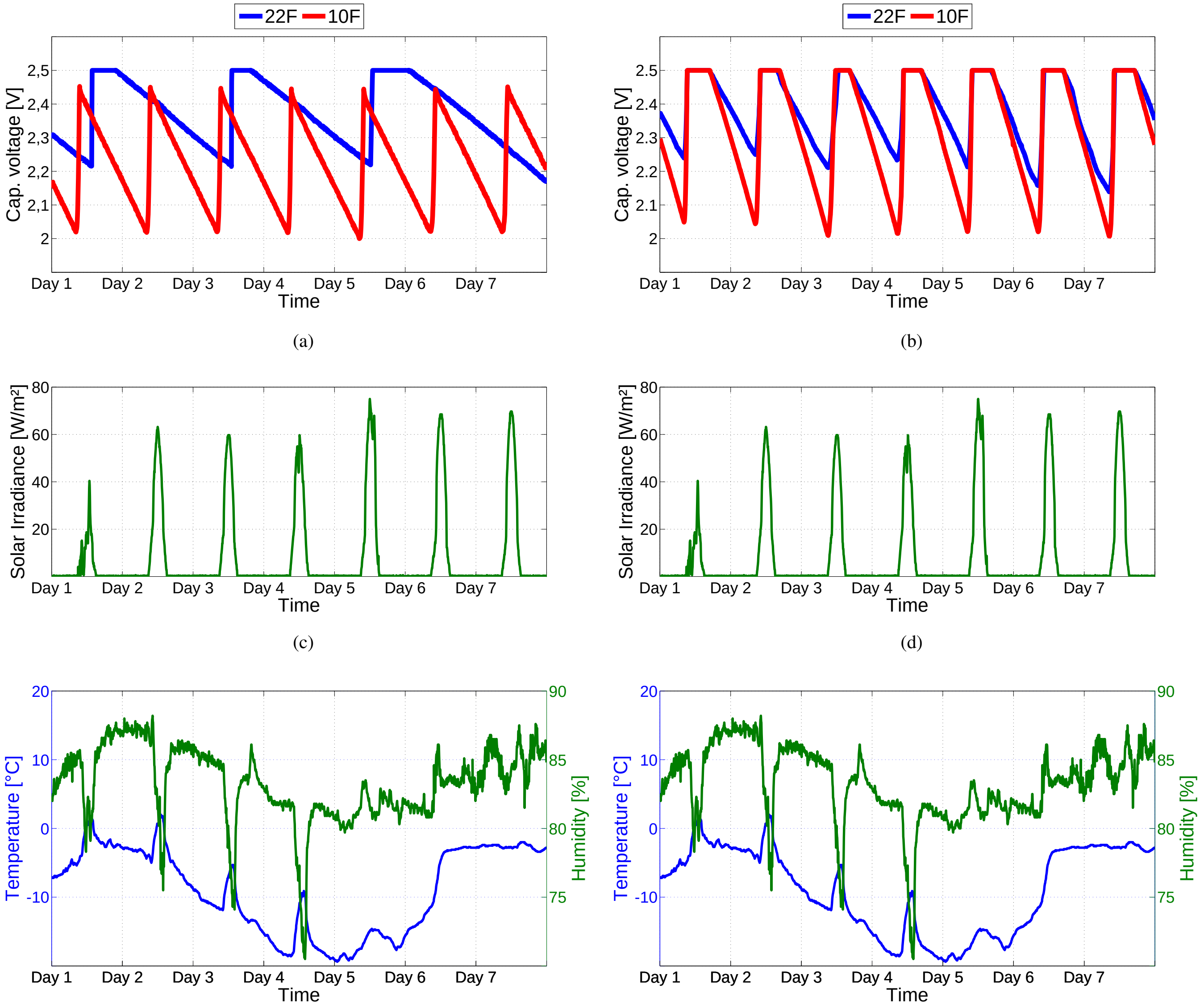


Figure 17. Charge behavior with the LDO regulated harvesting architecture

In Figure 17, the analysis of the charge behavior in the LDO-based harvesting architecture is shown. It can be observed, that the input voltage is linked to the output voltage of the regulator. While charging the capacitor the input voltage is pulled down to a voltage close to the regulator's output voltage. Only when the DLC is fully charged and does not draw a current from the source any longer, this relation releases and the input voltage raises to its defined level. This behavior is expected to be based on the internal architecture of the used linear dropout regulator. For the solar energy harvesting architecture this means, that the operating point of the solar panel will be between 2.5 V and 3 V, which provides higher power output then the coupled operating point in the directly-coupled architecture.

Finally, results from the fully implemented architectures, obtained in the outdoor deployment, are given in Figure 18. The graphs show one week of data in the deployment period



(a)

(b)

(c)

(d)

(e)

(f)

Figure 18. System behavior of two solar energy harvesting architectures during one week of deployment – (a),(b) capacitor voltage levels for directly-coupled architecture (left column) and LDO-based architecture (right column); (c),(d) irradiance conditions during the week; (e),(f) temperature and humidity during the week

and include the DLC voltages for both architectures with two different capacity sizes respectively. Additionally, the temperature, humidity and the solar irradiance during the period are provided as reference. As visible in Figures 18a and 18b, the voltage levels in the double layer capacitors follow the daily insolation variations. An exception to this is the directly-coupled architecture, which uses a 22 F DLC. The reason for this is the hysteresis of the over-voltage protection mechanism in this architecture.

While the LDO-based architecture only has one charging condition (i.e., the current insolation), the hysteresis band of the comparator in the directly-coupled architecture adds a second charging condition. This means, the DLC in the directly-coupled architecture does only charge when two conditions are fulfilled. (i) the solar irradiance level is high enough to lead to charge the capacitor and (ii) the lower hysteresis level has been crossed at least once since the last crossing of the higher hysteresis level.

Furthermore, the over-voltage protection leads to a discharge of the double layer capacitor as soon as the comparator voltage (i.e., ideally the nominal voltage) has been crossed. This is independent of external parameters, which means that even when enough solar energy is available, the stored energy will reduce. As opposed to this, the LDO-based architecture charges the storage and holds the DLC at full charge level as long as there is sufficient ambient energy.

In addition, one can observe external influences on the analog over-voltage protection in Figure 18a. Although hysteresis levels have been set to fixed levels, these levels vary depending on the device and external conditions. It is for example clearly visible, that the higher hysteresis level (i.e., the over-voltage trigger level) differs from its preset value. For the $10\,\mathrm{F}$ DLC a trigger voltage of $2.45\,\mathrm{V}$ was measured and for the $22\,\mathrm{F}$ DLC a voltage of $2.55\,\mathrm{V}$, whereas both voltages were set to $2.5\,\mathrm{V}$. Due to a measurement range limitation of the internal ADC of $2.5\,\mathrm{V}$, this is not visible for the $22\,\mathrm{F}$ DLC in the graph.

In Figure 18b the flat tips of the graph do not result from the same measurement range limitation, but occur, because current insolation holds the storage at full charge level. Comparing the discharge rates in Figures 18a and 18b, a higher discharge rate can be observed in the LDO-based harvesting architecture, which results from the higher consumption overhead of this architecture.

VI. Conclusions

Wireless sensor networks offer a number of advantages in the domain of environmental monitoring applications, including the autonomous observation of physical, chemical and biological values at large scale. In addition to area coverage, wireless sensor networks can provide great resolution within this area and deliver their samples to a designated collection point. Additional strength of this technology include the possibility to operate them remotely, which reduces the amount of human invasiveness to the monitored site.

On the other hand, environmental monitoring applications require wireless sensor networks to operate over a long period of time. This results from the typically slow processes being observed and the rather high effort of deploying the network in the environment. In contrast to this requirement, wireless sensor nodes are traditionally powered by batteries and thus have a limited energy resource. Reducing the energy consumption leads to more efficient use of the resource, but cannot alter the fact that the energy capacity is finite.

Energy harvesting provides an alternative supply method, which has the capability to power sensor nodes indefinitely. Solar energy harvesting, as one of these supply methods, offers high availability coverage and generally high energy levels. Furthermore, the conversion technology used in solar energy harvesting is low cost and can easily be scaled to different power level requirements.

However, a problem in solar energy harvesting is the dependency on external factors, particularly the location the final system is deployed in. In this paper we addressed the use of solar energy harvesting at locations with long distance to the equator. These systems have to deal with two main influences. (i) Annual solar radiation decreases with distance to the equator, and (ii) the variation of insolation levels increases.

We suggested two architectures for solar energy harvesting, which address these challenges by their system simplicity. This reduces the energy overhead spent by harvesting, which can easily reach similar magnitudes as the average load consumption of low-duty cycle sensor nodes. We further analyzed the behavior of these architectures in laboratory measurements, which resulted in our implementation choices. After that, the fully implemented architectures have been evaluated in an outdoor deployment in Sundsvall, Sweden ($62°24'N, 17°19'E$) during winter 2009/2010.

The laboratory experiments allowed us to get a feeling of the energy storage capability of double layer capacitors and thus to choose appropriate devices. We could further determine that a serial connection of double layer capacitors can not eliminate the need for an over-voltage protection circuit, even though the input supply voltage is lower than the combined nominal voltages of the used double layer capacitors. Finally, the laboratory measurements also enabled for the verification of the charge mechanisms in both architectures.

Comparing the two architectures, the system deployment showed that both architecture were capable of supplying sufficient power to the load during the whole deployment period. Nevertheless, there are some differences in their operation. While the directly-coupled architecture provides lower energy overhead, and therefore has a slower discharge period, it is affected by the additional charge/discharge condition, introduced by the hysteresis of the over-voltage protection. In addition, the low-power components used in the over-voltage protection module are easily influenced by external factors and thus show strong variations. While it will increase the stress on the double layer capacitor, in future deployments a lower hysteresis band is recommended, which should reduce the impact of these restrictions.

In contrast to this, the LDO-based architecture does not have this additional charge condition, which means charge/discharge behavior is only dependent on available ambient energy at any moment in time. While this utilizes the available energy more effectively, the stress on the double layer capacitor is larger and the overall energy overhead of the architecture is increased. Furthermore, the dimensioning decision for the input regulator, such as the maximum allowed current, limit flexibility of this architecture. This means that one implementation might not be usable for various application conditions (e.g., different solar panel power requirements).

Whereas both architectures come with their individual advantages and disadvantages and proven themselves in the outdoor deployment, once a decision has to be made, we see more advantages in the directly-coupled architecture. While this harvesting architecture shows some limitations due to its second charge condition, the influences of this limitations can be reduced by reducing the hysteresis of its over-voltage protection. Moreover, the system is more flexible to changes in application and location constraints and thus enables for a greater variety of application cases. Finally, its overall lower energy overhead allows for operation during longer dark-periods.

In future work, balancing of annual variations in the ambient energy income is a topic of interest. This can be addressed both, on the hardware and the software level. Furthermore, we will look in more detail into the issue of maximum-power-point-tracking for harvesting from low ambient energy sources.

Acknowledgment

Financial support of the project “Sensible Things that Communicate” by KK-foundation is gratefully acknowledged. Furthermore we would like to thank Torsten Schölzel from Solarwatt AG, Germany, for contributions supporting the work in this paper.

References

[1] S. Bader and B. Oelmann, “Enabling battery-less wireless sensor operation using solar energy harvesting at locations with limited solar radiation,” in *Fourth International Conference on Sensor Technologies and Applications, SENSORCOMM ’10*, 2010.

[2] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, “Wireless sensor networks for habitat monitoring,” *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pp. 88–97, 2002.

[3] J. Porter, P. Arzberger, H.-W. Braun, P. Bryant, S. Gage, T. Hansen, P. Hanson, C.-C. Lin, F.-P. Lin, T. Kratz, W. Michener, S. Shapiro, and T. Williams, “Wireless Sensor Networks for Ecology,” *BioScience*, vol. 55, no. 7, pp. 561–572, 2005.

[4] J. Hart and K. Martinez, “Environmental Sensor Networks: A revolution in the earth system science?” *Earth-Science Reviews*, vol. 78, no. 3-4, pp. 177–191, Oct. 2006.

[5] G. Werner-Allen, J. Johnson, M. Ruiz, J. Lees, and M. Welsh, “Monitoring volcanic eruptions with a wireless sensor network,” in *Proceeedings of the Second European Workshop on Wireless Sensor Networks*, 2005, pp. 108–120.

[6] K. Martinez, P. Padhy, A. Elsaify, G. Zou, A. Riddoch, and JK, “Deploying a sensor network in an extreme environment,” *Sensor Networks, Ubiquitous and Trustworthy Computing, June*, 2006.

[7] T. Le Dinh, W. Hu, P. Sikka, P. Corke, L. Overs, and S. Brosnan, “Design and deployment of a remote robust sensor network: Experiences from an outdoor water quality monitoring network,” in *32nd IEEE Conference on Local Computer Networks, 2007. LCN 2007*, 2007, pp. 799–806.

[8] J. Paradiso and T. Starner, “Energy Scavenging for Mobile and Wireless Electronics,” *IEEE Pervasive Computing*, vol. 4, no. 1, pp. 18–27, Jan. 2005.

[9] “U.s. energy information administration – website,” http://www.eia.doe.gov, (accessed 02-08-2010).

[10] K. Martinez, R. Ong, and J. Hart, “Glacsweb: a sensor network for hostile environments,” *2004 First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks, 2004. IEEE SECON 2004.*, pp. 81–87, 2004.

[11] T. Wark, W. Hu, P. Corke, J. Hodge, A. Keto, B. Mackey, G. Foley, P. Sikka, and M. Brunig, “Springbrook: Challenges in developing a long-term, rainforest wireless sensor network,” *2008 International Conference on Intelligent Sensors, Sensor Networks and Information Processing*, pp. 599–604, 2008.

[12] R. Morais, S. Matos, M. Fernandes, A. Valente, S. Soares, P. Ferreira, and M. Reis, “Sun, wind and water flow as energy supply for small stationary data acquisition platforms,” *Computers and Electronics in Agriculture*, vol. 64, no. 2, pp. 120–132, Dec. 2008.

[13] R. Torah, P. Glynne-Jones, J. Tudor, T. O’Donnell, S. Roy, and S. Beeby, “Self-powered autonomous wireless sensor node using vibration energy harvesting,” *Measurement Science and Technology*, vol. 19, no. 12, October 2008.

[14] C. J. Love, S. Zhang, and A. Mershin, “Source of sustained voltage difference between the xylem of a potted ¡italic¿ficus benjamina¡/italic¿ tree and its soil,” *PLoS ONE*, vol. 3, no. 8, p. e2963, 08 2008.

[15] C. Knight and M. Collins, “Results of a water based thermoelectric energy harvesting device for powering wireless sensor nodes,” in *Proc. SPIE 7288 – Active and Passive Smart Structures and Integrated Systems*, 2009.

[16] X. Jiang, J. Polastre, and D. Culler, “Perpetual environmentally powered sensor networks,” in *Proceedings of the 4th international symposium on Information processing in sensor networks*, 2005, pp. 463–468.

[17] V. Raghunathan, A. Kansal, J. Hsu, J. Friedman, and M. Srivastava, “Design considerations for solar energy harvesting wireless embedded systems,” in *Proceedings of the 4th international symposium on Information processing in sensor networks*, 2005, pp. 457 – 462.

[18] F. Simjee and P. Chou, “Everlast: long-life, supercapacitor-operated wireless sensor node,” in *Proceedings of the 2006 international symposium on Low power electronics and design*, 2006, pp. 197 – 202.

[19] C. Park and P. Chou, “Ambimax: Autonomous energy harvesting platform for multi-supply wireless sensor nodes,” in *3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks*, 2006, pp. 168–177.

[20] P. Corke, P. Valencia, P. Sikka, T. Wark, and L. Overs, “Long-duration solar-powered wireless sensor networks,” in *EmNets ’07: Proceedings of the 4th workshop on Embedded networked sensors*. ACM, 2007, pp. 33–37.

[21] C. Alippi and C. Galperti, “An adaptive system for optimal solar energy harvesting in wireless sensor network nodes,” *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 55, no. 6, pp. 1742 –1750, jul. 2008.

[22] D. Krüger, C. Buschmann, and S. Fischer, “Solar powered sensor network design and experimentation,” in *ISWCS’09: Proceedings of the 6th international conference on Symposium on Wireless Communication Systems*. IEEE Press, 2009, pp. 11–15.

[23] J. Alberola, J. Pelegri, R. Lajara, and J. Perez, “Solar inexhaustible power source for wireless sensor node,” in *Instrumentation and Measurement Technology Conference Proceedings, 2008. IMTC 2008. IEEE*, 2008, pp. 657 –662.

[24] T. Esram and P. L. Chapman, “Comparison of Photovoltaic Array Maximum Power Point Tracking Techniques,” *IEEE Transactions on Energy Conversion*, vol. 22, no. 2, pp. 439–449, Jun. 2007.

[25] J. Taneja, J. Jeong, and D. Culler, “Design, modeling, and capacity planning for micro-solar power sensor networks,” in *Proceedings of the 7th international conference on Information processing in sensor networks*, 2008, pp. 407–418.

[26] W. S. Wang, T. O’Donnell, L. Ribetto, B. O’Flynn, M. Hayes, and S. C. O’Mathuna, “Energy harvesting embedded wireless sensor system for building environment applications,” in *1st International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology 2009,*. IEEE, 2009.

[27] A. Kansal, J. Hsu, S. Zahedi, and M. B. Srivastava, “Power management in energy harvesting sensor networks,” *ACM Trans. Embed. Comput. Syst.*, vol. 6, no. 4, p. 32, 2007.

[28] D. Linden and T. Reddy, *Handbook of Batteries*. McGraw-Hill Professional, 2001.

[29] “Nasa/giss atmosphere-ocean model – website,” http://aom.giss.nasa.gov/srlocat.htm, (accessed 17-12-2010).

[30] *PowerStor B-Series (Datasheet 4307)*, Cooper Bussmann, 2008.

[31] *PowerStor Application Guidelines (PS-5006)*, Cooper Bussmann, 2007.

[32] “Sentio series of sensor nodes – website,” http://www.miun.se/stc/Sentio, (accessed 14-01-2011).

[33] *MGSF1N02ELT1 N-channel Mosfet*, ON Semiconductor, 2000.

[34] *MBR0520L Shottky Rectifier*, Fairchild Semiconductor, 2001.

TrickleTree: A Gossiping Approach To Fast And Collision Free Staggered Scheduling

Wojciech Bober, Chris J. Bleakley, Xiaoyun Li
UCD Complex & Adaptive Systems Laboratory
UCD School of Computer Science and Informatics
University College Dublin
Belfield, Dublin 4, Ireland
{wojciech.bober, xiaoyun.li, chris.bleakley}@ucd.ie

***Abstract*—In recent years, data gathering has received significant attention as an application of Wireless Sensor Networks (WSNs). Staggered data tree based protocols have been shown to be successful in reducing energy consumption in data gathering scenarios. An important part of staggered protocols is the process of schedule construction. In order to minimize energy consumption, this process must be fast. In this paper, we present TrickleTree, a fast distributed protocol for establishing staggered and collision free communication schedule. TrickleTree has three functions: to establish routes, i.e., construct a data gathering tree, to establish a staggered communication schedule, i.e, assign time slots to links, and to disseminate the maximal tree depth in the network. To minimize network setup time, TrickleTree combines neighborhood discovery and schedule construction into one step. To ensure that good neighbors are discovered before a node joins the network, TrickleTree uses a rating mechanism. Collisions during node association are reduced by using association slots. To increase the message delivery rate with small message overhead, TrickleTree uses adaptive gossiping. We provide a formal analysis of the protocol properties i.e., collision free scheduling and termination. The behavior of the proposed approach is evaluated in simulation. The results show up to 90% in a reduction in schedule setup time and a 50% reduction of duty cycle compared to a flooding approach.**

***Index Terms*—Wireless sensor networks, staggered schedule, schedule construction, fast association, collisions reduction, association ranking**

I. Introduction

This paper is an extended version of [1]. It contains a modified version of the previously proposed algorithm. It contains analysis of the algorithm with proofs of termination and collision free slot assignment. Additional simulations were added for in-depth evaluation of the algorithm.

The promise of cheap sensors deployed at large scale is attractive for areas such us microclimate research [1], and habitat monitoring [2]. Precise observations produce large quantities of data, that must be transmitted via the network. In addition, these networks are often expected to operate for long periods of time. Although various methods of energy harvesting have been proposed [3], so far using a battery is the most common method of powering nodes. Dutta et al. [4] have shown that radio operation is the main cause of power consumption. Therefore, communication protocols which reduce radio on-time are crucial for achieving the goal of long network lifetime.

Data gathering networks are characterized by a many to one traffic pattern. A common approach to routing in this class of networks is tree based routing. In tree base routing, a node selects a node closer to the sink as its parent. All messages are forwarded only to this node. In order to improve energy-efficiency and data latency a staggered approach has been proposed. In this approach, communication between nodes is scheduled according to their level in the tree (i.e., hop count from the root). Only two consecutive levels off the tree are active at any given time. Hence all nodes in the network must be aware of the maximal tree depth in order to schedule their communication correctly. The quality of wireless links can change considerably in a short amount of time [5]. This means a new schedule must be established each time the network topology changes. Therefore it is important that a staggered schedule is established quickly, so that the cost of control does not exceed the cost of data transmission. We address this problem by proposing TrickleTree, a protocol designed to establish staggered schedule quickly, yet with a small message overhead.

TrickleTree differs from existing protocols, in that it combines neighborhood discovery and schedule construction into one step. This reduces the number of messages which must be exchanged. To ensure that a balanced schedule is constructed, TrickleTree carefully selects the time at which a node starts its association process. This is done by delaying the association process accordingly to a ranking function. The function takes into consideration link quality and the number of potential parents. To reduce the likelihood of a node becoming an orphan these factors are weighted accordingly to the number of messages received by the node.

TrickleTree is based on gossiping, which is a simple but robust and reliable technique of information dissemination. We show how this technique can be applied to establish a staggered schedule. Thanks to adaptive mechanisms derived from the gossiping approach, we are able to balance the delay and communication overhead (energy consumption) required to establish the network tree and communication schedule. TrickleTree is able to derive a collision free schedule, therefore energy is not wasted resolving collisions during the data gathering phase. To the authors' knowledge, this is the first protocol for establishing staggered communication schedules.

The contribution of the work is a novel algorithm which has three functions: 1) to establish communication routes, i.e, construct data gathering tree, 2) to establish a collision free staggered communication schedule, i.e., assign time slots to nodes, 3) and to disseminate the maximal tree depth in the network.

The rest of the paper is structured as follows. In section II we discuss related work. Section III, describes the proposed protocol. In Section IV we provide proof of collision free scheduling and algorithm termination. Section V presents simulation results. We conclude the paper with Section VI.

II. Related work

In this section, we discuss two categories of protocols related to TrickleTree. The first category are protocols which can be used to create staggered schedule. The second, is a category of protocols used to disseminate information in Wireless Sensor Networks.

A. Staggered scheduling protocols

Staggered scheduling was first introduced in D-MAC by Lu [7]. In D-MAC, transmission times are staggered in very short time slots. This reduces latency and end-to-end delivery time because the receiver is guaranteed to be awake at the time of the sender's transmission. Because nodes with at same network depth have the same transmission time offset they must to compete for the channel. A CSMA scheme is used to deal with collisions. The authors do not discuss many practical issues related to the protocol. For example, how nodes learn the maximal routing tree depth in order to calculate their wake-up offset.

A similar concept is used in Merlin [8], a cross-layer protocol integrating MAC and routing. In Merlin staggered scheduling is used for up- and down-link traffic.

TIGRA [9] is a protocol designed for periodic collection of raw data from the network. The authors focus on minimization of the time required to collect the data from the entire network. The collection time is reduced by two techniques. Firstly, data from separate packets is merged into one packet. Note, this is different from data aggregation, where a single value is calculated to represent data from a number of sources. Secondly, collisions are eliminated by ensuring interference-free transmission scheduling.

ASLEEP [10] is a data gathering protocol focusing on reducing message latency. ASLEEP, like TIGRA, uses staggered data gathering to reduce latency. The protocol is able to adapt to varying bandwidth requirements by run-time schedule adjustment. ASLEEP adjusts the active radio time at each level of the tree. Schedule modification is made based on previous traffic trends. If bandwidth requirements are increasing over a certain period of time, then the active period is extended, otherwise the active period is decreased.

In [6], staggered scheduling was combined with synchronous low power listening to reduce energy consumption in low rate data collection networks. Figure1 illustrates the concept of a staggered schedule implemented in Bailigh [6]. The network is organized as a tree. Each node has exactly one parent, which forwards data to the tree root (the sink). Nodes at the same distance from the sink (hop count) are at the same level. At any time only two consecutive levels of the tree are active. In the example, nodes A, B and C, at Level 2 transmit data to nodes D and E, at Level 1, using links 1, 2, and 4. This approach reduces delivery latency because messages are almost immediately forwarded. When the slots are guaranteed to be collision and contention free there is no delay due to backoff, as would occur in a CSMA protocol. However, in the case of distributed scheduling, links 1 and 4 sometimes might use the same time slot. If nodes C and D are in radio communication range this may lead to collision. This is an example of the hidden terminal problem in present in networks using staggered scheduling.

B. Dissemination protocols

When all nodes in the network must share common information, message dissemination is necessary. Early systems used packet floods to disseminate common information such as parameters or commands. Flooding protocols rebroadcast packets they receive [11]. This is a very simple method, which has many disadvantages. Firstly, flooding is unreliable. Due to collisions, some nodes do not receive the information, so typically flooding is repeated. This leads to excessive rebroadcasting and is energy inefficient. To overcome this problem, adaptive protocols have been introduced. They modify node behavior depending on the information they receive. Dissemination protocols which have proven to work reliably in Wireless Sensor Networks are based on gossiping. Trickle [12] was the first proof of concept implementation designed for code dissemination. Trickle is an adaptive gossiping protocol. When nodes detect inconsistent information in the network, they broadcast new information quickly. When nodes agree, they slow down their communication rate exponentially, such that, when in a stable state, they transmit infrequently. Based on this concept, dissemination protocols have been proposed for various applications [13], [14]. TrickleTree is a modification of Trickle used to construct a staggered scheduling tree. We use adaptive beaconing for neighborhood discovery and tree depth dissemination.

TrickleTree is able to minimize collisions between nodes in the tree by scheduling individual transmission slots. We use 2-hop neighborhood information about scheduled time slots to reduce collisions. This technique is often used in TDMA MAC protocols [15]. Fang-Jing Wu [16] shows how collision-free scheduling can be taken advantage of tree based networks.

III. TrickleTree protocol

TrickleTree uses three types of packets Beacon (BCN), Join Request (JREQ) and Join Reply (JREP). Each packet contains a source and destination address. A BCN packet contains the sender's distance from the sink (hop count), parent address, slot number, and the maximal tree depth known to the node, and number of free slots. A JREQ packet contains the same fields. JREP contains the slot number assigned by the parent

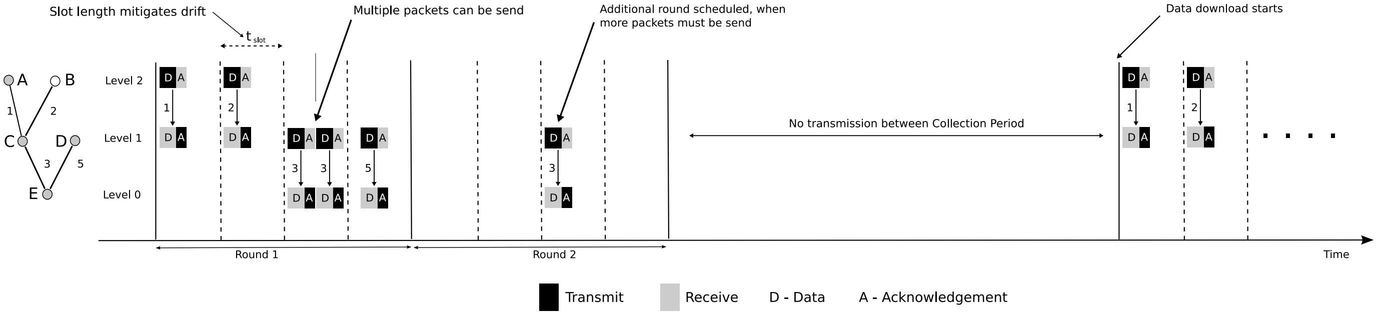

Slot length mitigates drift
t slot
Multiple packets can be send
Additional round scheduled, when more packets must be send
Data download starts
A
B
C
D
E
1
2
3
5
Level 2
Level 1
Level 0
D A
No transmission between Collection Period
Round 1
Round 2
Time
Transmit
Receive
D - Data
A - Acknowledgement

Figure 1: Staggered communication scheme used in [6].

to the child. Note that the purpose of TrickeTree is to quickly create a short living data gathering tree. We assume, that during the tree lifetime (tens of seconds), the network topology remains stable. Therefore, TrickleTree lacks functions typical for routing protocols, i.e., route maintenance. If required, this can be achieved by running TrickleTree periodically.

A node can be in one of six states: *Suspended*, *Listening*, *Joining*, *Collision*, *Gossiping or Connected*. Initially a node starts in a *Listening* state and waits for BCN packets. Upon receiving a BCN packet the node compute value of the ranking function R. After that it delays the start of its association process accordingly to the function value. In general, the lower the value of the function the delay is longer. Each time a BCN packet is received, the ranking function R is computed. If a BCN from a better candidate is received the association process delay is set accordingly to the new value. This means that a node waits for a better candidates for parent before starting association. This is necessary because the neighborhood is discovered during schedule construction.

TrickleTree uses a Shortest Path with threshold $(SP(x))$ metric to select the best parent. This metric selects a node with the smallest distance from the sink among neighbors with link quality exceeding x. The $SP(x)$ metric is used for simplicity. More complicated metrics like MintRoute [17] can be used.

A. Beacon Dissemination

Beacon dissemination based on gossiping is key to the proposed approach, serving multiple purposes. In principle, Trickle adjusts the frequency of information dissemination depending on consistency. When information in the network is consistent (e.g., a version of binary code) beacons are broadcasted infrequently. In contrast, when a node detects inconsistent information, the frequency of beaconing is increased. Consistency in the network is determined by overhearing beacons from other nodes. In Trickle dissemination, information is divided into metadata and the data itself. This allows separate transmission of large data (e.g., binary code) from version information. In TrickleTree only small beacons are broadcasted thus there is no separation between metadata and data.

Table I: SUMMARY OF SYMBOL DEFINITIONS

Symbol	Description	Symbol	Description
t	beacon timer	t_d	discovery timer
τ_l	low gossiping period	t_g	gossiping timer
τ_h	high gossiping period	s_a	# of available slots
j_s	join slot	t_{pkt}	packet duration
j_{max}	max # of join slots	t_{ack}	ack duration
l	node level	j_{dly}	join delay
d	tree depth		

Table II: TrickleTree dissemination algorithm pseudocode.

Event	Action
τ expires	If c $>$ 0 double τ, up to τ_h. Set $c = 0$, pick a new t.[1]
t expires	If $c < k$ or $c = 0$, transmit.
Receive BCN and $BCN(d) = d$	Increment c.
Join network; change level; Receive BCN and $BCN(d) \neq d$	Set τ to τ_l, Set $c = 0$, pick a new t.[1]

[1] t is a random value from the range $[\frac{\tau}{2}, \tau)$

The symbol definitions used in TrickleTree are described in Table I. TrickleTree uses a modified version of the Trickle algorithm. In the original algorithm, the gossiping period τ is doubled whenever the previous gossiping period expires. In TrickleTree τ is doubled only if in the previous gossiping period a beacon with consistent tree depth (d) was received ($c > 0$). This is a simple method of detecting collision due to a hidden terminal: if the node broadcasts a beacon, it should receive at least one from one of its neighbors. For the same reason, a beacon is transmitted whenever the beacon period t expires and no beacon with the same tree depth is received ($c = 0$). This is different from the original Trickle algorithm. We have also extended the list of events on which the gossiping period is set to its lowest value. Table II presents pseudocode for the TrickleTree algorithm.

Upon receiving a BCN packet, a node performs a set of actions. Information from the packet is used to construct a neighbor table. Each record of the table consists of a node address, distance from the sink, assigned slot number, received signal strength, and time synchronization data. Every time a node receives a BCN packet from a node which is already in the table, the information is updated.

Because staggered data gathering requires maximal tree depth for timing offset calculation, all nodes in the network must agree on a common value. Each node connected to the network disseminates the maximal tree depth it currently holds in the BCN packet. If a node overhears a BCN packet which has a maximal tree depth field value greater than its current tree depth value, the node updates its current tree depth to the received value. It then resets the counter c, sets τ to τ_l and picks up a new t value. This allows for fast dissemination of a new tree depth in the network. TrickleTree does not have explicit mechanisms to check the consistency of the tree depth value. We rely on the general convergence property of dissemination protocols based on gossiping.

Note that a node is allowed to participate in gossiping only when it is in the *Gossiping* or *Connected* state, i.e., after joining the network. Initially only the sink node is able to disseminate beacons.

B. Node association

Most schedule based data gathering protocols build schedules based on data gathering tree. To build such a schedule, a parent - child relationship must be established between nodes. To establish this relationship a node associates with a node closer to the sink. This node becomes the node's parent. In the simplest case, a node starts the association procedure immediately or shortly after receiving a beacon from a potential parent. This method, used in [9], [18], is shown in Figure 2b. Although simple, there are two main drawbacks of this method. Firstly, it relies on the MAC to resolve collisions. These collisions are caused by multiple nodes requesting association to the same node. Secondly, it does not take into consideration the quality of the link to a selected node. To address this issue, a threshold T_h on the link quality is often introduced. In this case, the association process is started only when the link quality is above minimal threshold. To address the former issue a method based on association slots was proposed in [19]. In this method, nodes use time slots to perform the association process (Figure2b). Whenever a node wants to join the network, it selects a random slot and starts the association process. Because the length of the slot is sufficient for the whole association process, i.e., exchange of request and reply packets, collisions at the MAC layer are reduced. As in the previous methods, a threshold on link quality is used to ensure that only good links are used. The method proposed herein improves on the method proposed in [19]. Instead of using a random slot, a node selects a slot according to a ranking function (1). The value of the function is computed each time a BCN packet from a neighboring node is received. The function takes into consideration the signal quality σ, the degree of a node δ (defined as a number of potential parents), and the total number of beacons received β. The aim of the function is to assign a higher rank i.e., earlier join slot, to nodes which have a better link to the node broadcasting a beacon. It also takes into consideration how many potential parents a node has, e.g., if a node has only one potential parent in its neighborhood its priority will be higher than a node with more potential parents, even if the link quality is worse. The function uses the number of beacons received from neighboring nodes as a weight for both factors. As the number of beacons increases, degree of a node δ gains on significance. This is to reduce the number of orphan nodes and forced associations.

$$q_\sigma = \frac{(\sigma - \sigma_L)}{\sigma_H - \sigma_L},\; q_\delta = \frac{(\delta - \delta_h)}{\delta_L - \delta_H}\; q_\beta = \frac{(\beta - \beta_L)}{\beta_H - \beta_L}$$

$$R(\sigma, \delta, \beta) = (1 - q_\beta)q_\sigma + q_\beta q_\delta \tag{1}$$

Based on the value of the ranking function, a node calculates association delay (2) and (3).

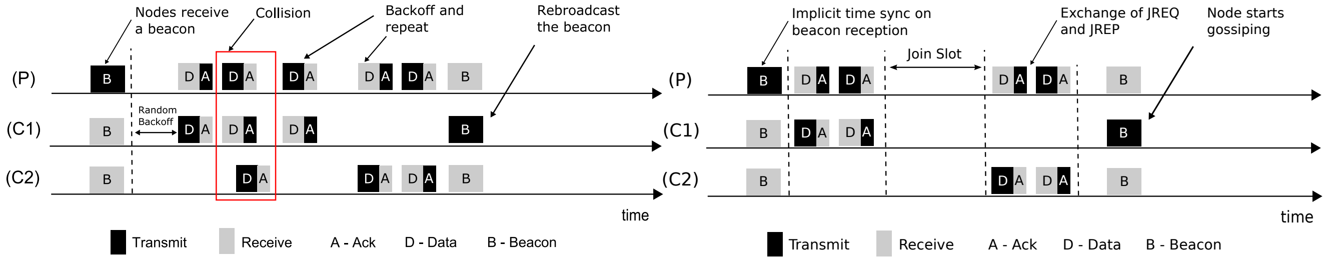
$$j_s = \lfloor j_{max}(1 - R) \rfloor \tag{2}$$

$$j_d = 2(t_{pkt} + t_{ack})j_s \tag{3}$$

Each BCN received by a node is an implicit synchronization point. Each join slot is long enough to allow for JREQ and JREP exchange as well to mitigate for clock drift. If the JREQ packet is delivered successfully, the node will set up a JREP timeout. Setting a timeout on the JREP prevents the node from infinite waiting in the case that the selected node does not reply. This might happen when the potential parent fails before it manages to send a JREP. In the case of delivery failure, JREP timeout, or join rejection, the node will return to the *Listening* state. It might happen, however, that two nodes will request association in the same slot. This is more probable when node density is high and the value of j_{max} is low. If both nodes are in the radio range, a node will detect an on going transmission. In this case the node will repeat the association procedure after receiving the next beacon. If both nodes are not in the radio range, most likely their JREQ packet will collide. This is reported by the MAC layer. A node switches to Random mode and repeats the association procedure after receiving the next beacon.

A node accepts JREQ packets only when it is in the *Gossiping* state. A parent node which receives a JREQ packet from a node selects an slot using Algorithm 1. The slot is marked as used by the node which requests the join and a JREP packet is sent back. In the case that there are no free slots left, the node will refuse to accept the join request and send a JREP packet with the REFUSED flag set.

Upon receiving the JREP packet, the node cancels the JREP timeout. If Collision Free (CF) mode is enabled, the node compares the assigned slot number with the slot numbers assigned to nodes stored in its neighbor table. If a node with the same slot number at the same level exists, it indicates a slot collision. In this case, the node enters the *Collision* state and initiates a collision resolution procedure (see Section III-E). If a node with the same slot number does not exist or CF mode is disabled, the node stores the slot number and enters the *Gossiping* state. Once connected, the node enables BCN packet dissemination as described in Section III-A. It also sets a gossiping timer t_g. This timer determines how long a node maintains in the *Gossiping* state. After the timer expires the node enters the *Connected* state.



(a) Association procedure based on MAC layer.

(b) TrickleTree association procedure.

Figure 2: Comparison of association procedures. (P) denotes a potential parent node, whereas (C1) and (C2) denote child nodes.

C. *Forcing association*

Occasionally a node has only one potential parent in its neighborhood. In this case, the node can force association with the selected node. The node sends a JREQ with the FORCE flag set. Upon receiving this packet, the parent will select the child with the highest degree d and remove it by sending JREP with the REFUSED flag set. The slot released in this way will be assigned to the node forcing join by sending a JREP packet. The node which was removed will attempt to join a different parent, as described in the previous section. Forced association ensures that the whole network is connected as long as good quality links are available.

D. *Detecting collisions*

As mentioned previously, TrickleTree can work in Collision Free (CF) mode. If CF mode is enabled, TrickleTree ensures that all nodes have individual transmission slots. To achieve this, once in *Gossiping* state, a node constantly monitors received BCN packets in order to detect potential collisions. This is done by comparing its level and slot number with the received values. Equal values indicate slot collision. In this case, the node will change its state to *Collision* and request a new slot from its parent as described in Section III-E. If a node is a parent and the level of the node in the received BCN packet is equal to the level of its children then the node will check if there is a child node assigned to the same slot number. If so, the node will use a collision resolution procedure to assign a new slot to its child. A node will always invalidate a given slot, so that it cannot be used to schedule children. It is possible that a BCN packet indicating collision is received by the parent and child at the same time. In order to prevent both nodes from requesting a new slot at the same time, join requests sent by children are delayed. This allows the parent to solve the collision first, which is preferred since it requires transmission of only one packet. The method used in TrickleTree to detect collisions uses two hop neighborhood information to assign individual slots. This method is often used by TDMA MAC protocols [15] to handle collisions from hidden terminals. In TrickleTree collisions from hidden terminals are detected and resolved by either the parent or child, as described in detail in the next section.

Algorithm 1 Slot selection algorithm

```
if $s_l > 0$ then
    $s_r \leftarrow random(s_c)$
    if $s_r < s_c - 1$ then
        $s_i \leftarrow s_r + 1$
    else
        $s_i \leftarrow 0$
    end if
    while $s_i <> s_r$ AND ($s_i$ is Free OR $s_i$ is Valid) do
        if $s_i < s_c - 1$ then
            $s_i \leftarrow s_i + 1$
        else
            $s_i \leftarrow 0$
        end if
    end while
    $s_l \leftarrow s_l - 1$
    return $s_i$
else
    return $NONE$
end if
```

E. *Resolving collisions*

The method of resolving a detected collision depends on the node's role. A node can solve a collision as a parent, child, or intermediate node.

1) *Parent collision resolving procedure:*

- A beacon from an unrelated child node is received: the collision is solved by assigning a new slot to the child node. The parent invalidates the collided slot and selects a different slot. Next, the parent sends a JREP packet to its child node with the new slot. If slot cannot be assigned, the child node is disconnected from the parent and a JREP packet with REFUSED flag is send back. This forces the child node to connect to a other parent. If a node which is a parent for different nodes cannot connect to the network it will send a JREP to all its children with the REFUSED flag set.
- A beacon from an unrelated parent node is received: in this case, a node compares s_a with $BCN(s_a)$. If $s_a > BCN(s_a)$ then the node changes its own child slot, as

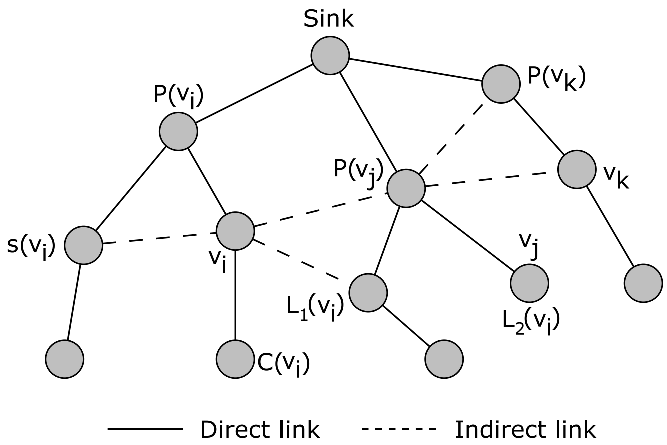


Figure 3: Classification of nodes with respect to v_i

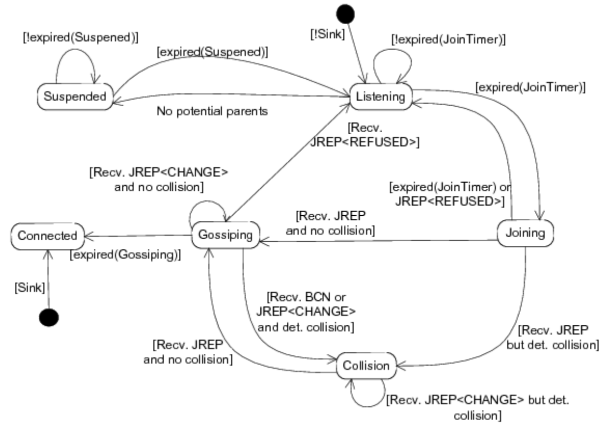

[!expired(Suspened)]
[expired(Suspened)]
[!Sink]
[!expired(JoinTimer)]
Suspended
Listening
[expired(JoinTimer)]
No potential parents
[Recv. JREP<REFUSED>]
[Recv. JREP<CHANGE> and no collision]
[expired(JoinTimer) or JREP<REFUSED>]
[Recv. JREP and no collision]
Connected
Gossiping
Joining
[expired(Gossiping)]
[Sink]
[Recv. BCN or JREP<CHANGE> and det. collision]
[Recv. JREP and no collision]
[Recv. JREP but det. collision]
Collision
[Recv. JREP<CHANGE> but det. collision]

Figure 4: TrickleTree finite state machine diagram. PACKET<FLAG> notation is used to denote a flag which must be set in a packet for a transition to occur. Expired() notation is used for timers.

described previously. If $s_a < BCN(s_a)$ then the node requests the other node to change slot.

2) Child collision resolving procedure: As a child node, the collision is resolved by requesting a new slot from its parent. This is done by sending a JREQ packet to the current parent.

IV. Analysis of the TrickleTree algorithm

In this section we provide formal analysis of TrickleTree algorithm, which is the core of the TrickleTree protocol presented in Section III. Formally we consider a network modeled as an unidirected graph $G = (V, E)$, where V contains all nodes and E contains all communication links. The goal is to construct a tree T and a communication schedule S from G rooted at the sink. The communication schedule uses a time-division model, where time is divided into fixed length slots. The slots are grouped in a frame, which has a fixed length. The communication schedule is constructed by assigning a slot s_i to each node $v_i \in V$.

Definition 1. *Given a node v_i and a data collection tree T in G, we define $level(v_i)$ as the distance in hops of v_i from the sink, $P(v_i)$ as $v_i's$ parent, $N_n(v_i)$ as the set of v_i's n-hop neighbors, $L_n(v_i)$ as the set of $v_i's$ n-hop neighbors where $\forall\, v_j \in L_n(v_i) : level(v_j) = level(v_i)$. We define v_i's interference set as $I(v_i) = L_1(v_i) \cup L_2(v_i)$.*

Note, that TrickleTree uses a dynamic interference set, i.e. as nodes join the data gathering tree, new nodes whose slot assignment might collide will appear in the interference set.

Theorem 1. TrickleTree ensures collision free collection schedule.

Proof: By definition for each pair of v_i and v_j where $level(v_i) \neq level(v_j)$ the schedule is interference free even if $s_i = s_j$ as the transmission is separated temporally. Therefore collision is possible only if a node $v_j \in I(v_i)$. We prove Theorem 1, by showing that TrickleTree provides a collision free assignment for all cases where $v_j \in I(v_i)$.

1) A node $v_j \in L_1(v_i) \wedge P(v_j) = P(v_i)$, in this case a node v_j is $v_i's$ sibling and collision is not possible as the parent assigns different slots to its children.
2) A node $v_j \in L_1(v_i) \wedge P(v_j) \notin N_1(v_i)$, in this case assignment $s_i = s_j$ is detected by overhearing BCN packets by either node v_j or v_i. The collision is then resolved using the child collision resolution procedure.
3) A node $v_j \in L_1(v_i) \wedge P(v_j) \in N_1(v_i)$, in this case assignment $s_i = s_j$ might be detected by overhearing by node v_j and $P(v_j)$ simultaneously. The procedure parent collision resolution procedure takes precedence over the child collision resolution procedure. Thus, the collision is resolved.
4) A node $v_j \in L_2(v_i) \wedge P(v_j) \in N_1(v_i)$, in this case assignment $s_i = s_j$ is detected by overhearing by node $P(v_j)$. The collision is resolved using the parent collision resolution procedure.
5) A node $v_j \in L_2(v_i) \wedge P(v_j) \notin N_1(v_i)$, in this case assignment $s_i = s_j$ is detected by intermediate node m_{ij}. The node m_{ij} sends a unicast message either to node v_i or v_j accordingly to the intermediate node collision resolution procedure.

■

Definition 2. *We define $C(v_i)$ as the set of potential parents where $\forall\, v_j \in C(v_i) : v_j \in N_1(v_i) \wedge q(v_j) > q_0$ and $q(v_i)$ is a function which allows for assessing quality of the link to a node v_i. We define $A \rightarrow B$ as transition from state A to state B, and $A \xrightarrow{t} B$ as timeout transition from state A to state B i.e., transition which happens after time t unless other condition occurs earlier.*

Theorem 2. TrickleTree terminates.

Proof: The finite state machine of TrickleTree is shown in Figure 4. We prove Theorem 2 by showing that, for each node v_i participating in schedule, a transition to the $Connected$ or $Suspend$ states occurs.

1) All nodes, apart from the sink, start in the $Listen$ state.

Table III: SIMULATION PARAMETERS.

Parameter	TelosB	Unit
P^*_{tx}	58.5	mW
P^*_{rx}	65.4	mW
P^*_{sleep}	0.015	mW
P^*_{poll}	14.1	mW
t_{poll}	2.5	ms
t_{cca}	2	ms
Data Rate	250	kbps
Voltage	3	V

Parameter	TT	Tigra
Retransmission #	3	10
Slot Count	10	
Buffer Size	20 packets	
Clock drift	100 ppm	
Initial Backoff	-	16
Packet Size	48 bytes	

Each node v_i sets a discover timer t_d. During time t_d a node v_i constructs set $C(v_i)$.

2) If $|C(v_i)| = 0$ after t_d, transition $Listen \overset{t_d}{\rightarrow} Suspend$ occurs. Hence, the algorithm terminates because there are no potential parents.
3) If $|C(v_i)| \neq 0$ after t_d, transition $Listen \overset{t_d}{\rightarrow} Joining$ occurs. A node v_i selects the first node $v_j \in C(v_i)$ as $P(v_i)$. The join procedure is started.
4) A node may go through a number of transitions $Joining \rightarrow Collision$, $Collision \rightarrow Gossiping$, and $Gossiping \rightarrow Collision$.
5) Each time a node v_i enters the $Collision$ state, v_i is assigned a new slot s_i by $P(v_i)$. If $P(v_i)$ cannot assign a new slot to v_i the node v_j is moved from $C(v_i)$ to $\overline{C}(v_i)$. The next node $v_j \in C(v_i)$ is selected as $P(v_i)$. The join procedure is repeated.
6) Since $C(v_i)$ is finite, a node v_i can enter the $Collision$ state a limited number of times. In this case, a node v_i selects a node v_j from $\overline{C}(v_i)$ and forces join. Due to this, the transition $Collision \rightarrow Connected$ is made and the algorithm terminates.
7) Each time a node v_i enters the $Gossiping$ state a timer t_g is set. Transition $Gossiping \overset{t_g}{\rightarrow} Connected$ occurs after t_g. The algorithm terminates.

■

V. EVALUATION

A. Simulation Model

In order to verify the behavior of the proposed algorithm, a set of simulations were performed using the OMNeT++ [20] simulator. The simulations were performed using 10-50 randomly deployed nodes. In the two first scenarios the simulation area was 35×35 m^2, whereas in the last the area was 65×65 m^2. In each case, the sink was placed in the center of the simulated area.

The simulation uses the Log-Normal Shadowing Model [21] for wireless signal propagation. The model takes into consideration the effects of wireless signal fading and shadowing. These effects, common in wireless transmissions, are modeled by adding a perturbation factor to the reception power. This factor follows a normal distribution, with a standard deviation σ which can be defined for each simulation run. In addition, the asymmetry of links is modeled. To capture the effects of wireless interference, the simulation uses an physical (additive) interference model [22]. In this model, reception probability is determined by signal-to-noise ratio. The sum of power from multiple concurrent transmissions may cause interference at a given node, even though separately each is below the receiver sensitivity.

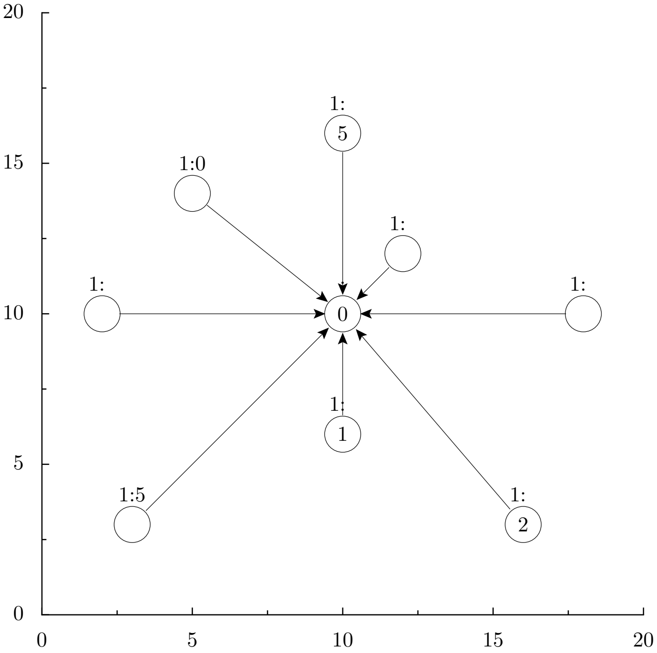


Figure 5: Example of node placement in association experiment.

A model of the Chipcon CC2420 [23] transceiver, which conforms to the IEEE 802.15.4 specification, was used in the simulation. The transceiver is modeled as a finite state machine consisting of tree states: sleep, receive, and transmit. Delays in transition between respective states were modeled, which allows for precise calculation of the duty cycle and energy consumption. In addition, the radio model includes properties such as, modulation type (PSK), sensitivity, and bit error rate. These properties influence the signal propagation of the model.

Each simulation was repeated 100 times and a 0.05 confidence level was used to calculate respective confidence intervals. For each simulation, nodes were uniformly distributed in the simulation field. Motes boot up with start-up times randomized according to a uniform distribution. In all simulations we measured the time required to establish a network schedule. We considered the network topology to be established when all nodes were connected, have the same maximal tree depth value, and no schedule collisions exist. TrickleTree can use any MAC protocol. Herein, the protocol is evaluated with B-MAC [24].

B. Results

1) Association time: The first set of experiments was performed to assess the impact of the association scheme on the number of collisions and on association time. In the experiment, the sink was placed in the center of the field. A number of nodes within radio range were placed around the

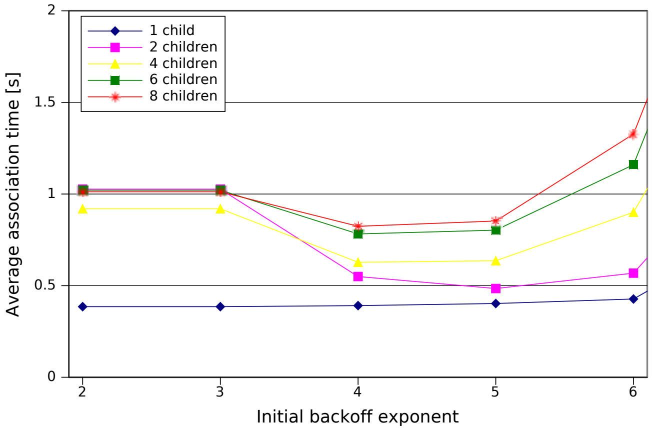


Figure 6: Avg. association time for *MAC-Exponential* scheme.

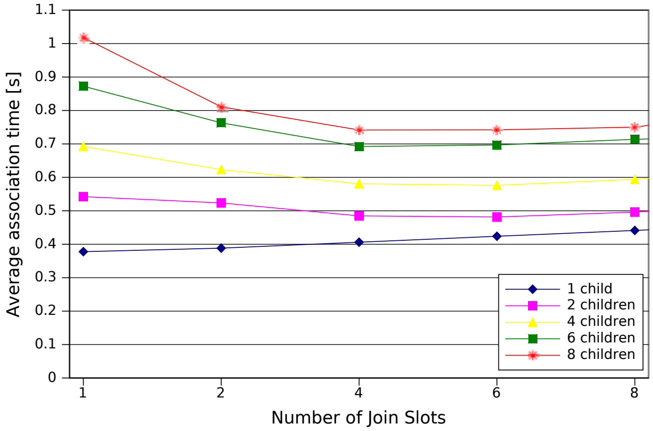


Figure 8: Avg. association time for *TrickleTree-Rank* scheme.

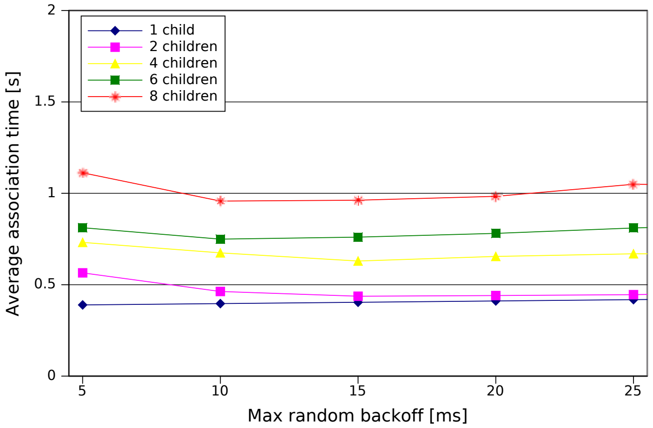


Figure 7: Avg. association time for *MAC-Random* scheme.

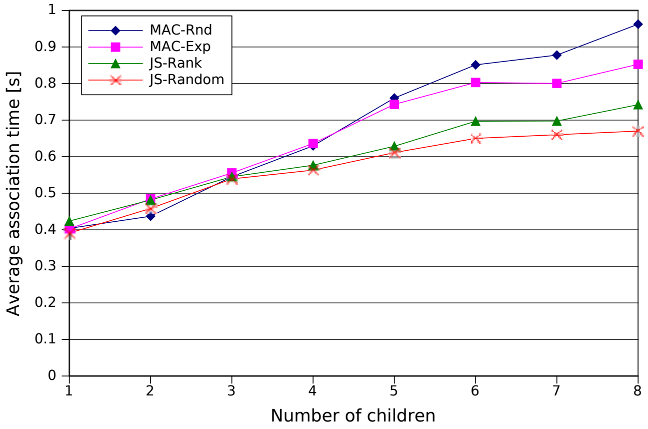


Figure 9: Comparison of avg. association time.

sink. The distance between each child and the sink was varied, so that a different RSSI value was achieved for each child. An example scenario is shown in Figure 5. Note that, association time is different from schedule construction time. The later includes the time required to resolve schedule collisions and disseminate the tree value, whereas the former denotes how fast a node associates with its parent. We compare four different association schemes. Two schemes are based on Medium Access Control: random backoff offset *(MAC-Rnd)* and exponential backoff offset *(MAC-Exp)*. We compare these MAC based schemes with two schemes based on join slots: slot calculation based on rank *(JS-Rank)* and randomly chosen slot *(JS-Random)*. In all cases, the protocols use TrickleTree's beacon dissemination method and the low gossiping period was $\tau_l = 0.5$s. The initial beacon was broadcasted at time randomly chosen between 0.25s-0.5s. For this reason, the average association time is greater than 0.35s. We disabled LPL in order to eliminate additional delay. Figure 6 shows the average association time for *the MAC-Exp* scheme. It can be seen that association based on exponential backoff has an optimum near backoff exponent 4-5. This corresponds to a backoff limit of 16-32ms. A similar observation, although the effect is less pronounced, can be made with *MAC-Rnd* (Figure 7). The optimal value for the *MAC-Rnd* scheme is close to 15ms. The results can be interpreted as follows: too small backoff value can cause high contention and a large number of collisions. This increases association time as nodes have to backoff multiple times (in the case of contention) or repeat the association process (in the case of collision). When a large backoff duration is used, delay is not increased due to contention or collision, but due to the duration of the initial backoff.

Figure 8 shows an average association time for the *JS-Rank* scheme. For this scheme, we varied the maximal number of join slots j_{max}. The optimal value of j_{max} should be close to the average neighborhood size of a node. This ensures that association requests are spread equally in time. When j_{max} is too low, a higher association time is needed due to the increased number of collisions. With high j_{max}, the ranking function has more impact on association delay. This can be seen clearly in the case of a single child. Since no collisions or contentions are involved, the association time depends solely on the outcome of the ranking function. Hence it increases with j_{max}. In Figure 9, we compare the association time of all

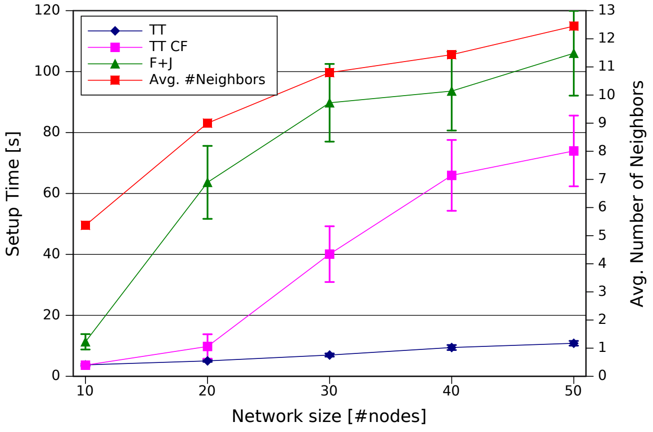


Figure 10: Tree setup time as function of network size. TT denotes TrickleTree algorithm, whereas F-J denotes the flooding join algorithm.

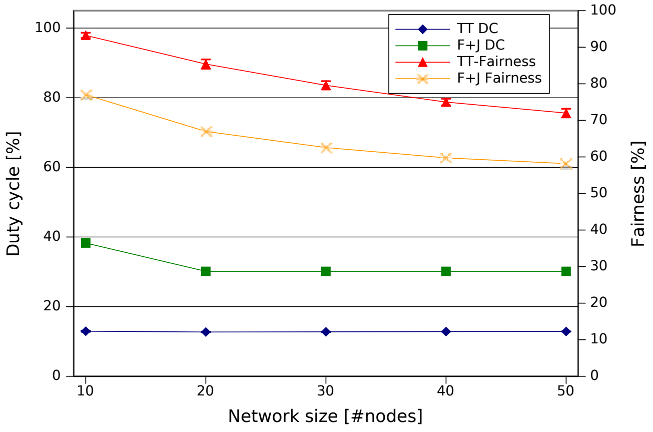


Figure 12: Duty cycle as function of network size. TT denotes the TrickleTree algorithm, whereas F-J denotes the flooding join algorithm.

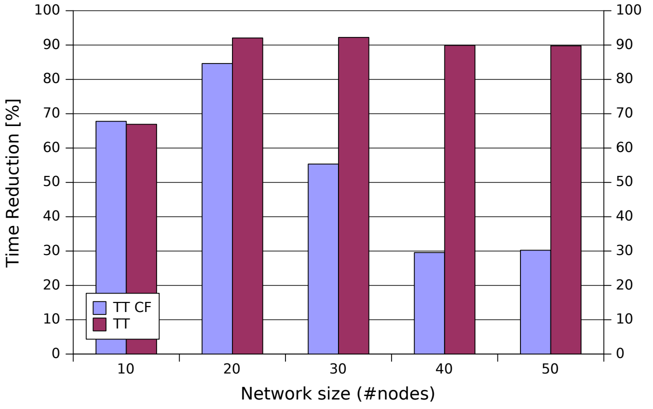


Figure 11: Network setup time reduction in relation to flooding join.

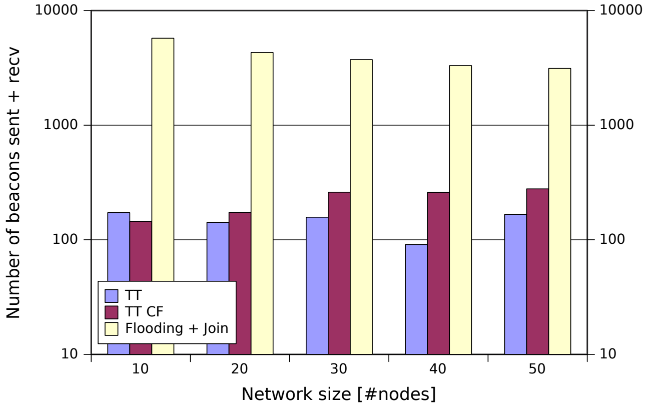


Figure 13: Sum of beacons sent and received for various network sizes.

schemes. For each scheme, we selected the lowest association time for a given number of children i.e., the one with optimal backoff duration or number of slots. It can be seen that for a small number of children (less than 3), all schemes perform similarly. As the number of contending nodes increases, the time required to complete association differs. For number of children greater than 3, the *JS-Rank* scheme performs better than both *MAC-Rnd* and *MAC-Exp*. On average, JS-Rank is faster than *MAC-Rnd* and *MAC-Exp* by 17% and 12%, respectively. The *JS-Random* scheme achieves slightly faster association time. This is because there is no delay introduced by the ranking function. On average, the JS-Random scheme is faster than *MAC-Rnd* and *MAC-Exp* by 21% and 17%, respectively.

2) Schedule construction time: The main goal of TrickleTree is to quickly establish a staggered data gathering tree with minimal energy overhead. TrickleTree uses join slots to reduce collisions and improve setup time. Adaptive gossiping is used to reduce message overhead and ensure that changes in tree depth are disseminated quickly. To verify TrickleTree performance it was compared with a flooding approach, similar to that used in [25], and Tigra [9].

In the flooding approach, the sink periodically broadcasts a tree setup beacon. Upon receiving the beacon, unconnected nodes attempt to join a selected parent. Nodes which are connected to the tree rebroadcast received beacons. The flooding approach relies on the MAC protocol to resolve collisions. If not all nodes can join the network in given simulation run, the results were rejected and not included in calculations.

Tigra is a state-of-the art data gathering protocol. It is one of a few protocols which discuss the process of staggered schedule construction. We implemented a slightly modified version of the algorithm described in [9]. The first modification was necessary because the original algorithm is designed to assign a round to a node based on the number of its descendants. In TrickleTree the round is the same as the level of a node in the tree. The modification does not change the number of exchanged messages nor the exchange procedure. We simply changed the contents of the packets. Instead of number of descendants the *RESPONSE* packet contains level

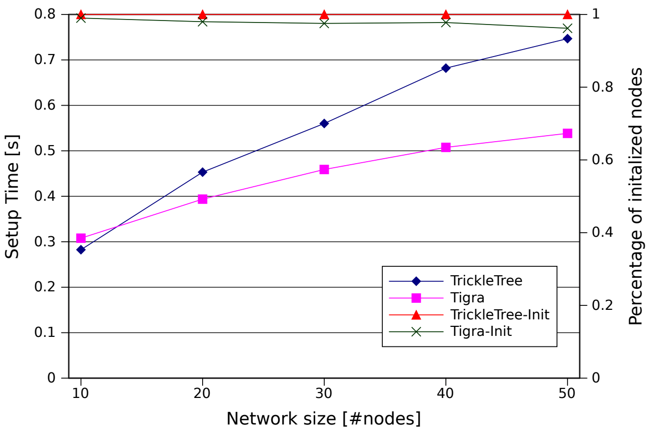


Figure 14: Comparision of Initialization time of Tigra and TrickleTree

of a leaf node. When all response packets get to the sink, the sink is able to determine the maximal tree depth. After that the *INIT2* packet is sent down the tree, to inform all nodes about the maximal tree depth.

Figure 10 shows the setup time and average neighborhood size. It can be seen that as the average number of neighbors grows, the performance of the flooding approach (F+J) deteriorates considerably. This is due to the high number of collisions which must be resolved by the MAC layer. Moreover, nodes ignore information contained in the received beacons and rebroadcast them, even though the network state is consistent. This causes additional collisions due to the hidden terminal problem. TrickleTree (TT) on the other hand, performs more consistently. Collisions are reduced, due to the fact that nodes calculate distinctive join slots, therefore the number of collisions which must be resolved at the MAC layer is reduced. When collision free scheduling is enabled (TT CF), Trickle Tree requires more time to establish the schedule. This is clearly seen in Figure 13, which shows the average number of beacons sent and received. It can be seen that the collision free version of TrickleTree requires up to 65% more beacons to be sent. This is because each change in a schedule (e.g., slot change) resets the gossiping period to the lowest value, therefore beacons are sent more frequently. Figure 13 shows the advantage of adaptive gossiping as used in TrickleTree, over the flooding approach. Adaptive gossiping reduces the average number of beacons from 4000 in the flooding approach to 145 and 223 for regular and collision free TrickleTree, respectively. Figure 11 shows the time reduction achieved using the regular and collision free versions of TrickleTree with respect to the flooding approach. The advantage of TrickleTree is increases with the network size. In the flooding approach large numbers of nodes produce more of collisions slowing down schedule setup. As indicated previously, the collision free version of TrickleTree requires more time to setup the network. There is a tradeoff between fast schedule setup time and possible collisions during data gathering and slower schedule setup and collision free transmissions. The decision as to which version of TrickleTree should be used depends on the application requirement and network stability. For unstable networks, low setup time is important as the network might be rescheduled frequently. Therefore control overhead should be reduced. In stable networks, higher control overhead for setting up collision free schedule will be balanced in the long run by a collision free data gathering phase.

Figure 14 shows a comparision of initialization time of TrickleTree and Tigra. As it can be seen Tigra, on average, is 15% faster than TrickleTree. This is because TrickleTree introduces a delay in the association in order to find good links. In Tigra nodes connect to a node from which the first beacon is received. However, Tigra was not able to complete schedule construction in most of the cases i.e., connect 100% of the nodes and agree on a common depth of the tree. This is because the Tigra assumes lossless links and reliable packet delivery by the MAC layer. In this regards results obtained in herein confirm results obtained by the authors in a testbed [26]. TrickleTree completed schedule construction in all simulated cases.

3) Duty cycle: In Figure 12, the duty cycle of both protocols is shown. The average duty cycle of TrickleTree over different network sizes is 12%, whereas the duty cycle of flooding is 32%. The duty cycle is reduced by adaptive gossiping. Although, the duty cycle of the flooding approach can be reduced by increasing the sink broadcast period it results in a longer network setup time. Fairness of both protocols is also shown in Figure 12 confirms that the main reason for TrickleTree's performance is reduction in collisions. In general, the improvement in network setup time depends on network size. TrickleTree reduces setup time by 68% for networks of 10 nodes, to nearly 90% for networks of 50 nodes.

VI. Conclusions

Data gathering is one of the most recognized applications of wireless sensor networks. As available network energy is a very limited resource and radio communication exploits this resource the most, developing efficient communication protocols is an important task. In this work, we proposed a practical approach to scheduling staggered networks based on gossiping. To validate the behavior of the proposed approach we performed a number of simulations and the results show up to 90% reduction of schedule setup time and 50% reduction duty cycle compared to the conventional flooding approach.

Acknowledgments

This work is supported by Enterprise Ireland grant number CFTD/07/IT/303, "Network-level power management for wireless sensor networks".

References

[1] G. Tolle, J. Polastre, R. Szewczyk, D. Culler, N. Turner, K. Tu, S. Burgess, T. Dawson, P. Buonadonna, D. Gay, and W. Hong, "A macroscope in the redwoods," in *Proc. of ACM SenSys*. ACM, 2005, pp. 51–63.

[2] R. Szewczyk, A. Mainwaring, J. Polastre, J. Anderson, and D. Culler, "An analysis of a large scale habitat monitoring application," in *Proc. of ACM SenSys*. ACM, 2004, pp. 214–226.

[3] S. Roundy, P. K. Wright, and J. M. Rabaey, *Energy Scavenging for Wireless Sensor Networks: With Special Focus on Vibrations*, Norwell, MA, USA, 2004.

[4] P. Dutta, D. Culler, and S. Shenker, "Procrastination might lead to a longer and more useful life," in *Proceedings of HotNets-VI, Atlanta, GA, November*, 2007, pp. 1–7.

[5] K. Srinivasan, M. A. Kazandjieva, S. Agarwal, and P. Levis, "The B-factor: measuring wireless link burstiness," in *Proc. of ACM SenSys*. New York, NY, USA: ACM, 2008, pp. 29–42.

[6] W. Bober and C. Bleakley, "Bailigh: Low power cross-layer data gathering protocol for Wireless Sensor Networks," in *Proc. of ICUMT*, Oct. 2009, pp. 1–7.

[7] G. Lu, B. Krishnamachari, and C. S. Raghavendra, "An adaptive energy-efficient and low-latency mac for tree-based data gathering in sensor networks: Research articles," *Wirel. Commun. Mob. Comput.*, vol. 7, no. 7, pp. 863–875, 2007.

[8] A. G. Ruzzelli, G. M. P. O'Hare, and R. Jurdak, "Merlin: Cross-layer integration of mac and routing for low duty-cycle sensor networks," *Ad Hoc Networks*, vol. 6, no. 8, pp. 1238–1257, 2008.

[9] L. Paradis and Q. Han, "A data collection protocol for real-time sensor applications," *Pervasive and Mobile Computing*, vol. 5, no. 4, pp. 369 – 384, 2009.

[10] G. Anastasi, M. Conti, and M. Di Francesco, "Extending the lifetime of wireless sensor networks through adaptive sleep," *Industrial Informatics, IEEE Transactions on*, vol. 5, no. 3, pp. 351 –365, Aug. 2009.

[11] J. Lu and K. Whitehouse, "Flash flooding: Exploiting the capture effect for rapid flooding in wireless sensor networks," in *Proc. of INFOCOM*, 2009, pp. 2491 – 2499.

[12] P. Levis, N. Patel, D. Culler, and S. Shenker, "Trickle: a self-regulating algorithm for code propagation and maintenance in wireless sensor networkstr," in *Proc. of USENIX NSDI*, Berkeley, CA, USA, 2004, pp. 2–2.

[13] G. Tolle and D. Culler, "Design of an application-cooperative management system for wireless sensor networks," in *Proc. of EWSN*, 2005, pp. 121–132.

[14] K. Lin and P. Levis, "Data Discovery and Dissemination with DIP," in *Proc. of IEEE IPSN*, 2008, pp. 433–444.

[15] M. Nunes, A. Grilo, and M. Macedo, "Interference-Free TDMA Slot Allocation in Wireless Sensor Networks," in *LCN '07: Proceedings of the 32nd IEEE Conference on Local Computer Networks*, 2007, pp. 239–241.

[16] Y.-C. T. Fang-Jing Wu, "Distributed wake-up scheduling for data collection in tree-based wireless sensor networks," vol. 13, no. 11, pp. 850 –852, November 2009.

[17] A. Woo, T. Tong, and D. Culler, "Taming the underlying challenges of reliable multihop routing in sensor networks," in *Proc. of ACM SenSys*. ACM, 2003, pp. 14–27.

[18] N. Burri, P. von Rickenbach, and R. Wattenhofer, "Dozer: ultra-low power data gathering in sensor networks," in *Proc. of IEEE IPSN*. IEEE Computer Society, 2007, pp. 450–459.

[19] W. Bober, C. Bleakley, and X. Li, "TrickleTree: A Gossiping Approach To Fast Staggered Scheduling For Data Gathering Wireless Sensor Networks," in *4th Int. Conf. on Sensor Technologies and Applications (SENSORCOMM)*. IEEE Computer Society, 2010, pp. 214–219.

[20] A. Varga. (Last accessed: 04/2010) OMNeT++. http://www.omnetpp.org.

[21] T. Rappaport, *Wireless Communications: Principles and Practice*. Upper Saddle River, NJ, USA: Prentice Hall PTR, 2001.

[22] A. Iyer, C. Rosenberg, and A. Karnik, "What is the right model for wireless channel interference?" in *QShine '06: Proceedings of the 3rd international conference on Quality of service in heterogeneous wired/wireless networks*. New York, NY, USA: ACM, 2006, p. 2.

[23] (Last accessed: 04/2010) CC2420 Data Sheet. http://www.ti.com. Texas Instruments.

[24] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in *Proc. of ACM SenSys*. ACM, 2004, pp. 95–107.

[25] S. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, "TAG: a Tiny AGgregation service for ad-hoc sensor networks," *SIGOPS Oper. Syst. Rev.*, vol. 36, no. SI, pp. 131–146, 2002.

[26] L. Paradis, "Tigra: Timely sensor data collection using distributed graph coloring," Master's thesis, Colorado School of Mines, 2007.

Phased Array Satellite Antenna Testing by an Aircraft Borne Emulation Platform

Iwan Kruger
Dept. of Electrical and Electronic Engineering
Stellenbosch University
Stellenbosch, South Africa
iwankruger@gmail.com

Riaan Wolhuter
Dept. of Electrical and Electronic Engineering
Stellenbosch University
Stellenbosch, South Africa
wolhuter@sun.ac.za

Abstract—**The KU Leuven- and Stellenbosch Universities are jointly developing an electronically beam steerable phased antenna array for satellite applications, including all the peripheral ground- and space segment subsystems. This paper covers the development of an Aircraft based Satellite Emulator to facilitate convenient aircraft based testing of an antenna array, intended for Low Earth Orbit satellite deployment. A flight strategy is developed to emulate such a satellite as best possible, with the strategy subsequently implemented in software on in-flight PC hardware. The emulator acts as a full interface between the aircraft avionics and satellite bus system, to enable generation of the required antenna steering commands and to create a satellite bus image to the payload. The emulator provides in-flight systems information to the satellite payload, as it would get from an actual satellite bus during spaceflight. The emulator ensures indifference to the payload, regardless of the fact that testing is aircraft based. An embedded control algorithm for the steerable antenna has also been developed and resides in the onboard computer of the payload. Excellent initial test results have been obtained from the aircraft flight simulator and actual flight telemetry data, proving the viability and cost-effectiveness of the approach. The system tests as reported on here, stopped just short of full equipment flight testing, as scheduled for in the near future. This is awaited with keen interest, as all results up to the present have been positive and in line with expectations.**

Index Terms—**Satellite Emulator; Phase Array Antenna; Beam Steering; Orbital Calculations; Link Budgets;**

I. Introduction

The feasibility of utilising electronically beam steerable antenna arrays (SAA) in space, is currently jointly being investigated by the ESAT-TELEMIC division of the Department of Electrical Engineering, Katholieke Universiteit Leuven, (KUL) Flanders, in partnership with the Department of Electrical and Electronic Engineering, Stellenbosch University, South Africa. Successful space implementation of such an antenna array promises a number of significant benefits and the development of advanced techniques therefor, has been the subject of our joint research and development for some time [1], [2]. Amongst others, it will reduce the cost of ground stations by eliminating tracking antennas and reducing RF chain complexity, while still retaining an acceptable link budget. By introducing beam steered satellite antenna tracking of ground stations during overflight, the link budget could be improved and ground station complexity reduced, particularly with regard to antenna design and the RF chain. These ground nodes are typically used for environmental and agricultural data acquisition and any improvement regarding the above are always beneficial. The Stellenbosch University member of this partnership, is responsible for development of the satellite platform hosting the SAA, ground station and accompanying ground-space communications link. The payload would be deployed on a next South African low earth orbit (LEO) satellite. Development and construction of any form of space borne system is normally expensive and associated with many risks. It was, therefore, decided to introduce an interim testing phase prior to actual space flight, by using a light aircraft as a pseudosatellite test platform. This will offer the obvious advantages of convenient and relatively cheap closed loop system testing and debugging. This must, without doubt, enhance the chances of eventual in-flight success. The aircraft itself has been fitted out for experimental use and will house the SAA, the entire satellite payload containing the On Board Computer (OBC), communications link component chain, steerable antenna control/status interface, power supplies, as well as an Aircraft Satellite Emulator (ASE). In actual deployment, all interaction with the rest of the satellite is via a system bus for purposes of telecommand, telemetry and attitude/positioning information. The ASE is obviously required to act as translator and emulator between the payload and aircraft, the latter acting as pseudo-satellite. As far as the payload is concerned, it should behave as if connected to the actual satellite bus. The ASE and developed emulation strategy can be adapted to various LEO satellite payloads and thus provides a general low cost test platform prior to space deployment. The purpose of this paper is to report on the development of the emulation platform and SAA control subsystems. Some very encouraging results have been presented elsewhere [1], which have since been confirmed and expanded by continued pre-flight testing. The paper will describe the required flight path mechanics, the feasibility of an emulation strategy, a brief description of the implementation and test results obtained from an aircraft simulator and aircraft flight telemetry data. The rest of the paper is structured as follows: An overview of the relevant orbital mechanics in Section II, is followed by the discussion of the required emulation strategy in Section III. Calculation of the aircraft flight path parameters in order to satisfy the emulation strategy, is covered in Section IV. These basic operational requirements are then utilised for the systems design as set out in Section V. Evaluation and test results were obtained from simulations, a flight simulator and actual flight telemetry data. These are presented in Section VI. Section VII contains a summary of the work performed, results obtained and a view of the way forward.

II. Orbital Calculations

Before discussing the emulation strategy as developed, it might be useful to present a brief refresher on satellite orbital mechanics, as these are fundamental to the system design. The calculations presented in this section are based on a spherical earth model, which is adequate for this particular type of application [3]. The oblateness of the earth and the varying topography on the surface, are treated as coordinates above or below the spherical surface of the earth.

A. Angular velocity

The angular velocity of a satellite in orbit can be calculated by:

$$\omega \quad = \quad \sqrt{\frac{GM_E}{r^3}} \qquad (s^{-1}) \tag{1}$$

where r (m) is the circular orbit radius, the universal gravitation constant $G = 6.672 \times 10^{-11}\,\mathrm{m^3kg^{-1}s^{-2}}$, and the earth mass $M_E = 5.974 \times 10^{24}\,\mathrm{kg}$[4].

B. Coordinates

The locations of the satellite and ground stations are specified in latitude, longitude and radius coordinates, and can be expressed in celestial coordinates originating at the earth's centre, as per Figure 1.

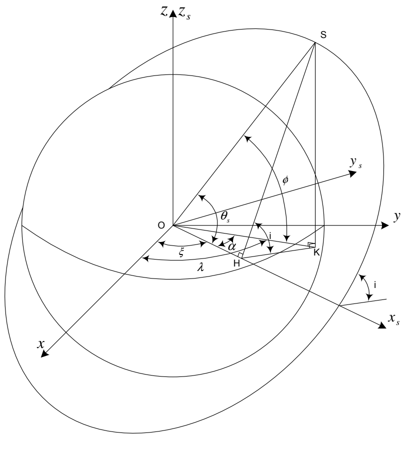


Fig. 1. Celestial coordinate structure

1) Satellite coordinates: The satellite position is first described by the (x_s, y_s, z_s) coordinate system (Figure 1). The x-axis is directed to the intersection of the satellite orbital path and the equatorial plane. The satellite coordinates in this coordinate frame are:

$$x_s \quad = \quad R\cos(\theta_s) \tag{2}$$

$$y_s \quad = \quad R\sin(\theta_s)\cos(i) \tag{3}$$

$$z_s \quad = \quad R\sin(\theta_s)\sin(i) \tag{4}$$

where i is the inclination angle and $\theta_s = \theta_0 + (\omega_{sat})(t)$ the orbit angle. The initial orbit angle can be calculated by:

$$\theta_0 \quad = \quad \arcsin\left(\frac{\sin(\phi_0)}{sin(i)}\right) \tag{5}$$

$$\tag{6}$$

where ϕ_0 is the initial latitude coordinate of the satellite and ω_{sat} the angular velocity.

A transformation is required to describe the (x_s, y_s, z_s) coordinates in a celestial coordinate system. The transformation is presented in the following matrix notation:

$$\begin{bmatrix} x_{Sat} \\ y_{Sat} \\ z_{Sat} \end{bmatrix} = \begin{bmatrix} \cos(\xi) & \sin(\xi) & 0 \\ -\sin(\xi) & \cos(\xi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_s \\ y_s \\ z_s \end{bmatrix} \tag{7}$$

The ξ angle is obtained as $\xi = \alpha - \lambda_{Sat,0}$ where $\lambda_{Sat,0}$ is the initial longitude coordinate of the satellite. α can be calculated as:

$$\alpha \quad = \quad \arccos\left(\frac{\cos(\theta_0)}{cos(\phi_0)}\right) \tag{8}$$

Thus, the satellite coordinates are given by:

$$x_{Sat} \quad = \quad R\cos(\theta_s)\cos(\xi) + R\sin(\theta_s)\cos(i)\sin(\xi) \tag{9}$$

$$\begin{aligned} y_{Sat} \quad &= \quad -R\cos(\theta_s)\sin(\xi) + \\ &= \quad R\sin(\theta_s)\cos(i)\cos(\xi) \end{aligned} \tag{10}$$

$$z_{Sat} \quad = \quad R\sin(\theta_s)\sin(i) \tag{11}$$

2) Ground station coordinates: From Figure 1, the ground station coordinates transformed to the celestial coordinate system, are defined by:

$$x_{GS} \quad = \quad R_{GS}\cos(\phi_{GS})\cos(\lambda_{GS}) \tag{12}$$

$$y_{GS} \quad = \quad R_{GS}\cos(\phi_{GS})\sin(\lambda_{GS}) \tag{13}$$

$$z_{GS} \quad = \quad R_{GS}\sin(\phi_{GS}) \tag{14}$$

with $\lambda_{GS}(t) = \lambda_{GS,0} + (\omega_{GS})(t)$, where $\lambda_{GS,0}$ is the initial longitude coordinate of the ground station and ω_{GS} the angular velocity of the Earth. [5]

C. Distance to satellite

If the coordinates of the satellite and ground station are known, the varying distance D between the ground station and the satellite can be obtained quite simply by Pythagorean geometry.

$$D = \sqrt{(x_{Sat} - x_{GS})^2 + (y_{Sat} - y_{GS})^2 + (z_{Sat} - z_{GS})^2} \tag{15}$$

D. Geocentric angle

The geocentric angle (ψ) between the satellite nadir point and a ground station placed at the centre of the earth, can be determined by the cosine rule.

$$\psi = \arccos\left(\frac{R_{GS}^2 + R_{sat}^2 - D^2}{2 \cdot R_{GS} \cdot R_{sat}}\right) \quad (16)$$

E. Elevation angle

The elevation angle (E) is the angle between the horizon and the satellite.

$$E = \left| \arcsin\left(\frac{R_{sat}\sin(\psi)}{D}\right) - \frac{\pi}{2} \right| \quad (17)$$

F. Azimuth angle

The azimuth angle is the angle measured Eastward from North, to the nadir point at the ground station, as per angle NPT in Figure 2.

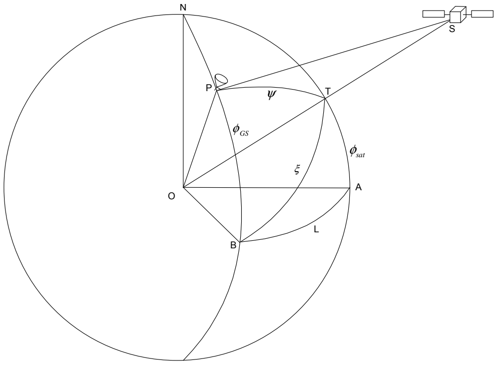


Fig. 2. Earth satellite geometry (reproduced from [6])

For the spherical triangle NPT of Figure 2:

$$\frac{\sin(NPT)}{\sin(90^\circ - \phi_{sat})} = \frac{\sin(NPT)}{\cos(\phi_{sat})} = \frac{\sin(PNT)}{sin(\psi)} \quad (18)$$

For the spherical triangle NBA the angle BAN and AON is equal to 90°, therefore

$$\frac{\sin(BNA)}{\sin(L)} = \frac{\sin(BAN)}{\sin(AON)} = 1 \quad (19)$$

Because angle BNA = PNT, equation 19 can be substituted into equation 18 with the result:

$$\sin(NPT) = \frac{\sin(L)\cos(\phi_{sat})}{\sin(\psi)} \quad (20)$$

$$a = NPT = \arcsin(\frac{\sin(L)\cos(\phi_{sat})}{\sin(\psi)}) \quad (21)$$

where ψ is the geocentric angle, ϕ_{sat} the latitude coordinate of the satellite and

$$L = |\lambda_{GS} - \lambda_{sat}| \quad (22)$$

the difference between the longitude coordinates of the ground station and the satellite. To obtain the true azimuth angle (A), we need to consider the position of the nadir, (point T in Figure 2) relative to the ground station (point P in Figure 2). The various cases can be summarised as follows:

$$A = \begin{cases} 180^\circ - a & \text{if } \lambda_{GS}(t) - \lambda_{sat}(t) > 0 \text{ and } \phi_{GS}(t) - \phi_{sat}(t) < 0 \\ a & \text{if } \lambda_{GS}(t) - \lambda_{sat}(t) > 0 \text{ and } \phi_{GS}(t) - \phi_{sat}(t) > 0 \\ 180^\circ + a & \text{if } \lambda_{GS}(t) - \lambda_{sat}(t) < 0 \text{ and } \phi_{GS}(t) - \phi_{sat}(t) < 0 \\ 360^\circ - a & \text{if } \lambda_{GS}(t) - \lambda_{sat}(t) < 0 \text{ and } \phi_{GS}(t) - \phi_{sat}(t) > 0 \end{cases} \quad (23)$$

G. Steering angles

This section will describe the basic strategy to calculate the ϕ (phi) and θ (theta) angles. These angles will enable the airborne object to track the specified ground station. The ϕ angle is measured from the positive x-axis of the body frame toward the positive y-axis, in the x-y plane. The θ angle is measured from the positive z-axis of the body frame, towards the position vector. Figure 3 defines the parameters used to calculate these angles.

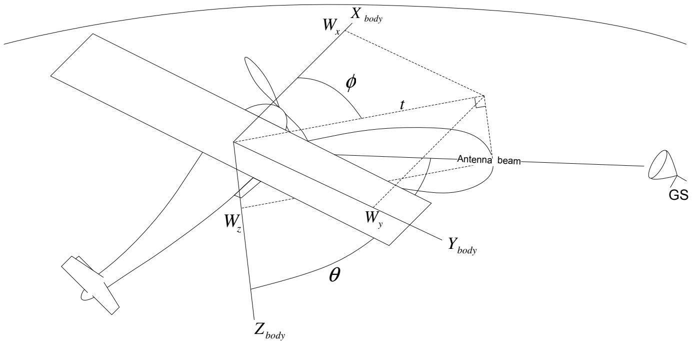


Fig. 3. Defining parameters used to calculate the θ and ϕ angles.

The coordinates (LLA), specified as latitude, longitude and altitude, are known for both the airborne object and ground station. The first step is to convert these coordinates from the LLA system to the ECEF frame, by making use of the World Geodetic System 1984 standard. (WGS-84)[7]. The results can be written as follows:

$$\vec{R}_{Sat} = \begin{bmatrix} X_{Sat} & Y_{Sat} & Z_{Sat} \end{bmatrix}^T and \quad (24)$$

$$\vec{R}_{GS} = \begin{bmatrix} X_{GS} & Y_{GS} & Z_{GS} \end{bmatrix}^T \quad (25)$$

The airborne object to ground station position vector is then calculated as

$$\vec{R}_{pos} = \vec{R}_{GS} - \vec{R}_{Sat} \quad (26)$$

The position vector can now be converted from the ECEF frame to the NED frame situated at the airborne object.

$$\vec{B} = \mathbf{K} \cdot \vec{R}_{pos} \quad (27)$$

where $\mathbf{K}$ is the transformation matrix [8].

It is necessary to take the attitude of the object into account by describing the object in terms of its pitch, roll and yaw Euler angles. By using an Euler 123 rotation [9] to transform the position vector from the NED frame to the body frame of the object, it is possible to describe the position vector from the object to the ground station in terms of the orientation of the object. The θ and ϕ angles can be calculated after accounting for the attitude of the object.

$$\vec{W} = \mathbf{A} \cdot \vec{B} \tag{28}$$

The θ and ϕ angles are relative to the body frame of the object. These angles direct the position vector to the target or ground station and can be calculated by means of simple trigonometry.

The theta θ angle is calculated as:

$$\begin{aligned} t &= \sqrt{W_x^2 + W_y^2} && (29) \\ \theta &= \frac{\pi}{2} - \arctan\left(\frac{W_z}{t}\right) && (30) \\ &= \frac{\pi}{2} - \arctan\left(\frac{W_z}{\sqrt{W_x^2 + W_y^2}}\right) && (31) \end{aligned}$$

The ϕ angle is calculated as:

$$\phi = \arctan\left(\frac{W_y}{W_x}\right) \tag{32}$$

III. Emulation Strategy

This section presents a few emulation strategies as considered and the reasons for selecting a particular one.

Two main emulation approaches were considered. The first approach was to emulate the satellite position relative to a ground station and the second to emulate the ground station position relative to a satellite. It is clear that for the first case the position could be described by elevation and azimuth angles [10]. The second approach describes the position of a ground station from the perspective of a satellite using the ϕ and θ angles. These angles are, in both cases, time dependent for LEO satellites. The function of the emulator is to calculate a flight route for an aircraft that would approximate these orbital characteristics as closely as possible.

A. Flight Strategy

1) The first emulation flight strategy considered was to fly in ascending concentric circles around a ground station. This strategy covers all the azimuth and ϕ angles for a specific elevation or θ angle. By spiralling upwards it is possible to cover many elevation and θ angles. The implementation of this strategy is however arduous. It is difficult for an aircraft to fly at a constant speed in an accurate, circular, upwards path around a ground station. However, the main disadvantage of this option is that the specific time variant behaviour of practical elevation-azimuth and ϕ-θ angles are not taken into account.
2) The second strategy entailed flying past a ground station in a straight path parallel to the earth surface at a constant speed and altitude. It is easier for a pilot to implement this strategy than the previous one and he will be able to maintain a more stable attitude. Because of this and with the aircraft flying parallel to the surface, the orientation of the antenna on the aircraft will match the predicted orientation of the antenna on the satellite more closely. The orientation of the antenna will enable the steering angles of the antenna to approach that of the actual satellite implementation, providing a more realistic scenario. The orientation will also facilitate the calculation of a more accurate linkbudget for a flight path. The linkbudget can then be emulated by compensating for the L_{FS} losses by attenuating the transmitting or receiving signal. The further advantage of this strategy is that the specific time variant behaviour of the elevation-azimuth and ϕ-θ angles of a LEO satellite are taken into account. This will also enable the relationship between the antenna steering angles and time to match that of the satellite application. For these reasons the second strategy is clearly the better one and was selected for actual implementation.

B. Transmission Link Strategy

With the aircraft based flight test, the direct LOS distance to a ground station is clearly much shorter than in the case of a real satellite. In order to emulate the satellite link budget, the free space loss (FSL) must be compensated for. The calculated FSL is shown in Figure 20. The aircraft link must, therefore, be attenuated to achieve the FSL of a satellite link. This could be simply done by adjustment of the transmit power for both the up- and down links.

Doppler shift is not a consideration for the aircraft flight test due to the low speed, but certainly affects the satellite link. The amount of required compensation will be determined by final orbit and receiver front-end selectivity bandwidth. For this project, Doppler compensation will probably be performed at the ground station and therefore, no compensation has been implemented on the emulator platform.

To minimise the affect of terrain scattering, the emulation flight tests are planned for a wide open semi-desert area.

Table I shows the losses caused by the distance between the ground station and the aircraft at various maximum elevation angles. Losses between the ground station and the satellite is displayed in the last column of Table I. The amount of attenuation can thus be calculated by subtracting the satellite free space losses from those of the aircraft at a specific altitude and maximum elevation angle.

IV. Calculation of Aircraft Parameters

In order to implement the chosen strategy as discussed in the previous section, it is necessary that the aircraft flight parameters be calculated in terms of the required trajectory. This calculation was done by means of a suitable script, based on the elevation-azimuth approach, as explained in Section III.

The script is fed with the maximum elevation angle as an input parameter. The maximum elevation angle occurs when the object is closest to the ground station. The script then calculates the time values for a LEO satellite in orbit, as it transits from minimum- to maximum elevation angle. An iterative method is implemented to calculate the parameters for the aircraft flight path, emulating the satellite elevation

Free space loss (dB)												
Altitude (km)	0.05	0.345	0.64	0.935	1.23	1.525	1.82	2.115	2.41	2.705	3	500
0 deg elevation	128.09	136.48	139.16	140.81	142	142.93	143.7	144.35	144.92	145.42	145.87	168.26
10 deg elevation	89.231	106	111.36	114.65	117.02	118.88	120.41	121.71	122.84	123.84	124.73	164.63
20 deg elevation	83.344	100.12	105.49	108.78	111.16	113.02	114.56	115.86	116.99	117.99	118.89	161.58
30 deg elevation	80.046	96.822	102.19	105.48	107.86	109.73	111.26	112.57	113.7	114.7	115.6	159.22
40 deg elevation	77.864	94.641	100.01	103.3	105.68	107.55	109.08	110.39	111.52	112.53	113.42	157.45
50 deg elevation	76.34	93.117	98.484	101.78	104.16	106.03	107.56	108.87	110	111	111.9	156.13
60 deg elevation	75.275	92.052	97.419	100.71	103.09	104.96	106.5	107.8	108.94	109.94	110.84	155.17
70 deg elevation	74.566	91.343	96.71	100	102.38	104.25	105.79	107.09	108.23	109.23	110.13	154.52
80 deg elevation	74.158	90.935	96.303	99.595	101.98	103.84	105.38	106.69	107.82	108.82	109.72	154.15
90 deg elevation	74.025	90.802	96.17	99.462	101.84	103.71	105.25	106.55	107.69	108.69	109.59	154.03

TABLE I
FREE SPACE LOSS CALCULATIONS

time window. An elevation time window is calculated for each combination of aircraft altitude and speed. It should be noted that if the altitude changes, so does the minimum distance to the ground station, which is the distance from the ground station to the satellite nadir point, when the aircraft is closest to the ground station.

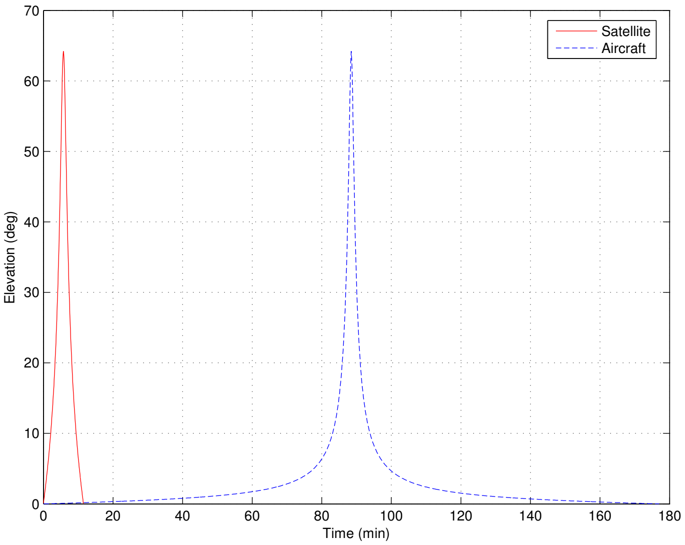


Fig. 4. Elevation angle versus time

Figure 4 compares the satellite and aircraft elevation angles versus time, for a chosen flight path. It is clear from Figure 4 that only a small interval of the visibility time period of an aircraft is suitable to emulate the behaviour of the time varying elevation angles of a satellite. For this reason the graph of the aircraft flight path is shifted to the left, to align the maximum elevation angles. The optimisation of the elevation time graph is achieved by calculating the area under each graph for a specified time window and subtracting the results. The smaller the result after the subtraction, the better the match between the two graphs. The results of these calculations are obtained from the script in the form of two figures. The dark blue areas in Figure 5 specify the areas where the speed and altitude of the aircraft best conform to the emulated elevation and azimuth angles. Figure 6 shows the distance for different altitudes from the aircraft's nadir point to the ground station at the point when the aircraft is closest to the ground station. Figures 5 and 6 enable us to choose either the desired speed, altitude or distance from the ground station and then use the figures to calculate the other parameters. Therefore, using the results from Figures 5 and 6, will allow us to specify the final flight route as required.

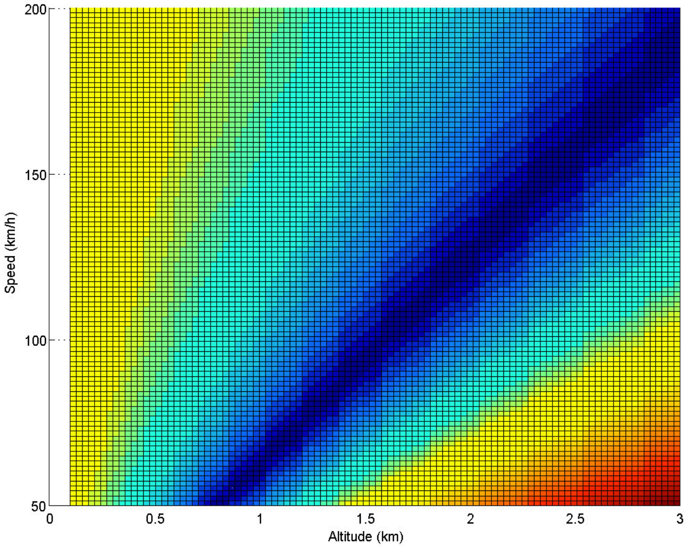


Fig. 5. Speed versus altitude of aircraft

It is thus possible to calculate aircraft flight path parameters of speed, altitude and distance from the ground station to satisfy the elevation and azimuth angles as related to the satellite's orbital flight. Figure 7 and Figure 8 illustrate this conformity between the elevation and azimuth angles of the satellite and the aircraft for the visibility time period of the satellite. Although deviations will occur at low elevation angles, a very useful time window for testing purposes can still be obtained.

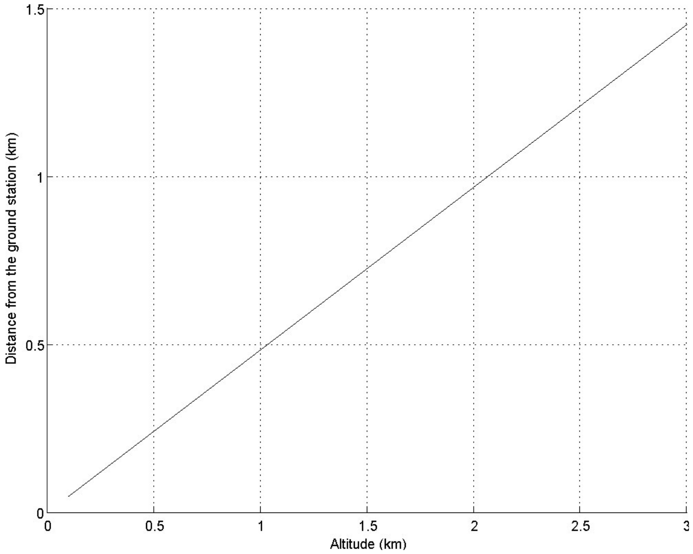


Fig. 6. Distance from ground station

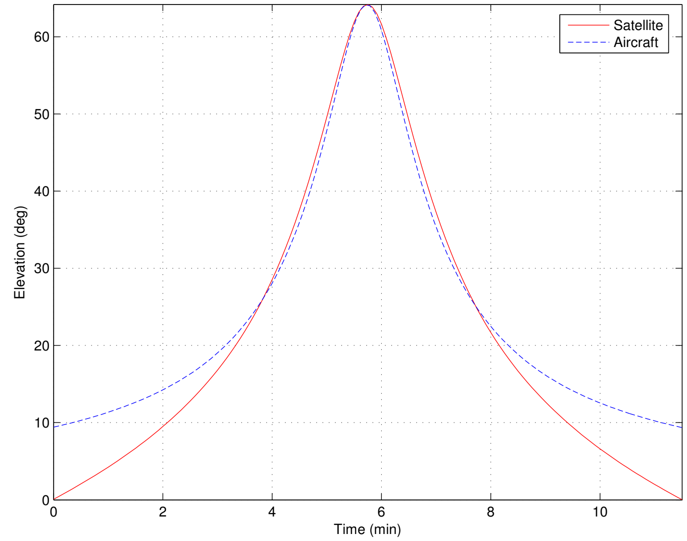


Fig. 7. Elevation angle versus time interval

V. System Design

The system design is comprised of two sections, the emulator and the satellite payload modules. Only the components indicated in the system diagram as per Figure 9, have a direct influence on the system design. Components not directly affecting the system, are not shown.

A. Satellite Payload Module

The following components on the payload are applicable to the system design:

- Steerable antenna developed by KUL.
- The OBC, an SH4 processor with a 32-bit RISC architecture.

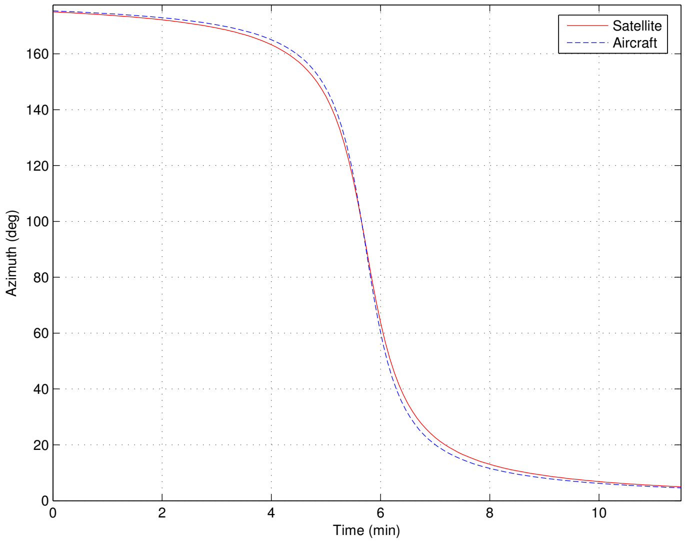


Fig. 8. Azimuth angle versus time interval

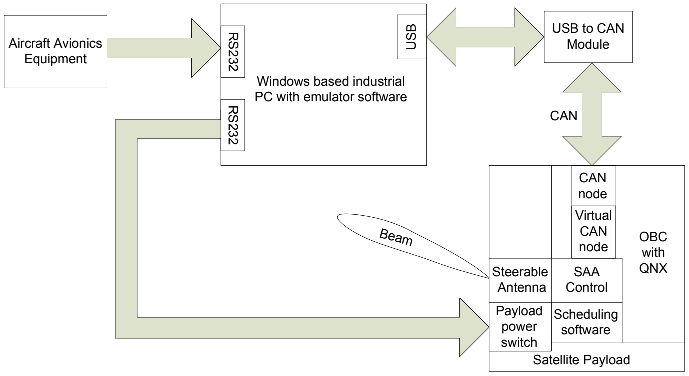


Fig. 9. System diagram

- A QNX operating system is run on the OBC. This UNIX based real-time operating system was provided by SunSpace, with additional software components.
- The scheduling module comprises software that schedules the communication times between the satellite and various ground stations.
- An SAA control module electronically controls the steerable antenna array. The required control algorithm was developed in the course of this project. The module uses the ϕ and θ steering angles to direct the antenna beam, according to the strategy to calculate these angles as discussed earlier. The SAA module is housed in the OBC of the payload.
- The CAN node, which allows devices to connect to the CAN network.
- A virtual CAN node enables the SAA control software to use the hardware of the exsisting CAN node on the OBC to communicate over the CAN bus.

B. Emulator Module

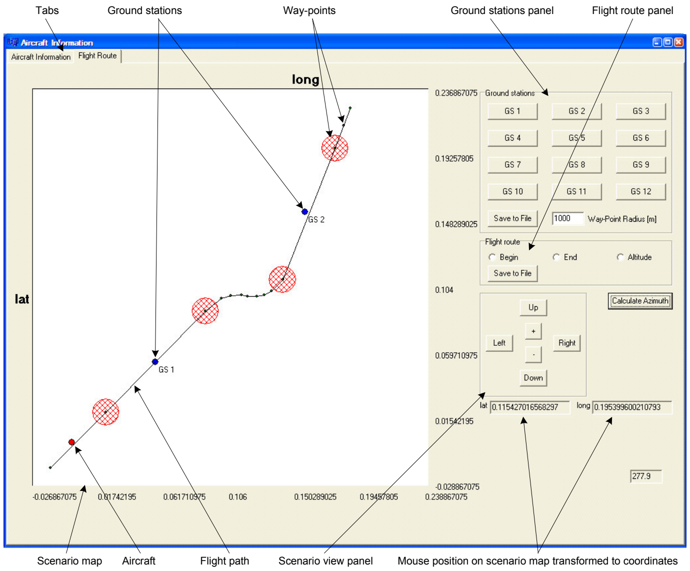
Instead of connecting the payload to a satellite through the CAN bus, the payload is connected to the emulator module, which mimics the behaviour of an actual satellite. The emulator module comprises an industrial PC with emulator software, known as the ASE, which provides an interface for the user to construct a flight path for an airborne object emulating the orbital characteristics of a satellite pass as closely as possible. As the emulation strategy is to fly the payload with the emulator module on an aircraft, the emulator module connects to the aircraft avionics equipment. Just as the satellite would provide data to the payload in flight, the emulator will provide the necessary data.

The aircraft satellite emulator design is divided into three sections, i.e., the aircraft avionics equipment interface, the ASE - payload interface and the satellite emulator software running on the industrial PC.

The ASE is connected to the aircraft avionics equipment via a RS232 port and to the payload CAN bus, via a USB to CAN module. The ASE software functions as implemented, can be summarised as follows:

1) One of the main tasks of the satellite emulator software, is to provide a GUI that will allow the user to construct a flight path for an aircraft to emulate a satellite, store flight data and display results. A GUI constructed flight path, with way points indicating the start- and end of the path, is shown in Figure 10. A GUI will allow a user to specify the ground station coordinates, maximum elevation angle and aircraft speed. The application will then calculate the speed, altitude and distance from the ground station required from the aircraft flight to satisfy the elevation and azimuth angles. The elevation and azimuth angles for the visibility time period as calculated, are displayed to the user, assisting with flight path definition.
2) The ASE software also calculates all the line-of-sight (LOS) link margins, elevation and azimuth angles of a satellite with specific parameters, for a specified amount of time.
3) The ASE software accepts the aircraft avionics inputs and does the necessary in-flight realtime translation to provide emulated data to the payload OBC via CAN bus. Flight data are recorded and used to calculate the aircraft LOS link margins, elevation and azimuth angles. The results of these calculations will be displayed to the user (typically an aircraft engineer/passenger), enabling him to to evaluate the flight continuously.

   The free space loss (FSL) parameter needs to be compensated for to emulate the satellite linkbudget. Figure 20 shows the calculated FSL over flight time. The aircraft link must thus be attenuated to obtain the correct satellite link FSL.
4) All the data received and sent by the ASE are logged by a script written in C. The SAA steering angles calculated by the control algorithm are also logged. This will enable the evaluation of the system performance after a flight. The SAA control can be thus be evaluated by perusing the logged data.
5) Close coordination was kept with the software developers of the OBC payload software, to ensure data/file compatibility and realtime data packet synchronisation between ASE and OBC/CAN bus.



1000 Way-Point Radius [m]

lat 0.115427016568297 long 0.195399600210793

277.9

Fig. 10. Screenshot of the flight route tab of the GUI

C. Functional analysis

An understanding of the system can be achieved by discussing the various hardware and software modules, as per the system's functional block diagram of Figure 11.

D. Concept of Execution

1) Use Cases: All the entities that interact with this system and their corresponding use cases are shown in Figure 12. The emulation scenario is set up in the first use case. This is done by the user creating a flight route. A map with the flight route scenario is then displayed to the user. The second use case demonstrates the peripheral devices of the emulator being initialised by the user. These peripheral devices include the interface of ASE with the aircraft avionics equipment and the payload. In use case three data is received by the system from the aircraft avionics equipment. The data received describes the attitude and position of the aircraft. The system switches the payload ”on” when the aircraft reaches the start of the flight route and ”off” when the aircraft reaches the end of the flight route. This is illustrated by use case four, which simulates the situations as the satellite switches the payload ”on” when communication with a ground station should start and ”off” when communication ends. In use case five, QNX starts the execution of the antenna control software on board the SH4. This will occur each time the payload is switched on and the SH4 boots. In use case six the scheduling software

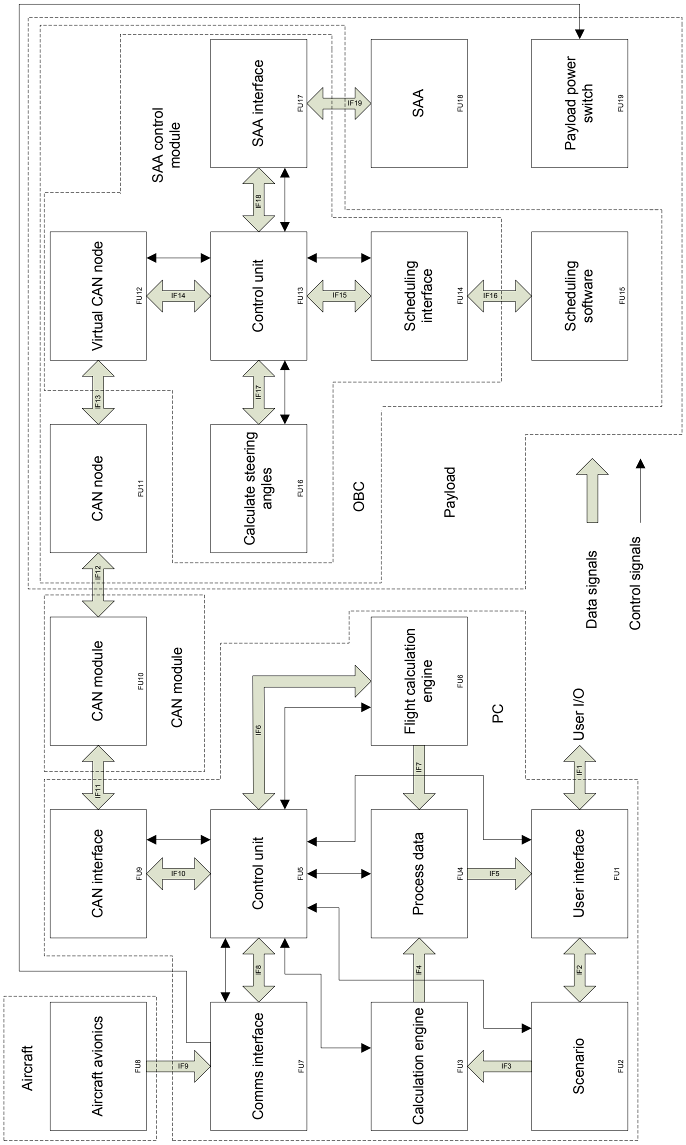


Fig. 11. Functional block diagram of the system

provides the coordinates of the target ground station to the system. The steering angles of the antenna are then calculated and the corresponding commands sent to the steerable antenna.

2) Sequencing: The sequential interaction of the various components defined by the use cases are illustrated by the sequence diagram of Figure 13. Note the following:

- The ASE continuously receives aircraft data from the aircraft avionics equipment.
- The ASE switches the payload on once the aircraft is at the start of the flight path. By switching the payload on, the steerable antenna will activate and QNX on the SH4 will boot. QNX will then start the execution of the scheduling and SAA control software. However, note that when the aircraft reaches the end of its flight path the ASE will switch the payload off. This will power down the steerable antenna, as well as QNX and the applications running on it.

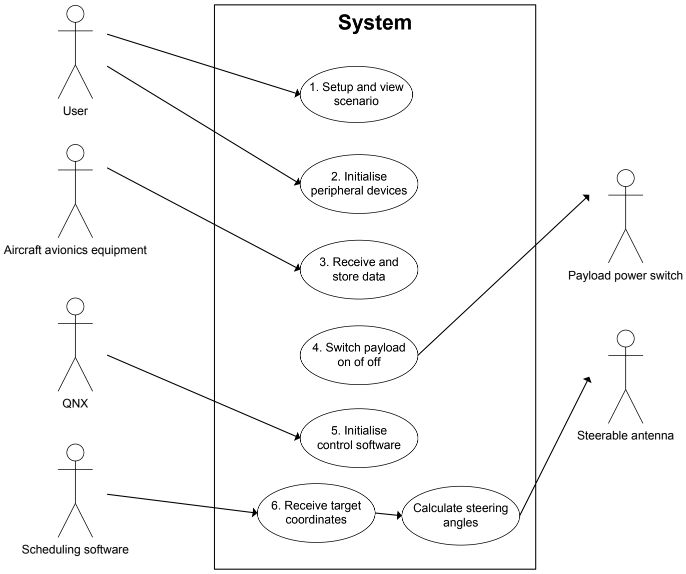

System
User
Aircraft avionics equipment
QNX
Scheduling software
1. Setup and view scenario
2. Initialise peripheral devices
3. Receive and store data
4. Switch payload on of off
5. Initialise control software
6. Receive target coordinates
Calculate steering angles
Payload power switch
Steerable antenna

Fig. 12. System use case diagram

E. Transmission Link

Subsection III-B describes the transmission link strategy adopted. The main concepts of this strategy are that:

- FSL should be compensated for to emulate the transmission link.
- The FSL compensation is achieved by attenuation of the received or transmitted signal.
- The attenuation will be applied at the ground station.
- For reasons mentioned in Section III-B, Doppler compensation is not considered. Doppler compensation will, however, be applied at the ground station when the system is implemented on an actual satellite.
- Flight tests will be performed in wide-open areas to minimise the effect of terrain scattering.

VI. Evaluation

In order to verify the design and subsequent usability of the system, some preliminary tests were performed to evaluate the emulator system performance before an actual flight test. The overall system performance was evaluated in four different ways, first with an aircraft avionics equipment emulator (AAEE), which was specifically developed for the evaluation of this system. The emulator serves as a substitute for an actual aircraft. Secondly, performance is evaluated with an aircraft simulator. The aircraft simulator closely models the light aircraft for which this system was developed. The aircraft simulator has input interfaces, such as a joystick and pedals, that contribute unknown factors such as pilot skill to the system evaluation. The third preliminary test used the aircraft simulator in conjunction with a prototype SAA to evaluate the system performance. The final ground based system test performed last, used actual aircraft telemetry data. This data

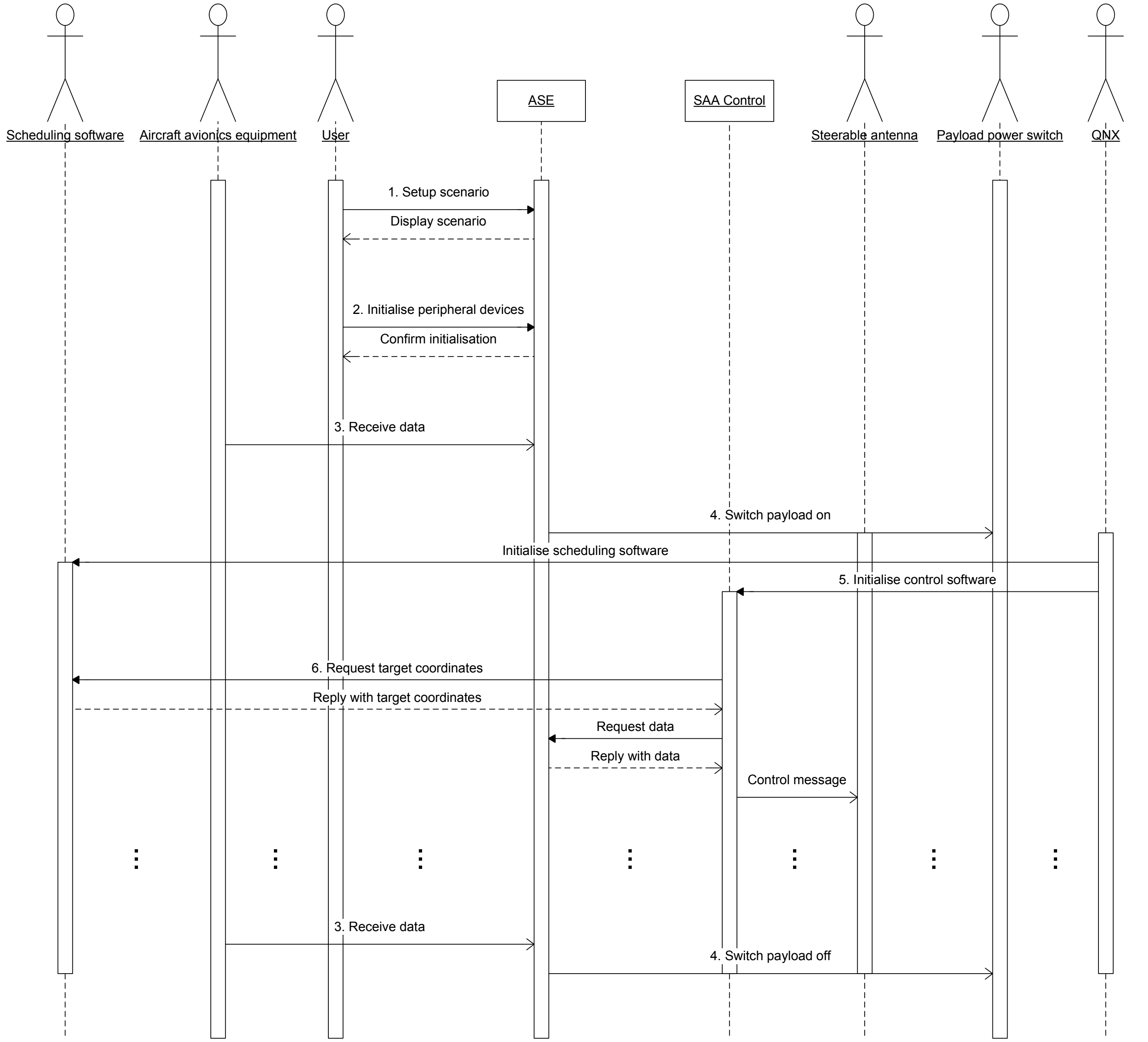


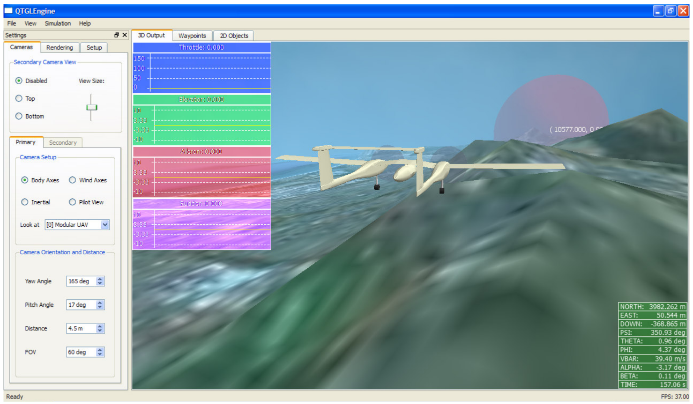
Fig. 13. System sequence diagram

was captured on a flight of the light aircraft for which this system has been developed. The tests were performed as follows:

1) During the course of the project, the AAEE software was specifically developed as part of the research. The software provides realistic aircraft avionics data with the correct protocol to the system, thereby providing a means to test the system without an actual flight test.
The user can specify a chosen flight path by specifying the flight start-and-end coordinates as well as aircraft parameters such as velocity, altitude and attitude. The AAEE software uses this information to provide aircraft avionics data to simulate an aircraft flight.
As a first attempt, the AAEE was used to produce realistic avionics flight data, as for an envisaged test flight. This was run on a separate CPU and the data was serially fed into the ASE in real time. The ASE output, in reaction to the flight path as configured and simulated data as received, was fed onto the payload CAN bus for processing by the OBC. The latter was to produce the required steering commands for the SAA. The required steering angles, and commensurate commands, were calculated in advance using the orbital calculations as discussed earlier. The entire process was

logged by means of suitable scripts and the output of the payload to the SAA was captured on a separate PC.

2) As a second round of more stringent tests, approaching practical flight, a hardware based, standalone aircraft flight simulator was coupled to the ASE. The simulator is used for training and automatic pilot development. It is controlled by a joystick, pedals and other realistic hard interfaces, thus contributing the unknown quantity of pilot skill. Because the simulator closely models the light aircraft on which this system will be flown, the simulation will react in much the same way as the actual aircraft in flight. The simulator thus presents an extra dimension to the evaluation, without an actual flight test. Figure 14 shows a screen shot of the simulator GUI.



Secondary Camera View
● Disabled
○ Top
○ Bottom
View Size:

Camera Setup
● Body Axes
○ Wind Axes
○ Inertial
○ Pilot View
Look at [0] Modular UAV

Camera Orientation and Distance
Yaw Angle 165 deg
Pitch Angle 17 deg
Distance 4.5 m
FOV 60 deg

Throttle: 0.000
Elevator: 0.0000
Aileron: 0.000
Rudder: 0.0000

NORTH: 3982.262 m
EAST: 50.544 m
DOWN: -368.865 m
PSI: 350.93 deg
THETA: 0.96 deg
PHI: 4.37 deg
VBAR: 39.40 m/s
ALPHA: -3.17 deg
BETA: 0.11 deg
TIME: 157.06 s

Ready
FPS: 37.00

Fig. 14. Screen shot of the aircraft simulator software GUI

The ASE software was used to construct a flight route with a maximum elevation angle of 60°. An above ground station altitude of $1.869\,\mathrm{km}$ is necessary for an aircraft cruising speed of $120\,\mathrm{km/h}$. As mentioned in Section III, the flight strategy would be to fly at a constant speed and altitude past a ground station, thereby emulating a LEO satellite pass. The results of this test are shown in Figures 15 to 20.

These figures present three sets of data. (For sake of clarity, some smaller portions of the graphs are magnified) The first two sets contain the predicted satellite and aircraft data. These were calculated with the help of calculations derived in Section II. The third set shows the aircraft flight data as generated by a simulated aircraft flight.

The attitude of the aircraft is described by Figure 15. The figure indicates the difficulty experienced in this case, in maintaining a constant attitude. It should be mentioned that the simulator was not flown on autopilot.

Figures 16 and 17 show the calculated elevation and azimuth angles. As can be seen from these two figures, the simulated flight data closely resembles that of the predicted aircraft flight. The elevation and azimuth angles are not that affected by the attitude, which is in line with the theoretical findings as presented earlier.

The calculated θ and ϕ angles are shown in Figures 18 and 19. A slight deviation is seen from the predicted aircraft data, which is attributed to the varying attitude of the aircraft. It is because of this expected deviation that the first approach mentioned in Section III was not chosen. A better approach is to emulate the elevation-azimuth angles more closely than the θ-ϕ angles, which will vary in any case. The antenna control algorithm uses these θ and ϕ angles to beam steer when the antenna steering control algorithm compensates for this varying attitude, as it should. The varying attitude should not affect the link budget significantly.

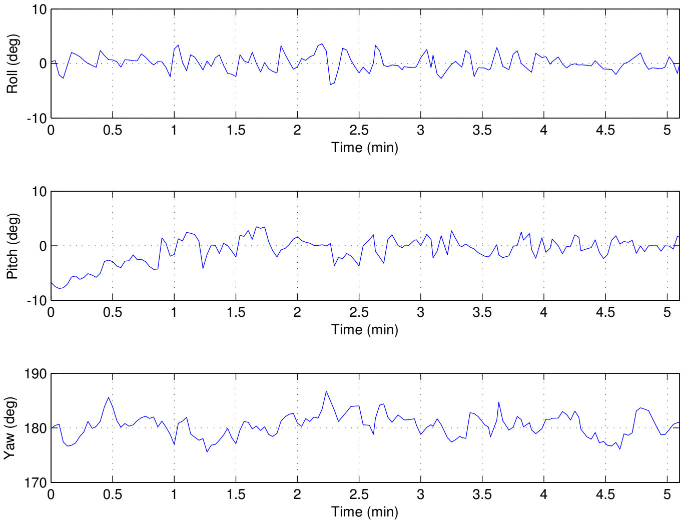


Fig. 15. Measured aircraft simulator flight test roll, pitch and yaw data

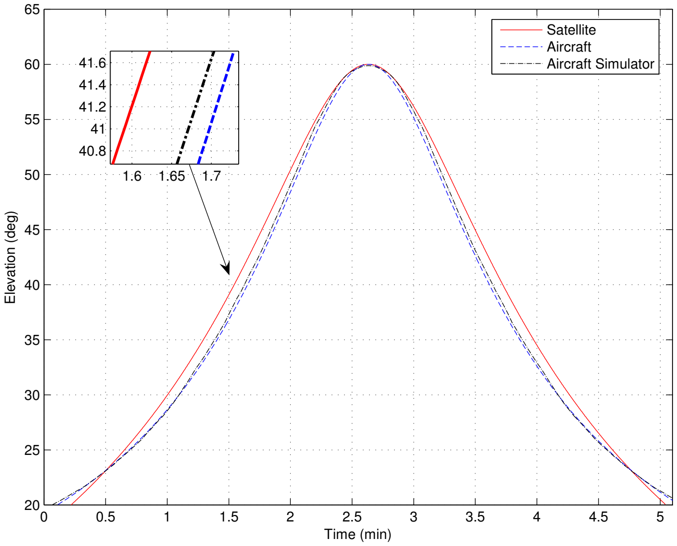


Fig. 16. Calculated satellite-, predicted aircraft- and measured aircraft simulator flight test elevation data

3) The following test was performed to evaluate the performance of the developed system with a prototype SAA, where the main focus was to see whether the two systems communicated correctly with each other. The

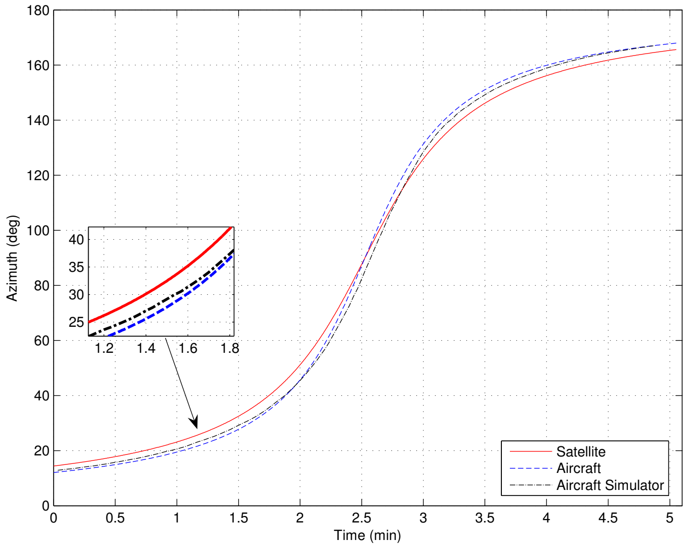


Fig. 17. Calculated satellite-, predicted aircraft- and measured aircraft simulator flight test azimuth data

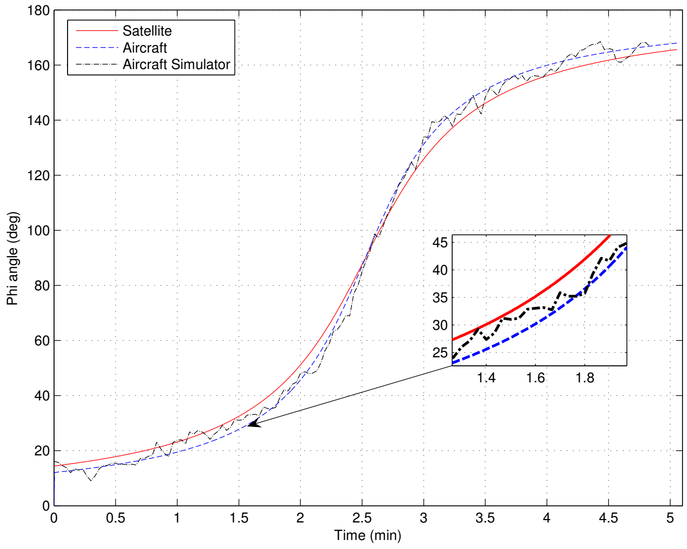


Fig. 19. Calculated satellite-, predicted aircraft- and measured aircraft simulator flight test ϕ angle data

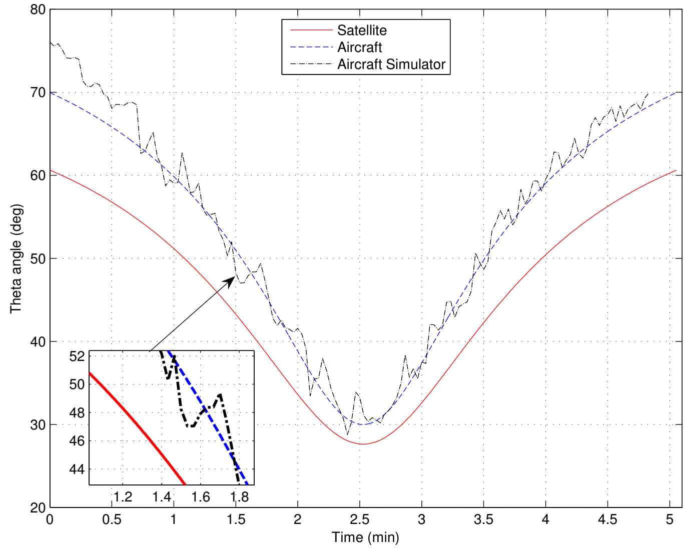


Fig. 18. Calculated satellite-, predicted aircraft- and measured aircraft simulator flight test θ angle data

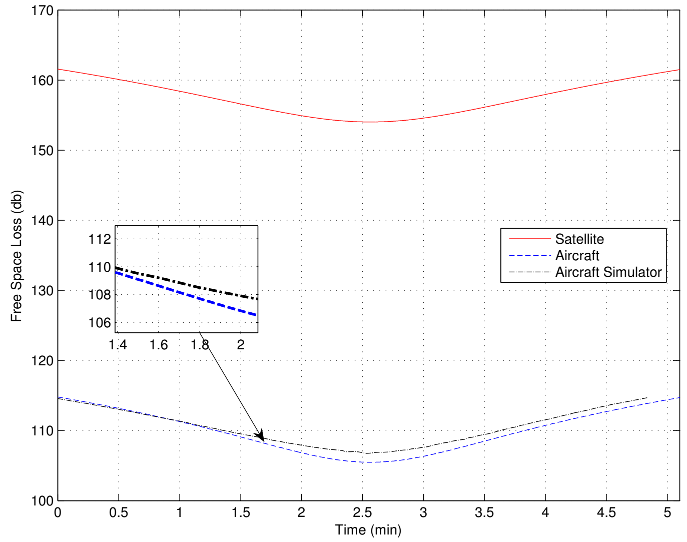


Fig. 20. Calculated satellite-, predicted aircraft- and flight simulator test FSL data

tests were performed in conjunction with KUL who, as mentioned earlier, developed the SAA.

The objective of the final test was first to confirm that the SAA control software on the payload can communicate with the SAA, then to verify that the SAA control software steers the SAA in the correct direction. To confirm the latter, the aircraft simulator was coupled to the system, a ground station was selected and the specified flight route flown.

A signal generator placed directly in front of the SAA provided a constant signal source to the SAA. A reduction of the received signal occurs when the SAA is steered. The SAA is steered according to the ASE indication of the aircraft on the flight route. Therefore, if the aircraft is directly above the ground station, no phase shift will be applied and the received signal will be at its strongest.

Data captured from the SAA confirmed that the SAA was indeed steered correctly. The SAA control module on the payload controlled the SAA as expected to enable the SAA to phase shift the signal correctly in order to direct the antenna beam to a specific ground station. KUL reviewed the data captured from the SAA and also concluded that the SAA was steered according to the specified flight path, confirming that communication between the payload and SAA was correct.

4) The final system test once more increased the level of realism, before an actual flight test. The system

test utilised captured telemetry data from a flight of the light aircraft test vehicle. Actual telemetry data introduces another unknown factor into the evaluation process, in the form of wind disturbances and other flight perturbations. The data generated by the system was captured and evaluated, as with the previous tests. The test proceeded as follows:

A flight path was first constructed with the help of the ASE software. The path past a ground station had a maximum elevation angle of $50°$, an altitude of $1.985\,\mathrm{km}$ above the ground station and required a cruising speed of $130\,\mathrm{km/h}$. The flight path was then flown with a light aircraft. The aircraft telemetry data was captured in flight and retrieved afterwards. Note however, that only half of the flight path constructed by the ASE, was flown. The reason for this is that the data for the second half of the flight path would be a mirror image of the first, which was therefore, considered adequate for these initial system tests. The light aircraft used for the flight, is depicted in Figure 21.

Fig. 21. The Jora flight test aircraft

The following four figures describe the telemetry data captured from the flight. The first set of data illustrated in Figure 22 shows the aircraft flight route as the aircraft flew in a North-Westerly direction past the ground station. Figure 23 shows the aircraft altitude above the ground station, Figure 24 the aircraft airspeed and Figure 25 the attitude. Note that Figure 24 illustrates the speed of the aircraft through the air and not the speed relative to the ground. Factors such as the wind needs to be taken into account when calculating the aircraft ground speed.

As depicted by the figures regarding the aircraft simulator based system test, these figures illustrate the difficulty experienced by the pilot in maintaining a constant heading, altitude, speed and attitude. These deviations can also partly be attributed to external factors such as wind disturbances. However, note that the attitude with these wind disturbances are more or less the same as the attitude obtained earlier with the aircraft simulator in Figure 15. These disturbances could possibly be minimised by an experienced pilot flying in a relatively calm day.

After the telemetry data was retrieved, the flight data was fed to the ground based system test setup. This setup is exactly the same as for the previous system tests, except that the input data is sourced from the actual aircraft flight telemetry record. The results of this test are shown in Figures 26 to 29.

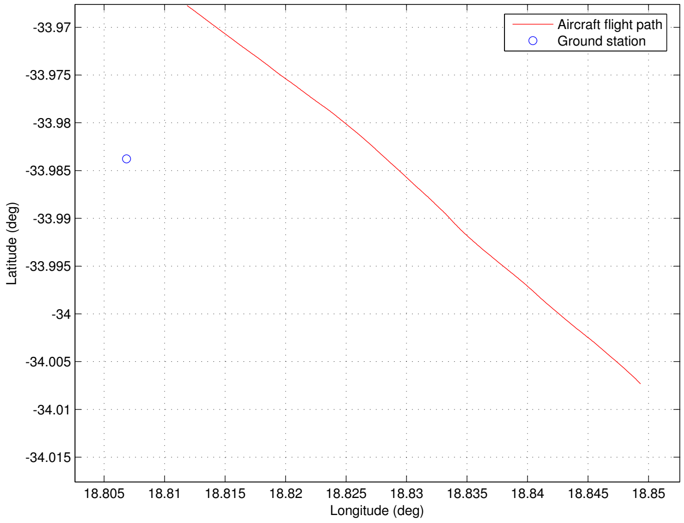


Fig. 22. Aircraft flight path past ground station

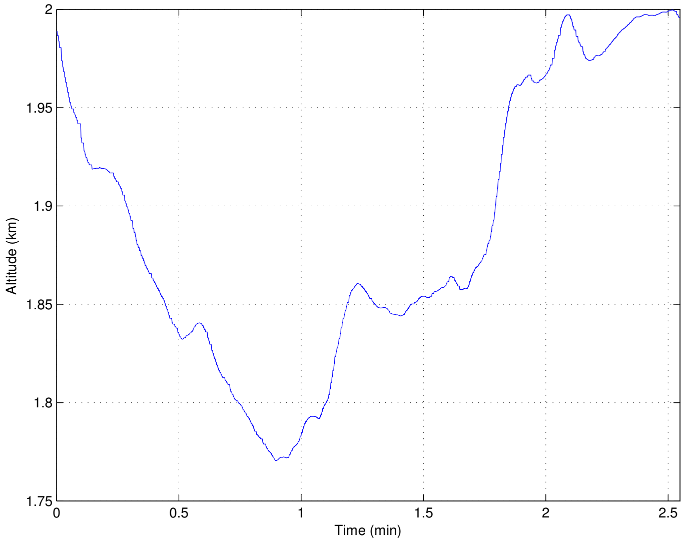


Fig. 23. Aircraft altitude above ground station

The elevation and azimuth angles are shown in Figure 26 and 27. It is clear from these results that the actual elevation and azimuth angles closely follow the predicted values, as in the case using the aircraft simulator. This is in line with the results from previous system tests and a gratifying confirmation thereof.

Figure 28 and 29 shows the θ and ϕ steering angles. These angles are also in line with the results obtained by the earlier system test. Similar to the indications of the earlier test, small deviations from the predicted aircraft data can be seen, attributed to varying altitude, speed and attitude. This data however, also shows that these

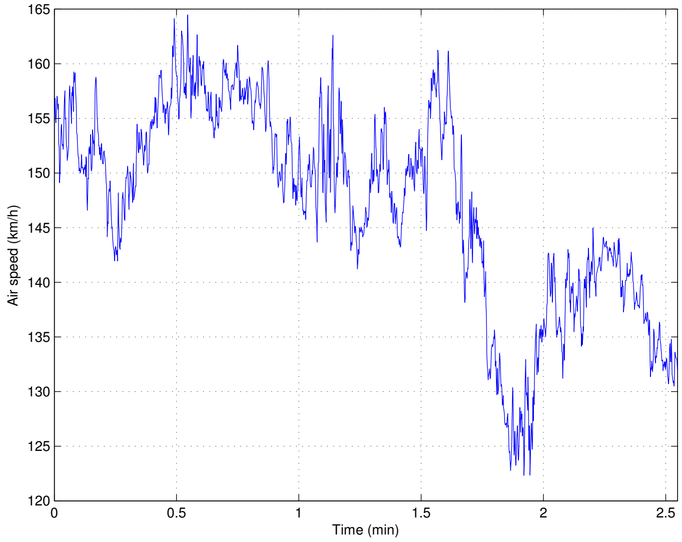


Fig. 24. Aircraft air speed

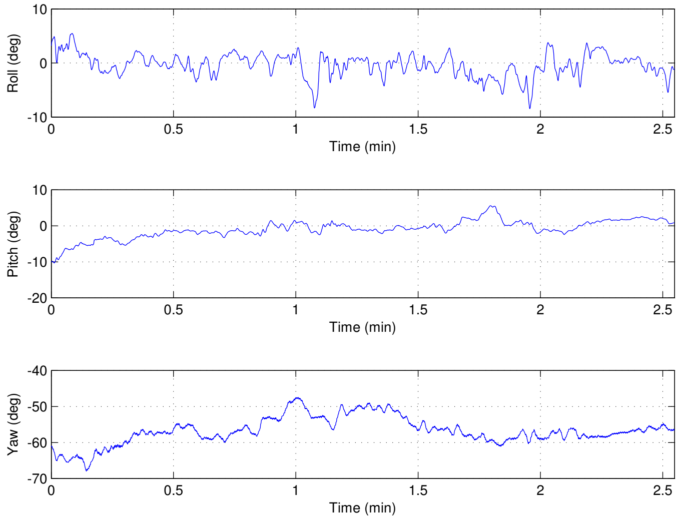


Fig. 25. Aircraft attitude

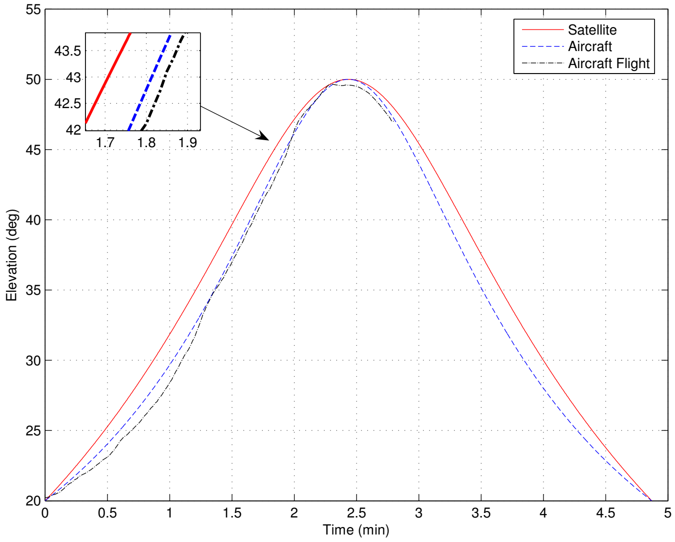


Fig. 26. Calculated satellite-, predicted aircraft- and measured aircraft flight test elevation data

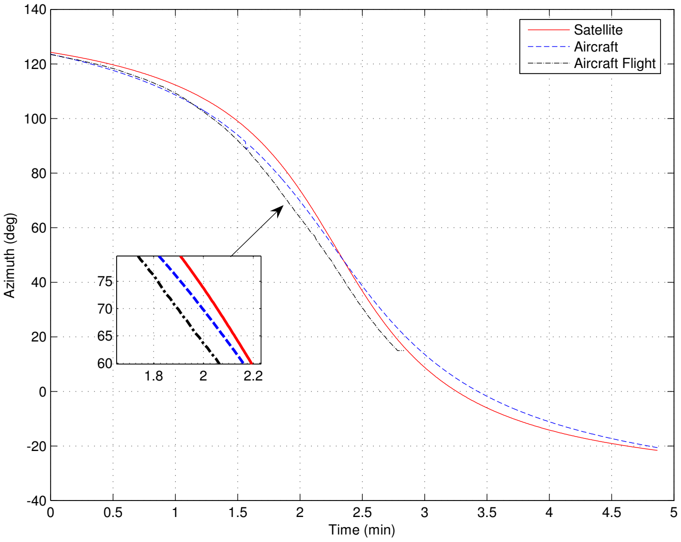


Fig. 27. Calculated satellite-, predicted aircraft- and measured aircraft flight test azimuth data

deviations are more or less the same as experienced with the earlier aircraft simulator based tests. It also serves to further confirm the soundness of concept using an appropriate aircraft based emulator as an interim test bed for this type of satellite system.

Figure 30 shows the calculated FSL. The losses resemble the predicted figures and the link budget will not be significantly affected by the aircraft attitude, due to the steerability of the antenna.

The tests as abovementioned, have proved the functionality of the design and no reason that the design should not perform as expected in the actual flight test, has been uncovered. The results obtained by using actual aircraft telemetry data confirmed those obtained previously with the aircraft simulator. At the time of of writing, an actual flight test with fully integrated system was scheduled for the second half of 2011. Equipment racks in the aircraft have already been prepared to receive the ASA, SAA and test platform hardware.

VII. Conclusion

The initial results obtained by means of an aircraft simulator and flight telemetry, confirm the requirement to follow an exact flight route while maintaining a constant attitude. Any deviation will affect the performance of the emulation. However, the elevation and azimuth angles are less affected than the θ and ϕ angles by small aircraft attitude fluctuations, for reason that the elevation and azimuth angles are measured from the perspective of the ground station and θ and ϕ from the aircraft.

Attitude changes obviously also change the orientation of

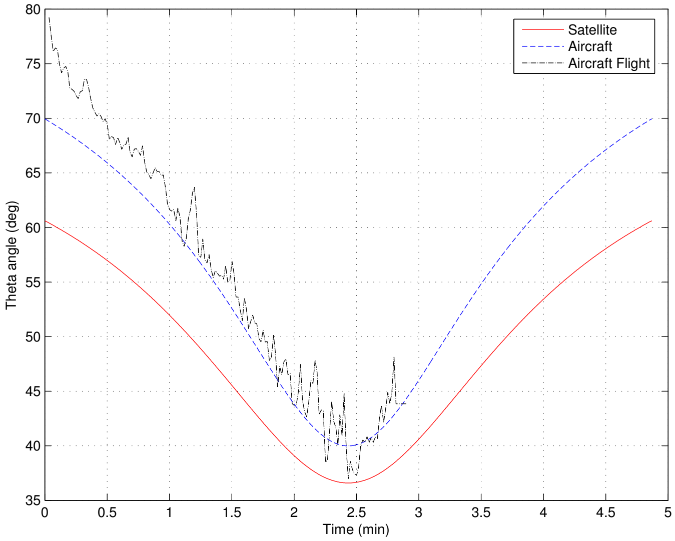


Fig. 28. Calculated satellite-, predicted aircraft- and measured aircraft flight test θ angle data

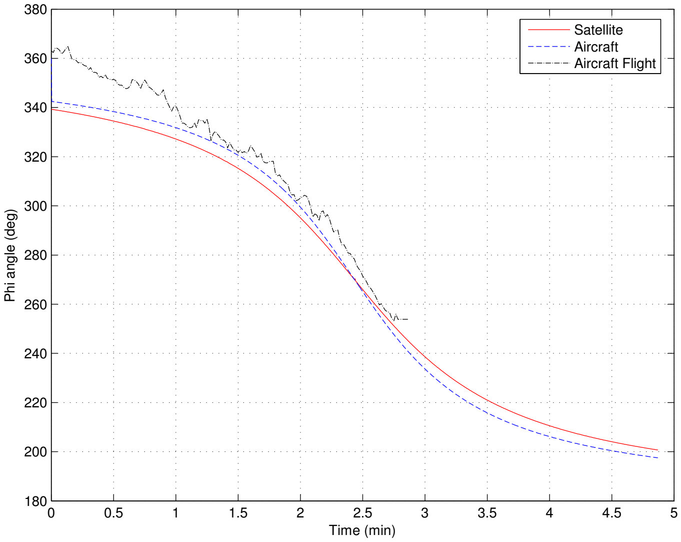


Fig. 29. Calculated satellite-, predicted aircraft- and measured aircraft flight test ϕ angle data

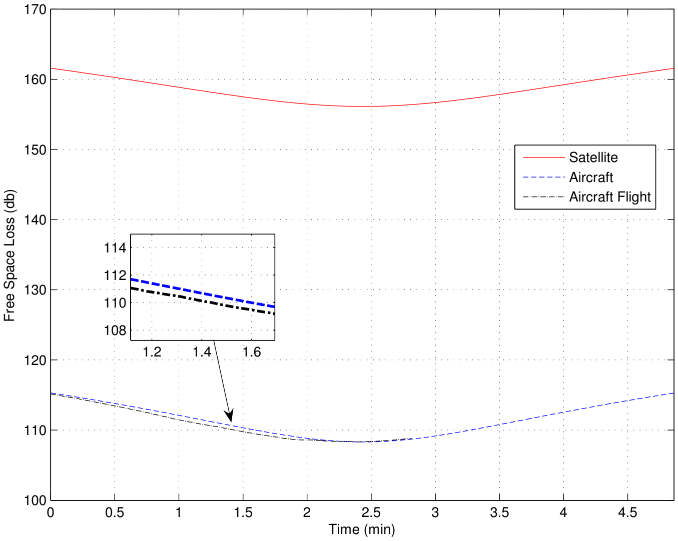


Fig. 30. Calculated satellite-, predicted aircraft- and flight test FSL data

the antenna on the aircraft. This, however, will be compensated for by the antenna control algorithm in correcting the θ and ϕ steering angles (Figures 18, 28 and 19, 29). The effect on the transmission link is, therefore, relatively small, apart from some required change in the induced FSL (Figure 20).

Maintaining an exact flight path presented a challenge to the pilots flying the aircraft simulator and the actual light aircraft. An experienced pilot is essential for the successful implementation of the flight strategy. This was even more evident in the actual flight where external factors such as wind disturbances played a role. These external influences can be minimised by flying in good weather conditions and augmenting the flight operations with an autopilot.

Furthermore, results from the previous sections indicate that deviations occur at low elevation angles. A flight path with a higher maximum elevation angle will, therefore, emulate the satellite more accurately. This is not seen as a major drawback, as the emulation time window is still adequate.

Test logs from the performed tests also indicate that the beam steering commands were correctly generated by the OBC in response to the ASE inputs, in response to the ASE inputs as processed from the aircraft avionics parameters, flight path data as defined and ground station coordinates.

The initial results clearly indicate that it is quite feasible to use a light aircraft equipped with a suitable emulator, to act as an initial flight test platform, in order to evaluate the performance of the beam steerable satellite antenna and peripheral associated payload components. Although many other factors, such as overall hardware configuration, space endurance etc., enter into the design of actual space flight hardware, the economics and convenience of the approach as set out, are beyond question. It was also proved that an actual LEO satellite flight path could be emulated to an acceptable degree. The ASE with incumbent software as developed, will furthermore act as a realistic and flexible interface between the aircraft and the satellite payload under development. The set of results obtained thus far using actual hardware and very realistic flight data, confirmed the accuracy and functionality of the ASE. The tests performed with the prototype SAA further proved the functionality of the interface between the two systems and the correct operation of the OBC-SAA control software. This paper described the implementation and intermediate testing of a practical and quite general, LEO aircraft based emulation platform. This development is not only suitable for the SAA payload in question, but can be adapted for interim testing of various satellite payloads. Such an approach is a flexible and clearly cost effective means of actual preflight system testing. The system tests as documented stopped just short of full airborne equipment flight testing, as scheduled for in the near future. However, as results up to the present have been very positive and in line with expectations, we look forward to that final step.

References

[1] I. Kruger and R. Wolhuter, “An aircraft based emulation platform for LEO satellite antenna beam steering,” in *Fifth International Conference on Systems and Networks Communication (ICSNC 2010)*. Nice, France: International Academy, Research, and Industry Association (IARIA), 22 - 27 August 2010, pp. 221 – 228.

[2] W. Aerts, P. Delmotte, and G. Vandenbosch, “Conceptual study of analog baseband beam forming: Design and measurement of an eight-by-eight phased array,” *IEEE Transactions on Antennas and Propagation*, vol. 57, pp. 1667–1672, 2009.

[3] J. Wertz, Ed., *Spacecraft Attitude Determination and Control*. Kluwer Academic, 1991.

[4] D. Charlambous, “Mathematical tools (physics studies),” Department of Physics Lancaster University, Tech. Rep., page 4.

[5] B. A. Campbell and S. W. McCandless, *Introduction to Space Sciences and Spacecraft Applications*. Gulf Publishing Company, 1996.

[6] G. Maral and M. Bousquet, *Satellite Communications Systems*, 2nd ed. John Wiley & Sons Ltd, 1993.

[7] D. Koks, “Using rotations to build aerospace coordinate systems,” Australian Department of Defence: Electronic Warface and Radar Division System Sciences Laboratory, Tech. Rep., 2006.

[8] “*Direction Cosine Matrix ECEF* to NED.” [Online]. Available: http://www.mathworks.com/access/helpdesk/help/toolbox/aeroblks/directioncosinematrixeceftoned.html,(March2010)

[9] A. Thompson, “*Untitled Article*.” [Online]. Available: http://atacolorado.com/eulersequences.doc(March2010)

[10] S. Cakaj, W. Keim, and K. Malaric, “Communications duration with low earth orbiting satellites,” in *4th IASTED International Conference on Antennas, Radar and Wave Propagation*, Montreal, Canada, ARP 2007, May 30 - June 1 2007, pp. 85 – 88.

Unifom Generators and Combinatorial Designs

Alexis Bonnecaze
Université de la méditerrané,
IML, ERISCS
Marseille, France
Email: bonnecaze@univmed.fr

Pierre Liardet
Université de Provence,
LATP
Marseille, France
Email: liardet@cmi.univ-mrs.fr

***Abstract*—The concept of randomness is fundamental in many domains and in particular in cryptography. Intuitively, a system, which is unpredictable is more difficult to attack and as a consequence, creating sequences that look like random represents a major issue. In this paper, we first study theoretically how a source of symbols with positive entropy can be turned into a true random generator called Bernoulli. We concentrate on a special type of generators, which consists in randomly choosing k elements out of n elements. After studying some existing algorithms, which are of Las Vegas type, we introduce new constructions from a binary generator taken as a primary random source of symbols. Our method is based on combinatorial block designs and we construct algorithms of Monte Carlo type involving random walks. We analyze in detail properties of our general method. Several explicit constructions of k-out-of-n generators are given. We show that the speed of convergence to the uniform distribution is better than any known method using algorithms with bounded running times.**

***Keywords*-Random Generator; Design; k-out-of-n Algorithms; Markov Chain; Random Algorithms;**

I. Introduction

Random or pseudo-random generators of numbers are omnipresent in cryptography. The concept of randomness is used for various purposes. Salt and nonce are well known examples of random values. A nonce (number used once) is used to check the freshness of a message or as an initialization vector. In conjunction with password, salt is frequently used in order to complicate a dictionary attack. Many cryptographic primitives also require random or pseudorandom inputs like keys or values to make algorithms probabilistic. It is well known that digital signatures or challenges in authentication protocols require the use of random quantities. For these reasons, finding of good pseudo-random generator is a stake in first importance. There exist in the literature lots of pseudo-random generators, which imitate in some sense a sequence of independent random variables X_n, uniformly distributed like, for example, the Blum Blum Shub generator (BBS) [5].

However, some applications require more complex generators, called k-out-of-n generators. They consist in picking randomly k elements in a set of n elements. The need for such generators is multifarious. They help to reach load balancing in certain distributed systems like high-availability clusters for example. They are also useful in security protocols, like threshold signatures [27] or time-stamping schemes [6], [7]. Suppose we decide to create a service of authentication (signatures and time-stamps). Most of protocols use the concept of trusted third party even though it may be difficult to build a third party server that can be trusted. Indeed a server may be corrupted or victim of Denial of Service attacks (DoS). Moreover, the problem may not have a malicious origin but a hardware or software one. An important requirement of existing protocols is to prevent the server from failing. In fact, schemes relying on a unique third party server cannot be fully trusted. Therefore, such a scheme should use a multi-server architecture that could be described as follows: the protocol uses n third party servers. For each request to the system, k servers out of the n servers are randomly chosen to process the request. These k servers are said to be the active servers. In this configuration, an attacker does not know a priori what are the active servers for a given request. The attack is then much more difficult to operate. Moreover, the randomness of the generator allows the system to be load balancing.

The above example is at the origin of this work as we noticed that the construction of a uniform k-out-of-n generator from a unbiased (or not) 0-1 valued Bernoulli generator were not so much studied in the literature where uniform random number generators in the unit interval are commonly used. The underlying general problem is in fact to construct unbiased Bernoulli generators from a binary Bernoulli one, supporting a possible bias.

The paper is organized as follows. Section II recalls necessary background, regarding (true) random generators from a theoretical point of view. To this aim, basic properties of symbolic Bernoulli dynamical systems are given. They serve as theoretical models of random sources of symbols with positive entropies. Section III reviews some existing solutions of the k-out-of-n problem. Both first ones are very simple and of classical conception, while the third one called RANKSB algorithm in [17] is due to Nijenhuis and Wilf. This former algorithm leads to important bias in comparison to the uniform distribution. Most of them are Las Vegas algorithms. Section IV is devoted to our constructions that are supported by Monte Carlo algorithms, hence with a bounded running times. Such algorithms approach the uniform distribution exponentially fast. We first propose a generator based on the existence of some special combinatorial objects, namely block t-designs or including some Steiner systems. This construction generates k elements from a set of n elements in a uniform fashion from a binary generator. It consists in randomly picking a block, called word, from the blocks of the design. This word is then modified in order to obtain a vector of weight k and length n with the desired property. We will see that it can be useful to introduce the notion of block codes, since codewords of a fixed Hamming weight in some codes hold a design.

The second type of constructions is based on random walks on a finite set following the action of a finite number of generators of a group acting transitively. The first construction uses the permutation group $\mathfrak{S}_n$. The second construction is very related to the method based on block designs. The main difference is in the way of randomly picking a word in the appropriate set. The algorithm makes use of the automorphism group of the design and executes a random walk on its blocks.

This article is an extended version of [1]. We present in detail our methods, including mathematical foundations. Compare to the conference version, our construction makes use of block designs and not just Steiner systems. Indeed, block designs are widespread and, as a consequence, our construction can be applied for a wide range of parameters k and n. In order to illustrate our methods, we give several explicit constructions. For each chosen couple of parameters k and n, we exhibit a correct design and calculate the speed of convergence of the generator.

II. Randomness and mathematical foundations

A. From a binary random number generator to a q-ary one

In [18], the NIST defines a random bit sequence as follows. “A random bit sequence could be interpreted as the result of the flips of an unbiased *fair* coin with sides that are labeled “0" and “1," with each flip having a probability of exactly $1/2$ of producing a “0" or “1." Furthermore, the flips are independent of each other: the result of any previous coin flip does not affect future coin flips". Similarly, replacing the set of issues $\{0, 1\}$ by a given finite set $\mathcal{A}$, one can define a random $\mathcal{A}$-valued sequence as an $\mathcal{A}$-valued independent and identically distributed random variables $X_n(\cdot)$ with common law the uniform distribution on $\mathcal{A}$. For our purpose, it is convenient to translate these notions in term of Bernoulli dynamical systems. To this aim, we recall basic definitions and general results related to symbolic dynamical systems and entropy.

Assume that $\mathcal{A}$, also called alphabet, is equipped with the discrete topology, has q elements (or letters) with $q \geq 2$, and set $\Omega(\mathcal{A}) = \mathcal{A}^{\mathbb{Z}}$ the product space endowed with the usual compact product topology. Elements ω of $\Omega(A)$ are infinite bilateral sequences

$$\omega = (\ldots, \omega_{-3}, \omega_{-2}, \omega_{-1}; \omega_0, \omega_1, \omega_2, \ldots)$$

with origin pointed at ω_0. The alphabet $\mathcal{A}$ is usually endowed with the uniform distribution denoted by $U(\mathcal{A})$ but we also consider other distributions. The space $\Omega(\mathcal{A})$ is equipped with the σ-algebra of its Borel subsets. Any probability μ on $\mathcal{A}$, induces the infinite product probability μ^∞ on $\Omega(\mathcal{A})$, which is defined from cylinder sets. More precisely, for any $a := a_0 \ldots a_{n-1}$ in A^n, let

$$[a] := \{\omega \in \Omega\,;\, \forall\, i \in \{0, \ldots, n-1\},\ \omega_i = a_i\}$$

be the cylinder set of base a, then

$$\mu^\infty(\sigma^k[a]) = \mu(\{a_0\}) \ldots \mu(\{a_{n-1}\})$$

for any $k \in \mathbb{Z}$. The shift $\sigma : \Omega(\mathcal{A}) \to \Omega(\mathcal{A})$ is defined by $\sigma(\omega)_n = \omega_{n+1}$. Now, the triplet $B(\mathcal{A}, \mu) := (\Omega(\mathcal{A}), \sigma, \mu^\infty)$ is by definition the Bernoulli (random generator) on $\mathcal{A}$ with source distribution μ. In case $\mu = U(\mathcal{A})$ we set $U^\infty(\mathcal{A})$ for μ^∞ and $B(\mathcal{A}, U(\mathcal{A}))$ is simply denoted by $B(\mathcal{A})$. Let $\pi_0 : \Omega(\mathcal{A}) \to \mathcal{A}$ be the central projection ($\pi_0(\omega) = \omega_0$), then the maps $B(\mathcal{A})_n := \pi_0 \circ \sigma^n$ defined on the probability space $(\Omega(\mathcal{A}), U^\infty(\mathcal{A}))$ form an $\mathcal{A}$-valued sequence of independent random variables identically distributed with distribution law $U(\mathcal{A})$. This corresponds to a q-ary random number generator for $q = \#\mathcal{A}$.

B. Symbolic random sources, entropy and factors

To fix some notations and for convenience of the reader, we recall basic definitions and facts from ergodic theory. For more details and proofs we refer to the monographs [32] and [25], and specific references below. Our mathematical model of random source of letters in $\mathcal{A}$ should be identified to a symbolic dynamical system (SDS) with symbols in $\mathcal{A}$, that is to say a triple $(\Omega(\mathcal{A}), \sigma, \nu)$ where ν is a Borel measure on $\Omega(\mathcal{A})$, which is σ-invariant, *i.e.*, $\nu = \nu \circ \sigma^{-1}$. The entropy of such a system is, by the classical Kolmogorov-Sinai Theorem or by definition,

$$H(\sigma, \nu) := -\lim_N \frac{1}{N} \sum_{a \in \mathcal{A}^N} \nu([a]) \log \nu([a])$$

with the convention $0. \log(0) = 0$.

Let $\Omega' := (\Omega(\mathcal{A}'), \sigma', \nu')$ and $\Omega := (\Omega(\mathcal{A}), \sigma, \nu)$ be symbolic dynamical systems. Their direct product is the SDS $\Omega \times \Omega' = (\Omega(\mathcal{A} \times \mathcal{A}'), \sigma \times \sigma', \nu \otimes \nu')$ and so $H(\sigma \times \sigma', \nu \times \nu') = H(\sigma, \nu) + H(\sigma', \nu')$. This construction means that the source corresponding to Ω and Ω' are independent. If there is a measure preserving map $f : \Omega' \to \Omega$ commuting with the shifts, *i.e.*,

$$\begin{cases} \nu = \nu' \circ f^{-1}\,, \\ \sigma \circ f = f \circ \sigma' \ (\nu\text{-almost everywhere})\,, \end{cases} \tag{1}$$

then Ω is said to be a factor of Ω' with factor map f. In that case $H(\sigma, \nu) \leq H(\sigma', \nu')$. In the following, the value

$\frac{H(\sigma,\nu)}{H(\sigma',\nu')}$ will be called entropy rate and denoted by $\tau(\nu,\nu')$ or simply by τ. If f is invertible (up to negligible sets), with measure preserving inverse map, then Ω is said to be isomorphic to Ω' with conjugate map f and if Ω' is a Bernoulli random generator, then by extension Ω is also called a Bernoulli SDS.

We have $H(\sigma^{-1},\nu) = H(\sigma,\nu)$ and more generally $H(\sigma^{\pm k},\mu) = kH(\sigma,\nu)$ for any natural number k. In fact the k-th iterate $(\Omega(\mathcal{A}),\sigma^k,\nu)$, $k \neq 0$, is canonically identified with $(\Omega(\mathcal{A}^k),\sigma^{(k)},\nu^{(k)})$, where $\sigma^{(k)}$ is the shift on $\Omega(\mathcal{A}^k)$ and $\nu^{(k)}$ is induced by ν restricted to cylinder sets of $\Omega(\mathcal{A}^k)$ viewed as particular cylinder sets from $\Omega(\mathcal{A})$. Therefore $H(\sigma^{(k)},\nu^{(k)}) = kH(\sigma,\nu)$.

This construction has the following important consequence. Assume that we have got a binary source with positive entropy h, for example a source extracting from random jitter in an electrical circuit or quantum effects in semiconductors or timing of running current process, or a combination of these sources. Then, we derive a source of binary blocks of length k having entropy kh.

In case of a Bernoulli system $B(A,\mu)$ (hence $\nu = \mu^\infty$ with the above definition), its entropy is easy to compute:

$$H(\sigma,\mu^\infty) = -\sum_{a\in\mathcal{A}} \lambda(\{a\}) \log \lambda(\{a\}).$$

In particular $H(\sigma,U^\infty(\mathcal{A})) = \log(\#\mathcal{A})$.

As a consequence of a deep result of Y. Sinai (see [28]), if the entropy $H(\sigma',\nu')$ of $\Omega' = (\Omega(A'),\sigma',\mu')$ is greater than or equal to $\log\#A$ then there exists a factor f from $(\Omega(A'),\sigma',\mu')$ onto the uniform Bernoulli generator $B(A)$. Properties (1) shows that f is determined by the central coordinate map $f_0 = \pi_0 \circ f$ since we have $f = (\dots,f_{-2},f_{-1};f_0,f_1,f_2,\dots)$ with $f_k = f_0 \circ \sigma^k$ $(k \in \mathbb{Z})$. In particular f_0 is constant equal to a on $C_a := f^{-1}([a])$. Moreover, all partitions $\{\sigma'^m(C_a)\,;\ a \in A\}$ $(m \in \mathbb{Z})$ are independent in between. Hence, building such a partition is usually intractable by computer except in particular cases pointed out below. Another consequence of the above construction and Sinai's theorem is that, from any given random binary source of positive entropy, theoretically there exists a factor built from of a suitable power of this source, which is $B(\{0,1\})$, the factor map consisting in distributing binary sequences in two parts equally likely and independently in the time. This is the most hard problem to be solved in practice for constructing, from a suitable physical random source, a binary random generator according to the NIST definition.

Following results of D. S. Ornstein [19], we recall that the family of Bernoulli dynamical systems is remarkably stable. In particular, they are characterized by their entropy (two such systems of equal entropy are isomorphic), any direct product of Bernoulli systems and any non trivial iterate (and also any root) of a Bernoulli system are Bernoulli. Moreover any factor of a Bernoulli system is also Bernoulli. These properties imply that any probability algorithm that takes in input a Bernoulli source and output a random source of symbols always gives rise to a Bernoulli SDS, isomorphic to some $B(\mathcal{A},\mu)$. In this paper we propose algorithms that take as inputs the outcomes of an appropriated Bernoulli source $B(\{0,1\}^k)$ and output a random source of letters in a given alphabet $\mathcal{A}$ whose distribution of letters is exactly or approximate accurately the uniform distribution. We may distinguish three sorts of such random algorithms.

(A1) Algorithms that output the uniform distribution on $\mathcal{A}$ in a bounded running time.

(A2) Las Vegas algorithms: they output the uniform distribution on letters with unbounded running time but with a finite expectation.

(A3) Monte Carlo algorithms: they end in a bounded running time, output a distribution usually distinct to the uniform distribution but arbitrarily closed to it in term of total variation.

The following theorem depicts the first case.

Theorem 1: If an algorithm of type (A1) takes input from issues of the source $B(\{0,1\}^k)$ and produce a uniform Bernoulli source of entropy $\log_2 q$, then $q = 2^s$ with $s \leq k$.

Proof: By assumption, the algorithm can be identified to a factor maps f with central coordinate map f_0 having its values in $\{0,1,\dots,q-1\}$. That leads to the partition $\{f_0^{-1}(\{j\})\,;\ 0 \leq j < q\}$ of $\Omega(\{0,1\}^k)$ with $U(\mathcal{A}^k)(f_0^{-1}(\{j\}) = 1/q$ and there exists an integer $L \geq 1$ such that each $f_0^{-1}(\{j\})$ is the union of some cylinder sets of the form $C_L(a) := \{\omega \in \Omega(\{0,1\}^k)\,;\ (\forall i)(|i| > L \text{ or } \omega_i = a_{i+L})\}$, $a_0 \cdots a_{2L} \in (\{0,1\}^k)^{2L+1}$. This implies that q divides 2^{kL} and since $\log q \leq k$ one has $q = 2^s$ with $s \leq k$. ■

Obviously, having in hands a uniform binary source, like tossing an unbiased coin, for cryptographic applications is impractical. But such an abstract uniform Bernoulli generator of binary sequences serves as a benchmark for evaluation of random generators and pseudo-random generators. In fact, security of most cryptographic algorithms and protocols using pseudo-random generators is based on the assumption that it is infeasible to distinguish use of a suitable binary pseudo-random number generator (PRNG) from use of a (truly) random number generator (RNG) defined as the SDS $B(\{0,1\})$. As an example, the pseudorandom generator BBS has been proven secure in the sense that an attacker cannot predict, in a reasonable time, the next bit of the outcome with a probability greater than $1/2$ (see [5]).

Putting apart the independency, the first major problem is then to construct generators G_n of elements (called states or symbols) of a finite set $\mathcal{A}$, such that the distribution law P_n of G_n converges to the uniform distribution $U(\mathcal{A})$ on $\mathcal{A}$ as n tends to infinity. In order to quantify this convergence, we use the total variation distance between P_n and $U(\mathcal{A})$.

This distance is defined by

$$d(P_n, U(\mathcal{A})) = \frac{1}{2} \sum_{a \in \mathcal{A}} \left| P_n(a) - \frac{1}{\#\mathcal{A}} \right|$$

where $P_n(a)$ is the probability that the generator G_n outcomes the state a.

A classical method to solve this problem is to introduce a transitive and irreducible Markov chain of transition matrix T, with space of states $\mathcal{A}$, such that the uniform distribution on $\mathcal{A}$ is the stationary distribution of the chain. Constraints of the problem are essentially on the incidence matrix of the chain since each state a can only transit on a number $\tau(a)$ of states such that $0 < \tau(a) \leq \tau_{\max}$ where $\tau_{\max}$ is a small constant compare to $\#\mathcal{A}$. The stationary distribution is then approached by considering Markov random walk on a finite graph. In fact, the general theory of finite Markov chains shows that (by Perron-Frobenius's theorem) there exist two constants $C > 0$ and $\rho \in]0,1[$ such that for every pair of states (i,j), one has

$$\left| (T^n)_{ij} - \frac{1}{\#\mathcal{A}} \right| \leq C\rho^n \,. \qquad (2)$$

When $\#\mathcal{A}$ is big, computation of constants C and ρ satisfying (2) is generally not effective, even if we assume that n is large enough. In fact, if m_0 is an integer such that for a constant $c \in]0,1]$ we have $(T^{m_0})_{ij} \geq c\frac{1}{\#\mathcal{A}}$ for all pairs of states (i,j), then inequality (2) becomes

$$\left| (T^n)_{ij} - \frac{1}{\#\mathcal{A}} \right| \leq (1-c)^{\lfloor n/m_0 \rfloor} \,. \qquad (3)$$

More details concerning finite Markov chains can be found in [26] or [13]. Random walks on groups or finite graphs is treated in [23], [10] and a survey on recent results on the subject can be found in [24].

III. Existing algorithms

Construction of a k-out-of-n generator greatly depends on the requirements of the applications. They could involve the level of security, the amount of resources (CPU, memory, etc.) needed or the generators used as a primary source of randomness. In this former case, our reference, the generator BBS, corresponds to the abstract model $B(\{0,1\})$. Given the set $\mathcal{P}_k^n := \{F \subset E\,;\ \#F = k\}$ endowed with the uniform distribution, the ultimate goal is then to construct from $B(\{0,1\})$ a sequence of independent random variables X_m of distribution P_m such that $d(P_m, U(\mathcal{P}_k^n)) \leq Ce^{-cm}$ for $m \geq m_0$, where C, c and m_0 are explicit constants that can be used in practice. Now we review some standard k-out-of-n generators according to above classification $A1$–$A3$.

(1) An algorithm of type (A1) exists if and only if $\binom{n}{k}$ is a power of 2. This implies that $n = 2^s$ and $k = 1$. In that case, the algorithm is just the identity map: the output is equal to the input given by the generator $B(\{0,1\}^s)$ or by iterating $B(\{0,1\})$ s times.

(2) The most obvious algorithm is based on the construction of a set of k elements by randomly picking an integer between 1 and n, then renew the process to obtain an other element between 1 and n but distinct from the first one and so on. It is typically a Las Vegas algorithm. From a probabilistic point of view, this process needs an average of $n\left(\frac{1}{n} + \frac{1}{n-1} + \cdots + \frac{1}{n-k+1}\right)$ random runs (see [14]), an average, which is $\mathcal{O}(n)$ for $1 \leq k \leq n/2$.

(3) The probably oldest probabilistic algorithm to uniformly and randomly pick k elements among n elements relies to the Fisher-Yates shuffle algorithm for generating a random permutation σ of $\mathcal{E}_n = \{1, \ldots, n\}$ but stoping the construction as soon as the values $\sigma(1), \ldots, \sigma(k)$ are constructed. The Fisher and Yates original method consists to randomly pick an element e_1 from $\mathcal{E}_n$ with the uniform distribution then pick an element from $\mathcal{E}_n \setminus \{e_1\}$ with the uniform probability and so on, k times. The underlying probabilistic model is based on the representation of integers in factorial basis. More precisely, let $I_k := \{0, \ldots, n-1\} \times \{0, \ldots, n-2\} \times \cdots \times \{0, \ldots, n-k\}$ equipped with the uniform probability. Any element $i := (i_1, i_2, \ldots, i_k)$ of I_k corresponds univocally to the integer $n_i \in \{0, \ldots, n!-1\}$ given by its expansion in the factorial basis:

$$n_i = i_k.(n-k)! + i_{k-1}.(n-k+1)! + \cdots + i_2.(n-2)! + i_1.(n-1)!$$

and the corresponding subset $P_i = \{p_{i_1}, \ldots, p_{i_k}\}$ of $\mathcal{E}_n$ given by $p_{i_1} = i_1 + 1$, then p_{i_2} is the (i_2+1)-th coordinate of the n-tuple M_2 deduced from the n-tuple $M_1 = (1, 2, \ldots, n-1, n)$ by exchanging the n-th coordinate with the (i_1+1)-th one. For $2 \leq s \leq k-1$, p_{i_s} is constructed by induction as the (i_s+1)-th coordinate of the n-tuple M_s deduced from M_{s-1} by exchanging the $(n-s+2)$-th coordinate with the $(i_{s-1}+1)$-th coordinate. Notice that there exist $k!$ integers i, which give the same subset. The major drawback of this process is that it requires to have independent uniform generators on s letters, for $n-k < s \leq n$. The given construction leads to a factor map from $B(A', \mu')$ onto $B(A, \mu)$ with $A' = I_k$ and A equal to $\mathcal{P}_k^n$. The entropic rate is thus $\frac{\log\binom{n}{k}}{\log n! - \log(n-k)!}$.

A better version of the Fisher-Yates algorithm was introduced by R. Durstenfeld in [11] and later by D. Knuth in [14] with his Algorithm P (Shuffing) on page 145. A random number generator G uniformly distributed in $[0,1]$ is used in Algorithm P. It can be formally obtained from $B(\{0,1\})$ by computing $G(\omega) = \sum_{n=0}^{\infty} \omega_n 2^{n-1}$ $(\omega \in \Omega(\{0,1\})$. G is applied for computing a random integer $k(\omega) = 1 + \lfloor jG(\omega) \rfloor$, between 1 and j. In this way, computation of $\lfloor jG(\omega) \rfloor$ could never stop (but only for n particular issues of ω). Nevertheless, the algorithm is of type (A2). Recently, R. Rolland in [22] proposed an algorithm of Las-Vegas type, which is analogous to

Fisher-Yates algorithm. This algorithm constructs a k-out-of-n random generator involving only a uniform Bernoulli random generator $B(\{0,1\}^{\ell})$ with $n \leq 2^{\ell}$.

(4) RANKSB algorithm proposed in [17] takes into account some algorithmic constraints (in particular in terms of CPU and execution time), which are not verified by the preceding methods. We give a simplified version of RANKSB. It consists in subdividing the interval $[1,n]$ in k sub-intervals R_j with approximately the same length, and randomly choose the number r_j of elements to be selected in each R_j. If we don't take into account that $r_j \leq \#R_j$, the k-tuple $(r_1,\ldots,r_k)$ of integers $r_j \geq 0$ such that $r_1+\cdots+r_k = k$ follows binomial law constructed from k independent runs of integers in $\{0,\ldots,n\}$, with the uniform law. In order to avoid that r_j be greater than $\#R_j$, we recompute the subdivision in sub-intervals R_j then in each R_j we select r_j elements using method (2), (3) or any others. The algorithm uses a source of entropy $k\log n$ corresponding to $B(\{1,...,n\}^k)$, to obtain the factor corresponding to a shift of Bernoulli on the set of k-tuples $(r_1,\ldots,r_k)$ as above (distributed according to the binomial law), which is of entropy

$$H_k := -\sum_{r_1+\cdots+r_k=k} \frac{1}{k^k}\binom{k}{r_1,\ldots,r_k}\log\Big(\frac{1}{k^k}\binom{k}{r_1,\ldots,r_k}\Big).$$

When k is small, let us use (2) to pick the r_j elements in R_j so that the corresponding entropy rate is (in average) assumed closed to 1. The entropy rate of the algorithm is then $H_k/k\log n$. Since H_k is less than $\log k^k$, this entropy rate is then less than $\log k/\log n$. Notice that if n/k is small, the output distribution has a non negligible bias.

In the sequel we describe k-out-of-n generators, which are of Monte Carlo type.

IV. Proposed algorithms

In this section, we propose two types of constructions. The first one is based on some remarkable configurations of points in binary vectors spaces, namely, block designs [15]. It leads to optimal uniform generators and exists for a wide range of parameter values k and n since designs are very common. The second type of constructions is based on random walks on some groups or finite sets.

A. Block t-design based constructions

A combinatorial block t-design D with parameters t-(v,k,λ) is an incidence structure $(\mathcal{P},\mathcal{B})$ (where elements of $\mathcal{B}$, called blocks, are subsets of $\mathcal{P}$) satisfying the following conditions:

- $\#\mathcal{P} = v$,
- $\forall\, B \in \mathcal{B}, \#B = k$,
- $\forall\, S \subset \mathcal{P}$ such that $\#S = t,\ \#\{B \in \mathcal{B}\,;\ S \subset B\} = \lambda$.

It is known that a necessary condition for the existence of a t-design is that

$$b_s = \lambda\frac{\binom{v-s}{t-s}}{\binom{k-s}{t-s}},\ \text{for all } s \text{ satisfying } 0 \leq s < t,$$

and where b_s corresponds to the number of blocks that contain any s-set (*i.e.,* set of s elements) of points from $\mathcal{P}$. The Web site [9], maintained and regularly updated by Dan Gordon, gives a database of known t-designs.

A Steiner system is a particular case of a block design. It is simply a block design with parameters t-$(v,k,1)$ and is currently denoted by $\mathrm{S}(t,k,v)$. A Steiner system for $t=2$ is called balanced incomplete block design and for $v = s^2+s+1$, $k = s+1$, the system corresponds to the combinatorial notion of finite projective plane. A necessary condition for the existence of a Steiner system is that the number

$$\frac{\binom{v-s}{t-s}}{\binom{k-s}{t-s}}$$

is an integer for all s satisfying $0 \leq s < t$. This condition is not sufficient and there is no Steiner system with, for example, parameters $(2,6,36)$, $(3,7,37)$ or $(5,6,18)$. In fact, there is no known general sufficient condition on the existence of Steiner systems. For $t=2$ and 3, there exist infinitely many families of Steiner systems and for $t \geq 4$, we refer to [8] or [31].

There exist many ways to construct a design. The easiest one is probably to consider an error correcting code. Indeed, there exist codes whose words of a fixed weight hold designs. For example, words of weight 3 in the Hamming codes yield 2-designs and 5-designs can be constructed from Golay codes. Quadratic residue codes constitute another example of family of codes that yield designs. An other way to obtain designs is to consider Hadamard matrices. In this article, examples of such constructions are given. Finally, there exist many other constructions like, for example, recursive methods [20].

For any design D, its group of automorphisms, denoted by $\mathrm{Aut}(D)$, plays a fundamental role to understand the geometrical structure of D. Elements σ of this group belong to the permutation group $\mathfrak{S}(\mathcal{P})$ of $\mathcal{P}$ and verifies the following condition

$$\mathcal{B}^{\sigma} = \mathcal{B}$$

where $\mathcal{B}^{\sigma}$ denotes the set of $\sigma(B)$ with $B \in \mathcal{B}$.

A brief description of our construction of a k-out-of-n generator from a design can be expressed as follows. First, consider a k-(n,b,λ) design $(\mathcal{P},\mathcal{B})$ and after indexing $\mathcal{P}$ by i, $1 \leq i \leq \#\mathcal{P}$ identify each block B in $\mathcal{B}$ with a binary string of length $\#\mathcal{P}$, the place of 1's indicating the elements of B. Now, choose randomly a block m from the blocks of the design. By construction, the Hamming weight of m is b. Then, randomly choose $b-k$ coordinate positions of 1 in m, replace the corresponding 1 by 0. We get the output m'. Since we are only concerned by the set of positions of the 1's, we may identify it to $\{1,\dots,b\}$ so that the elimination of $b-k$ digits 1 is done by selecting at random, independently of the random choice of m, a subset of $b-k$ elements in $\{1,\dots,b\}$, issuing m' independently of the possible λ-blocks m that cover m'. Hence, the algorithm outputs a random word of length n with Hamming weight k. Notice that finding a k-out-of-n generator is equivalent to finding a $n-k$-out-of-n generator. Of course this algorithm needs uniform random generators on sets of symbols, which are not necessarily of cardinality a power of 2, hence it is usually of type (A2).

Example 1: As an example, we explain in detail the construction of a 5-out-of-24 generator, which is based on a $\mathrm{S}(5,8,24)$. In this case, the blocks of the design can be represented as the words of weight 8 in the extended binary Golay code $\mathcal{G}_{24}$. Our construction is based on the following property [16, page 67]: every binary vector of Hamming weight 5 and length 24 is covered by exactly one word of $\mathcal{G}_{24}$ of weight 8. It turns out that a random generator in $\mathcal{P}_5^{24}$ is easily obtained from a random generator of the words of weight 8 of the Golay code (and vice versa).

We recall the main combinatorial properties of $\mathcal{G}_{24}$. In the sequel $W(m)$ denotes the Hamming weight of a binary string m (also called vector as element of the underlying vector space); the weight distribution of $\mathcal{G}_{24}$ is classical and given by the following table:

weights	0	8	12	16	24
number of words	1	759	2576	759	1

Table 1: Weight distribution of the words of the Golay code $\mathcal{G}_{24}$

A remarkable property of this code is that the set of words of a given weight forms the blocks of a design. Hence, the words of weight 8, called octads, form the blocks of a 5-$(24,8,1)$ design and words of weights 12 form the blocks of a 5-$(24,12,48)$ design. It is worth noticing that, in the case of octads, the parameter λ of the design is equal to 1. This means that a vector of length 24 and Hamming weight 5 is covered by exactly one octad of the code. Thus, octads form a Steiner system with parameters $S(5,8,24)$. Note that there exist in $\mathcal{G}_{24}$ other Steiner systems like $S(4,7,23)$, $S(3,6,22)$ or $S(2,5,21)$, leading to similar constructions of generators.

Since $\mathcal{G}_{24}$ decode at most three errors, it happens that when changing three bits from the value 1 to the value 0 in an octad, one can construct $\binom{8}{5}$ vectors of weight five. If we repeat this process for every octad, we obtain a total of $759 \times \binom{8}{5} = \binom{24}{5}$ vectors of weights five, which is the cardinal of $\mathcal{P}_8^{24}$.

This leads to the following construction:

```
(a) choose a base $\{b_1,\dots,b_{12}\}$ of $\mathcal{G}_{24}$;
(b) choose a random generator $G$ on 12 bits (corresponding to a 12 iterations of a binary Bernoulli);
(c) pick a binary vector $g := g_1 \cdots g_{12}$ from the random generator $G$;
(d) compute the word $m(g) := \bigoplus_{j=1}^{12} g_j b_j$;
(e) if $W(m) = 8$ then $m = 0^{a_1}10^{a_2}1\dots0^{a_8}10^{a_9}$ and do
    (e1) randomly pick a vector $x = x_1 \dots x_8$ of weight three and length eight (there exist $\binom{8}{3} = 56$ such vectors. They can be kept in memory);
    (e2) compute
         $m'' := 0^{a_1}(1\oplus x_1)0^{a_2}(1\oplus x_2)\dots0^{a_8}(1\oplus x_8)0^{a_9}$;
(f) if $W(m) = 16$, then $m := m \oplus 1^{24}$ and go to (e);
(g) if $W(m) = 0$ or $W(m) = 24$, then go to (c);
(h) if $W(m) = 12$, then ask $G$ a binary vector $g' := g'_1 \cdots g'_{12}$ and compute $m' := m(g')$
    if $W(m') = 8$ then $m := m'$ and go to (e)
    else if $W(m' \oplus m) = 8$ then $m := m' \oplus m$ and go to (e)
    else go to (c);
(i) output $m''$.
```

This algorithm makes use of the generator $\Gamma = B_8 \times B(\{1,\dots,56\})$ where B_8 is the generator induced by $B(\{0,1\}^{12})$ on the set P_8 of words of weight eight or sixteen. Since the counting probability of P_8 is $\mu(P_8) = \frac{2\times759}{2^{12}} = 0,37\dots$, the entropy of Γ is equal to $h(\Gamma) = (12\log 2 + \log 56)/\mu(P_8) = 33.30\dots$. It gives in output the uniform generator $B(\mathcal{P}_5^{24})$ with entropy $\log\binom{24}{5}$. Hence we obtain the entropy rate

$$\frac{\log\binom{24}{5}}{h(\Gamma)} = 0,319\dots.$$

In this example, we obtained the blocks of the design by considering the words of a fixed weight in the Golay code. This is not the only method as we will show later.

B. Random walks methods

In this section, we show how random walks can lead to a distribution as closed as desired to the uniform k-out-of-n generator. The first construction applies random walk on a finite group and in particular on the symmetric group $\mathfrak{S}_n$. However, this method is not practical for large values of n since the size of the group becomes huge. The second construction makes use of a Markov walk on the set of blocks of a k-design. In both case the convergence to the uniform distribution is exponential. We illustrate our method by some examples. We consider codes, like Hamming, Golay, or quadratic residue codes, in order to obtain the appropriate design. We also give an example related to Hadamard matrices.

1) Random walk on a finite group, generalities: this topic in cryptographic context was investigated by Sloane in a nice survey [29]. Let us introduce objects and notations necessary for our study.

A random walk on a finite group G is currently defined by a probability Q on G and a homogeneous Markov chain

Γ_n, of space of states G, of transition matrix T given by $T_{g,h} = Q(gh^{-1}) = P(\Gamma_{n+1} = g|\Gamma_n = h)$. If the support of Q generates G, the chain is irreducible and its stationary distribution is the uniform distribution $U(G)$ on G. From the identity ($\Gamma_0 = \{e\}$), and a sequence of independent random variables X_n, the walk can be described inductively by $\Gamma_1 = X_1$, $\Gamma_n = X_n\Gamma_{n-1}$, the law of Γ_n being given by $Q^{(1)} = Q$ for Γ_1 and convolution product $Q^{(m)}(g) = Q{*}Q^{(m-1)}(g) = \sum_{h\in G} Q^{(m-1)}(gh^{-1})Q(h)$ for Γ_m (≥ 2). Here, we suppose that the chain is irreducible and aperiodic. Hence there exist an integer m_0 and a constant $c > 0$ such that $T^{m_0}_{g,h} \geq c\frac{1}{\#G}$ and then inequality (3) can be applied. This inequality can be translated in terms of distance (2), and can be improved in the case of symmetric walks ($Q(g) = Q(g^{-1})$) or for particular groups previously analyzed in a probabilistic way. For more details, see [2], [3], [10], [23], [24].

One application in cryptography is the popular stream parity de-skewer. Here G is the additive group $\{0, 1\} := \mathbb{Z}/2\mathbb{Z}$ and $Q(0) = \frac{1}{2} - \beta$, $Q(1) = \frac{1}{2} + \beta$ with $0 \leq 2\beta < 1$. Then $T = \frac{1}{2}J + \beta S$ with $J := \begin{pmatrix}1 & 1\\ 1 & 1\end{pmatrix}$ and $S := \begin{pmatrix}-1 & 1\\ 1 & -1\end{pmatrix}$. Using $JS = SJ = 0$, $J^2 = 2J$ and $S^2 = 2S$ we get the explicit formula

$$T^n = \frac{1}{2}J + (2\beta)^n S$$

and consequently

$$d(Q^n, U(\{0, 1\})) = e^{-n\log(1/2\beta)}.$$

Moreover, for $k \geq 1$ fixed, the sequence of random variables $(\Gamma_{kn})_{n\geq 1}$ defines the Bernoulli SDS $B(\{0, 1\}, Q^{(k)})$ whose entropy is, after simplification,

$$\begin{aligned} H_k &= \log 2 - \frac{1}{2}(1 + (2\beta)^k)\log(1 + (2\beta)^k) \\ &\quad + \frac{1}{2}(1 - (2\beta)^k)\log\left(\frac{1}{1 - (2\beta)^k}\right). \end{aligned}$$

Hence, $H_k - \log 2 \sim (2\beta)^{2k}$ as k tends to infinity.

2) Random transposition on the symmetric group $\mathfrak{S}_n$ and k-out-of-n generators: random walks on the symmetric group $\mathfrak{S}_n$ have been intensely studied (see the preceding references). Consider a uniform Bernoulli generator $B(\mathfrak{S}_n)$ on the group of permutations $\mathfrak{S}_n$. This generator defines a sequence of random variables $\Sigma_n(\cdot) = B(\mathfrak{S}_n)_n$. Then $C_n = \Sigma_n(\{1, \dots, k\})$ is a sequence of random variables uniformly distributed in the set $\mathcal{P}^n_k$.

An interesting method to construct generators distinct from Fisher-Yates shuffle algorithm is to choose a set E of generators of $\mathfrak{S}_n$ and use a Bernoulli $B(E)$. A result of [10] states that the speed of convergence is in $e^{-\gamma}$ when the walk has a sufficiently large number of steps γ: more precisely, for $E = \{Id, (1, 2), (1, 2, \dots, n), (n, n-1, n-2, \dots, 1)\}$ one has

$$d(G^{(36n^3(\log n+\gamma))}, U(\mathfrak{S}_n)) \leq \alpha e^{-\gamma}\,, \tag{4}$$

where $\alpha > 0$ is a universal constant and for all integers $\gamma \geq 0$. Then, random variables Γ_m (see above) represent a generator converging to the Bernoulli generator $B(\mathfrak{S}_n)$; it outputs, at each step, a permutation σ and $\sigma(\{1, \dots, k\})$. Let $\Gamma_m[k]$ denotes this generator. Its distribution $Q^{(m)}_k$ on $\mathcal{P}^n_k$ is given, for each set A of k elements of $\mathcal{E}_n = \{1, \dots, n\}$, by

$$Q^{(m)}_k(A) = \sum_{\substack{\sigma\in\mathfrak{S}_n \\ \sigma(\{1,\dots,k\})=A}} Q^{(m)}(\sigma).$$

We observe that $\#\{\sigma \in \mathfrak{S}_n\,;\ \sigma(\{1, \dots, k\}) = A\} = k!(n-k)!$ so that

$$\begin{aligned} d(\Gamma_m, U(\mathfrak{S}_n)) &= \frac{1}{2}\sum_{A\in\mathcal{P}^n_k}\left|Q^{(m)}_k(A) - \frac{k!(n-k)!}{n!}\right| \\ &\leq d(\Gamma_m[k], U(\mathcal{P}^n_k))\,. \end{aligned}$$

This last bound validates the construction of $\Gamma_m[k]$ with the same convergence property than that of Γ_m. However, this generator is not practical for large value of n, and in particular for $n = 24$.

3) Homogeneous symmetric random walk on a finite set: Let $\mathcal{M}$ be a finite set of elements and let E be a set of bijections of $\mathcal{M}$. We set

$$\mu := \#\mathcal{M} \quad \text{and} \quad \chi := \#E.$$

A general random walk on $\mathcal{M}$ with instructions in E is given by a distribution law $\mathcal{L}$ on $\mathcal{M}$ and a sequence of E-valued random variables $(X_n)_n$ of given distribution $\mathcal{P}_n$. An outcome $x = (x_k)_k$ of the random sequence $(X_k)_{k\geq 1}$ leads to a walk on $\mathcal{M}$ consisting to start from an initial point m_0 in $\mathcal{M}$, selected according to the law $\mathcal{L}$ (step 0) and the location of the walk after n steps is $m_n = x_n \circ x_{n-1} \circ \cdots \circ x_1(m_0)$. In the sequel, we only pay attention to a Markov symmetric homogeneous and uniform walk, that means a walk satisfying the following properties:

(j) The initial point m_0 is fixed.
(jj) Symmetry: for all Y in E the inverse map Y^{-1} belongs to E and the identity map belongs to E.
(jjj) The random variables X_n are independent, with uniform distribution $\mathcal{P}_n := U(E)$.

Therefore, the space of states of the corresponding Markov chain is $\mathcal{M}$ and the stochastic transition matrix T is given by

$$T_{i,j} := \frac{\#\{Y \in E\,;\ Y(i) = j\}}{\chi} \quad ((i, j) \in \mathcal{M}^2)\,.$$

According to property (jj), the matrix T is symmetric and so, has the uniform distribution on $\mathcal{M}$ for stationary probability. For our applications we assume that

(jv) The chain is mixing.

This assumption is equivalent to the fact that a power of S has all entries positive. Hence, we may define the following important parameter of the chain

$$\kappa := \min\{k \geq 0\,;\ \forall\,(i, j) \in \mathcal{M}^2,\ (T^k)_{i,j} > 0\}.$$

In other words κ is the minimum number of necessary steps to go from any state to any state.

Our next goal is to estimate the so called spectral hole of S. To this aim we use the symmetry of the chain by considering appropriated quadratic forms on the vector space $\mathbb{R}^{\mathcal{M}}$ equipped with the euclidean scalar product denoted by $\langle \xi | \xi' \rangle = \sum_{(i,j) \in \mathcal{M}^2} \xi_i \xi'_j$, with norm $|| \cdot ||$. Each generator Y in E, acting on $\mathcal{M}$ is identified to an automorphism of $\mathbb{R}^{\mathcal{M}}$ permuting the canonical basis. It is represented by an orthogonal matrix, still denoted by Y. The action of Y applies at i will be denoted by $Y \cdot i$. Explicitly, Y is given by $Y_{i,j} = 1$ if $j = Y \cdot i$ and $Y_{i,j} = 0$ otherwise. The inverse Y^{-1} of Y corresponds to the transpose matrix Y^*. The stochastic transition matrix T defined above is now given by

$$T := \frac{1}{\chi} \sum_{Y \in E} Y.$$

By symmetry of T and Perron-Frobenius's theorem, T has μ eigenvalues λ_ν $(0 \leq \nu < \mu)$ whose the largest one is equal to 1, with multiplicity 1. Let ρ be the greater eigenvalue of T distinct from 1, we ordered real eigenvalues as follows:

$$-1 < \lambda_{\mu-1} \leq \cdots \leq \lambda_1 = \rho < \lambda_0 = 1 \,.$$

Theorem 2: With the preceding definitions and notations, we have

$$-1 + \frac{2}{\chi} \leq \lambda_{\mu-1} \quad \textit{and} \quad \rho \leq 1 - \frac{4}{\kappa(\kappa+1)\chi} \,.$$

Proof.

1. The first inequality is easy to prove. By (jj), diagonal terms of T are equal to $1/\chi$, hence the matrix $\frac{\chi}{\chi-1}(T - \frac{1}{\chi} I)$ is stochastic with eigenvalues $\frac{\chi \lambda_\nu - 1}{\chi - 1}$ between -1 and 1. In particular $-1 \leq \frac{\chi \lambda_{\mu-1} - 1}{\chi - 1}$, which gives $-1 + \frac{2}{\chi} \leq \lambda_{\mu-1}$, as expected.

2. The second inequality is more complex to prove. It relies on the comparison of two quadratic forms on $\mathcal{R}^{\mathcal{M}}$.

To every symmetric matrix A indexed on $\mathcal{M}$ we associate the quadratic form $Q_A(\xi) = \langle A\xi | \xi \rangle$. Eigenvalues $\alpha_0 \leq \alpha_1 \leq \cdots \leq \alpha_{\mu-1}$ of A are given by Courant-Fisher theorem (also called mini-max theorem) [12]:

$$\alpha_\nu = \min_F \{ m(F) \,;\, \dim(F) = \nu + 1 \}, \quad 0 \leq \nu < \mu,$$

the minimum being calculated over the set of all subspaces F of $\mathbb{R}^{\mathcal{M}}$ of dimension $\nu + 1$ and

$$m(F) := \max\{ \langle A\xi | \xi \rangle \,;\, ||\xi|| = 1, \text{ and } \xi \in F \}.$$

A straightforward consequence of this theorem is

Corollary 1: Let Q_A and $Q_{A'}$ two quadratic forms on $\mathbb{R}^{\mathcal{M}}$, of symmetric matrices A and A' and eigenvalues λ_ν, λ'_ν respectively (indexed in decreasing order). If for a constant $C > 0$ we have $Q_{A'} \leq C Q_A$, then $\lambda'_\nu \leq C \lambda_\nu$ for every index ν, $0 \leq \nu < \mu$.

For our purpose, choose $A = I - T$. Then we have

$$\begin{aligned} Q_{I-T}(\xi) &= \frac{1}{2} \sum_{i,j} (\xi_i - \xi_j)^2 T_{i,j} \\ &= \frac{1}{2\chi} \sum_{\substack{Y \in E \\ i \in \mathcal{M}}} (\xi_i - \xi_{Y \cdot i})^2 \end{aligned}$$

The bound will result from the following main lemma.

Lemma 1: Consider the symmetric matrix $B = I - \frac{1}{\mu} J$ where J has all its coefficients equal to 1. *Then*

$$Q_B \leq \frac{\chi \kappa(\kappa+1)}{4} Q_{I-T} \,. \qquad (5)$$

Proof. For every pair (i,j) of states, let $Y^{i,j} = Y^{i,j}_{k(i,j)} \cdots Y^{i,j}_0$ with $Y^{i,j}_0 = I$ be a composition of elements of E such that $j = Y^{i,j} \cdot i$ with $k(i,j)$ minimal. Set $|Y^{i,j}| := k(i,j)$. We have $|Y^{i,j}| \leq \kappa$ and

$$\xi_i - \xi_j = \sum_{s=0}^{|Y^{i,j}|-1} (\xi_{Y^{i,j}_s \cdots Y^{i,j}_0 \cdot i} - \xi_{Y^{i,j}_{s+1} \cdots Y^{i,j}_0 \cdot i}).$$

Applying Cauchy-Schwarz inequality gives

$$\begin{aligned} (\xi_i - \xi_j)^2 = |Y^{i,j}| & \sum_{s=0}^{|Y^{i,j}|-1} (\xi_{Y^{i,j}_s \cdots Y^{i,j}_0 \cdot i} - \xi_{Y^{i,j}_{s+1} \cdots Y^{i,j}_0 \cdot i})^2 \\ \leq \chi |Y^{i,j}| & \sum_{s=0}^{|Y^{i,j}|-1} (\xi_{Y^{i,j}_s \cdots Y^{i,j}_0 \cdot i} - \xi_{Y^{i,j}_{s+1} \cdots Y^{i,j}_0 \cdot i})^2 T_{\lambda_{ijs}} \end{aligned}$$

where $\lambda_{ijs} := Y^{i,j}_s \cdots Y^{i,j}_0 \cdot i, Y^{i,j}_{s+1} \cdots Y^{i,j}_0 \cdot i$.
Multiply these inequalities by $1/\mu$ and add all of them, first by summing on j and then on i. The summation on the left side simply gives

$$\frac{1}{\mu} \sum_{(i,j) \in \mathcal{M}^2} (\xi_i - \xi_j)^2 = 2 Q_B(\xi).$$

Hence, the summation on the right side is greater than $2Q_B(\xi)$. By collecting all right terms according to the values taking by $|Y^{i,j}|$ for i fixed we get

$$(\chi/2) \frac{1}{\mu} \sum_{k=1}^{\kappa} k (2 Q_{I-T})$$

since by minimality, there is no loop in the path going from i to j and constructed from $Y^{i,j}$. Therefore, summing over i now leads to the desired inequality $Q_B(\xi) \leq \frac{\kappa(\kappa+1)\chi}{4} Q_{I-T}$.

Eigenvalues of $\frac{1}{\mu} J$ being 1 and 0, those of B are then 1 (with multiplicity $\mu - 1$) and 0. Then Corollary 1 gives $1 \leq \frac{\kappa(\kappa+1)\chi}{4}(1 - \rho)$, which is the second inequality of theorem 2.

Using Theorem 2 and the fact that T is a symmetric stochastic matrix of order μ, we get the following inequality

$$|||T^n - \frac{1}{\mu} J|||_2 \leq \Big(1 - \frac{4}{\kappa(\kappa+1)\chi}\Big)^n ,$$

where $\Big|\Big|\Big|\cdot\Big|\Big|\Big|_2$ denotes the quadratic norm of operators. Now let $P_m^{(n)}$ be the distribution of the walk obtained from the state m_0, and set

$$d(n) := \max_{m\in\mathcal{M}} d(P_m^{(n)}, U(\mathcal{M})).$$

Let I_{m_0} be the column vector in $\mathbb{R}^{\mathcal{M}}$ with all entries 0 except the entry corresponding to m_0. From Cauchy-Schwarz inequality and symmetry of T^n,

$$\begin{array}{rcl} d(P_m^{(n)}, U(\mathcal{M})) & = & \dfrac{1}{2}\displaystyle\sum_{j\in\mathcal{M}} \Big|T^n_{i,m_0} - \frac{1}{\mu}\Big| \\ & \leq & \dfrac{1}{2}\sqrt{\mu}\Big(\displaystyle\sum_{j\in\mathcal{M}} \Big(T^n_{i,m_0} - \frac{1}{\mu}\Big)^2\Big)^{1/2} \\ & \leq & \dfrac{1}{2}\sqrt{\mu}\Big|\Big|\Big(T^n - \dfrac{1}{\mu}J\Big)I_{m_0}\Big|\Big|_2 . \end{array}$$

Since

$$\begin{array}{rcl} ||(T^n - \dfrac{1}{\mu}J)I_{m_0}||_2 & \leq & |||(T^n - \dfrac{1}{\mu}J)|||_2||I_{m_0}||_2 \\ & \leq & \Big(1 - \dfrac{4}{\kappa(\kappa+1)\chi}\Big)^n, \end{array}$$

we obtain

$$d(n) \leq \frac{\sqrt{\mu}}{2}\Big(1 - \frac{4}{\kappa(\kappa+1)\chi}\Big)^n$$

that can be transformed into

$$d(\lceil ab + b\gamma\rceil) \leq e^{-\gamma}\,. \qquad (6)$$

with

$$\begin{array}{rcl} a & = & \dfrac{1}{2}\log\mu - \log 2 \\ b & = & -\dfrac{1}{\log\Big(1 - \frac{4}{\kappa(1+\kappa)\chi}\Big)}\,. \end{array}$$

This inequality exhibits the speed of convergence of the walk to the uniform distribution.

In the next subsection, we apply this general theory to the specific case of random walks on the blocks of a design.

4) Random walk using the automorphism group of a design: We now introduce an efficient uniform k-out-of-n generator, using a random walk on a block of a $k-$design. The walk consists in acting on the block a set E of appropriate generators of the automorphism group of the design.

Gk-n(N) Algorithm

```
INPUT : $N$
OUTPUT : a binary vector of Hamming weight $k$ and length $n$
Choose a block $m$ of weight $b$ among the blocks of a $k-(n, b, \lambda)$ design. The automorphism group $A$ of the design must be transitive on the blocks.
If $A$ is $(b-k)$-transitive on the blocks,
  then
(b.1)  replace $m$ by $m'$, replacing the first $b-k$ coordinates equal to 1 in $m$ by zeros
(b.2)  randomly act on $m'$ the generators of $G$, $N$ times
(b.3)  output the obtained word.
  else
(c.1)  randomly act on $m$ the generators of $G$, $N$ times, and obtain $m'$
(c.2)  randomly choose $k$ coordinates equal to 1 in $m'$ using a $k$-out-of-$b$ generator
(c.3)  output the obtained word.
```

We give in the sequel, examples of constructions for various parameters k and n.

Example 2: Generator 5*-out-of-*24 *associated to the Mathieu group* M_{24}

Let G_{24} be the extended binary Golay code. The Mathieu group, M_{24}, is the automorphism group of G_{24} and can be generated by the following four permutations acting on the coordinates of the words of the Golay code:

$$S : i \mapsto i+1, \quad V : i \mapsto 2i, \quad U : i \mapsto -1/i$$

and

$$W : \begin{cases} \infty \mapsto 0,\ 0 \mapsto \infty, & \\ i \mapsto -(i/2)2 & \text{if } i \text{ is a quadratic residu modulo 23,} \\ i \mapsto (2i)2 & \text{otherwise.} \end{cases}$$

G5-24(N) Algorihm

```
INPUT : $N$
OUTPUT : a binary vector of Hamming weight 5 and length 24
  (a) choose an octad of $G_{24}$ : $m$
  (b) replace $m$ by $m'$, replacing the first three coordinates equal to 1 in $m$ by zeros
  (c) randomly act on $m'$ the four generators or their inverse or the identity, $N$ times
  (d) output the obtained word.
```

Note that M_{24} is 5-transitive on octads. This is why step (b) can be done before acting the generators.

We have now to explicitly construct the random generator of octads. Since the size of M_{24} is huge ($\#M_{24}$ = 210.33.5.7.11.23), the speed of convergence of a walk on the Mathieu group would be mediocre.

Thus we introduce a Markov walk on the set $\mathcal{M}$ of octads by the action of the four aforementioned generators of M_{24}: S, V, U et W. Let I be the identity. Now we make the walk symmetrical by taking the following transition set

$$E := \{I, S, S^{-1}, U, V, V^{-1}, W, W^{-1}\},$$

with the uniform probability.

To show that *G5-24(N) Algorihm* realizes a uniform 5-out-of-24 generator asymptotically with exponential speediness, we determine equation (6) with the correct parameters.

We have to calculate the minimal number of times we have to act elements of E on a given octad in order to obtain all the octads. Since the walk is symmetric, this number corresponds to κ. Taking into account that the identity belongs to E, Table 2 shows that $\kappa = 7$.

Number of octads	Numbers of steps
683	6
76	7

Table 2: number of steps to obtain, during the walk, all octads from a specific octad

With the above notations, we have $\mu = 759$, $\chi = 8$, and $\kappa = 7$. We obtain

$$d(292 + 111\gamma) \leq e^{-\gamma}.$$

The following histogram (Fig. 1) gives a statistical view of what is going on in the case of a very short walk. It represents the number $N(f)$ of octads obtained f times during 7590 walks of length 11 (7590 is equal to ten times the number of octads). The distribution is rather good.

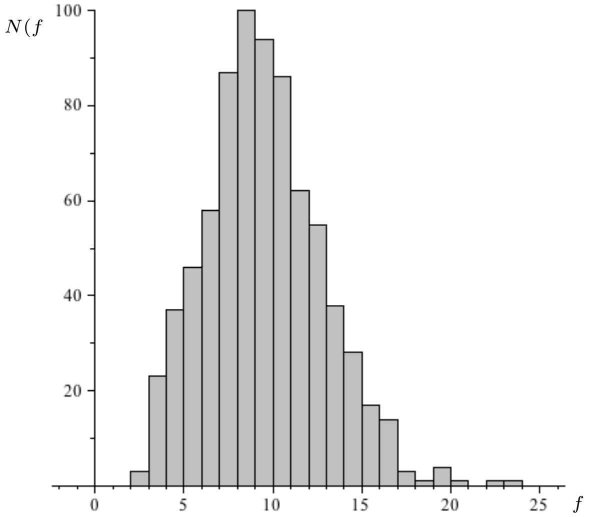


Figure 1. Number $N(f)$ of octads obtained f times during 7590 walks of length 11

Example 3: Random walk on a ternary Golay code

In the previous example, we focused on the binary Golay code. There also exists a ternary Golay code with parameters $[12, 6, 6]$ whose words of Hamming weight equal to 6 yield a 5-$(12, 6, 1)$ design. The number of blocks of the design being equal to 132. Thus, with a similar construction, we can obtain a 5-out-of-12 generator. The automorphism group of the design is of order $95040 = 2^6.3^3.5.11$ and is 5-transitive on the blocks of the design. This group is generated by the following four permutations on the set of 12 coordinates.

$A1 = (5, 9, 12, 7)(6, 10, 11, 8)$ of order 4
$A2 = (3, 12, 7, 9)(4, 6, 10, 8)$, of order 4
$A3 = (1, 3)(4, 8)(7, 11)(9, 12)$ of order 2
$A4 = (2, 4, 5, 8)(6, 9, 10, 12)$ of order 4

and we consider

$$E := \{A1, (A1)^{-1}, A2, (A2)^{-1}, A3, A4, (A4)^{-1}, I\},$$

where I represents the identity permutation. We have $\chi = 8$ and $\kappa = 6$ from Table 3.

Number of blocks	Numbers of steps
74	5
58	6

Table 3: number of steps to obtain, during the walk, all blocks from a specific block

With the appropriate parameters, equation (6) becomes

$$d(146 + 83\gamma) \leq e^{-\gamma}.$$

Example 4: Generator 2-out-of-31

In this example, we consider a 2-$(31, 7, 7)$ design whose automorphism group A is not 2-transitive on its blocks. The number of blocks is 155. The group A is a permutation group acting on the set of 31 coordinates. It is of order $465 = 3.5.31$ and is generated by the following two permutations

$A1 = (1, 16, 15, 13, 9)(2, 18, 19, 21, 25)(3, 20, 23, 29, 10)$
$(4, 22, 27, 6, 26)(5, 24, 31, 14, 11)(7, 28, 8, 30, 12)$
and
$A2 = (2, 29, 10, 5, 20, 6, 17, 15, 21, 3, 26, 19, 9, 8, 11)$
$(4, 23, 28, 13, 27, 16, 18, 12, 30, 7, 14, 24, 25, 22, 31)$.

Consider the set of generators

$$E := \{A1, (A1)^{-1}, A2, (A2)^{-1}, I\}$$

and act the elements of E on a block m, N times, in order to obtain a random block m'. This block can be represented by a vector of weight 7 and length 31.
Computation gives $\kappa = 6$ and with the appropriate parameters, equation (6) becomes

$$d(95 + 52\gamma) \leq e^{-\gamma}.$$

Then we have to randomly choose 2 coordinates equal to 1 in m' using a 2-out-of-7 generator. To do this we consider a 2-$(7, 3, 1)$ design. The automorphism group of the design is $PSL(2, 7)$ and is 2-transitive on its blocks. We just have to replace the first 1 in m' by a zero and carry out a walk on this vector. We then obtain a random vector of weight 2 and length 31. Notice that generators 2-out-of-31 and 2-out-of-7 can be executed in parallel.

Example 5: 3-out-of-16 generator.

It is important to choose a correct design in order to carry out the walk. In fact, to obtain a uniform generator, our method requires to consider designs with automorphism group that is transitive on the blocks.

We introduce here an example of automorphism group of a design, which is not transitive on the blocks. The design is constructed from a Hadamard matrix. Recall that the order of a Hadamard matrix is necessarily a multiple of 4 and Sylvester construction shows that there exist Hadamard matrices of order 2^i for all positive integer i. Let H be the following Hadamard matrix of order 16

$$\begin{bmatrix}
1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1\\
1&1&1&1&1&1&1&1&-&-&-&-&-&-&-&-\\
1&1&1&1&-&-&-&-&1&1&1&1&-&-&-&-\\
1&1&-&-&1&1&-&-&1&1&-&-&1&1&-&-\\
1&1&-&-&-&1&1&-&1&-&-&1&-&-&1&1\\
1&1&-&-&1&-&-&1&-&1&1&-&-&-&1&1\\
1&1&-&-&-&-&1&1&-&-&1&1&1&1&-&-\\
1&-&-&1&-&-&1&1&1&1&-&-&1&-&-&1\\
1&-&-&1&1&1&-&-&-&-&1&1&1&-&-&1\\
1&-&1&-&1&-&1&-&1&-&1&-&1&-&1&-\\
1&-&1&-&1&-&-&1&1&-&-&1&-&1&-&1\\
1&-&1&-&-&1&1&-&-&1&1&-&-&1&-&1\\
1&-&1&-&-&1&-&1&-&1&-&1&1&-&1&-\\
1&-&-&1&1&-&1&-&-&1&-&1&-&1&1&-\\
1&-&-&1&-&1&-&1&1&-&1&-&-&1&1&-\\
1&1&1&1&-&-&-&-&-&-&-&-&1&1&1&1
\end{bmatrix}$$

where the symbol "$-$" stands for "-1". This matrix yields a 3-$(16, 8, 3)$ design with 30 blocks. The automorphism group is of order $2688 = 2^7.3.7$ and is generated by the following permutations on the set of 16 coordinates:
$A1 = (1, 5)(2, 10)(3, 12, 15, 8, 14, 6)(4, 7, 16, 11, 13, 9)$
$A2 = (2, 3, 4)(6, 8, 7)(9, 12, 10)(13, 16, 15)$
$A3 = (5, 10)(6, 9)(7, 12)(8, 11)$.
When acting the generators on the blocks, we obtain 2 orbits. One of order 28 and the other one of order 2. It means that from a given block, it is not possible to obtain all other blocks during the walk.

Let us now consider the design D whose blocks are the words of weight 12 of the extended Hamming code of parameters $[16, 12, 3]$. This is a 3-$(16, 12, 55)$ design with 140 blocks. The automorphism group of D is a permutation group A acting on a set of cardinality 16. It is of order $322\,560 = 2^{10}.3^2.5.7$ and is generated by the following automorphisms:
$A1 = (5, 13)(6, 10)(7, 16)(9, 15)$
$A2 = (5, 16)(6, 9)(7, 13)(10, 15)$
$A3 = (3, 6)(4, 7, 10, 12)(8, 13, 15, 14)(9, 11)$
$A4 = (1, 5)(3, 6)(7, 12)(8, 15)$
$A5 = (2, 10, 13, 12, 4)(3, 16, 11, 8, 15)(5, 7, 9, 6, 14)$.
Consider the set of 8 generators

$$E := \{A1,\ A2,\ A3,\ (A3)^{-1},\ A4,\ A5,\ (A5)^{-1},\ I\}.$$

Then, choose a block m of the design and, since Aut(D) is 3-transitive, just perform a walk on its first 3 coordinates using the elements of E. Here $\kappa = 6$ and so we get a random word of weight 3 and length 16 as desired.

Number of blocks	Numbers of steps
4	4
119	5
19	6

Table 4: number of steps to obtain, during the walk, all blocks from a specific block

The speed of convergence is given by

$$d(148 + 83\gamma) \leq e^{-\gamma}.$$

V. Conclusion

In this paper, we introduced a type of generators, which has not been deeply studied in the literature. Yet, k-out-of-n generators have a wide practical interest, particularly for developing secure applications. Our constructions make use of t-designs in order to obtain uniformity and run random walks in order to control the accuracy of convergence.

We proposed methods to efficiently construct such generators and studied in detail special cases. The speed of convergence of our generators is better than any known k-out-of-n generators.

Acknowledgment

This work has partially been supported by ANR NUGET.

References

[1] Bonnecaze A. and Liardet P., *Efficient Uniform k-out-of-n Generators*, ICSNC 2010, Fifth International Conference on Systems and Networks Communications (ICNS 2010), pp. 177-182.

[2] Aldous D., *Random walks on finite groups and rapidly mixing Markov chains*, Séminaire de Probabilités XVII (1981/82), Lecture Notes in Mathematics, 1059, Springer, Berlin (1983), pp. 243-297.

[3] Aldous D., *Shuffling cards and stopping times*, the American Mathematical Monthly, 93, (1986), pp. 333-348.

[4] Atlas of Finite Group Representations. `http://brauer.maths.qmul.ac.uk/Atlas/v3/spor/M24/`, January 2011.

[5] Blum L., Blum M. and Shub M., *A Simple Unpredictable Pseudo-Random Number Generator*, SIAM Journal on Computing, volume 15, May 1986, pp. 364-383.

[6] Bonnecaze A., Liardet P., Gabillon A. and Blibech K., *Threshold Signature For Distributed Time Stamping Scheme*, Annals of Télécommunications, Volume 62, N. 11-12 (2007), pp. 1353-1364.

[7] Bonnecaze A. and Trebuchet P., *Secure Time-Stamp Schemes: A Distributed Point Of View*, Annals of Télécommunications, Volume 61, N. 5-6, (2006), pp. 662-681.

[8] Colbourn C. and Mathon R., *Steiner Systems*, in Handbook of Combinatorial Designs, second edition, Discrete Mathematics and Its Applications (2007), pp. 102-110.

[9] `http://www.ccrwest.org/cover.html`, January 2011

[10] Diaconis P. and Saloff-Coste L., *Comparison techniques for random walks on finite groups*, The Annals of Probability, Volume 21, No 4, (1993), pp. 2131-2156.

[11] Durstenfeld R., *Algorithm 235: Random permutation*, Communications of the Association for Computing Machinery, volume 7, issue 7, (1964), pp. 420.

[12] Horn R.A. and Johnson R.C., *Matrix analysis*, Cambridge Press, Cambridge, 2nd edition, (2008).

[13] Kemeny J. and Snell L., *Finite Markov chains*, Van Nostrand company, Princeton, (1960).

[14] Knuth D.E., *The Art of Computer Programming, Volume 2: Seminumerical Algorithms*, Addison-Wesley, 2nd edition, (1998).

[15] van Lint J.H. and Wilson R.M, *A Course in Combinatorics*, Cambridge University Press (1992).

[16] MacWilliams F.J. and Sloane N.J.A., *The Theory of Error-Correcting Codes* North-Holland, Eight impression, (1993).

[17] Nijenhuis A. and Wilf H.S., *Combinatorial Algorithms for Computers and Calculators*, Academic Press, Inc., 2nd edition, (1978).

[18] NIST: A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications: `http://csrc.nist.gov/publications/nistpubs/800-22-rev1/SP800-22rev1.pdf`, January 2011.

[19] Ornstein D.S., *Ergodic theory, randomness and dynamical systems*, Yale Mathematical Monographs No. 5, Yale Univ, (1974).

[20] Ray-Chaudhuri D.K. and Tianbao Z., *A recursive method for construction of designs*, Discrete Mathematics, Volumes 106-107, (1992), pp. 399-406.

[21] Rolland R., *Sécurité des générateurs pseudo-aléatoires*, `http ://www.acrypta.fr.`, January 2011.

[22] Rolland R., *Personal communication*, April 20, 2010.

[23] Saloff-Coste L., Lectures on finite Markov chains, in Lectures on Probability Theory and Statistics, Ecole d'été de Probabilités de Saint-Flour XXVI-1996, E. Giné, G.R. Grimmett and L. Saloff-Coste (Authors), Lecture Notes in Math, No. 1665, pp. 301-413.

[24] Saloff-Coste L., *Random Walks on Finite Groups*, Probability on discrete structures, 263–346, Encyclopaedia of Mathematical Sciences, 110, Springer, Berlin. Harry Kesten Editor, (2004), pp. 263–346.

[25] Shields P., *The Ergodic Theory of Discret Sample Paths*, Graduate Studies in Mathematics: 13, American Mathematical Society, (1996).

[26] Seneta E., *Non-negative Matrices and Markov Chains*, Springer Series in Statistics, Springer, Revised Printing (2006).

[27] Shoup V., *Practical Threshold Signatures*, EUROCRYPT'00: Proceedings of the 19th international conference on Theory and application of cryptographic techniques, Springer-Verlag (1999), pp. 207-220.

[28] Sinai Ya. G., *On a weak isomorphism of transformations with invariant measure*, Matematidheskii. Sbornik. (N.S.), 63 (105), No. 1 (1964), pp. 23-42.

[29] Sloane N.J.A., *Encrypting by Random Rotations*, Thomas Beth (Ed.): Cryptography, Proceedings of the Workshop on Cryptography, Burg Feuerstein, Germany, March 29 - April 2, 1982. Lecture Notes in Computer Science 149 Springer (1983), pp. 71-128.

[30] Steiner J., (1853), *Combinatorische Aufgabe*, Journal fur die Reine und Angewandte Mathematik 45, pp. 181-182 .

[31] Steiner Systems, `http://www.ccrwest.org/cover/steiner.html`, January 2011.

[32] Walters P., *An Introduction to Ergodic Theory*, Graduate texts in mathematics: 79, Springer-Verlag (1982).

[33] Yao A., *Theory and applications of trapdoor functions.* In Proceedings of the 23rd IEEE Symposium on Foundations of Computer Science (1982), pp. 80-91.

CUPID: A Communication Pattern Informed Duty Cycling for Large-scale, Low-delay Sensor Applications

Daniela Krüger, Stefan Fischer, and Dennis Pfisterer
Institute of Telematics, University of Lübeck, Germany
{*krueger, fischer, pfisterer*}*@itm.uni-luebeck.de*

***Abstract*—In an Internet of Things (IoT), a plethora of tiny devices will extend the Internet to the physical world and allow for a completely new class of applications. Two possible IoT scenarios are public safety (e.g., surveillance of areas and borders) and smart cities that offer smart services to improve the lives of a city's inhabitants. Both share many underlying challenges in terms of realization. They comprise large-scale deployments of sensor nodes and require long-term, battery-driven operation, low-delay reporting of events, as well as secure and attack-resilient design. However, experiences from real-world trials have shown that decent trade-offs between these two conflicting goals are hard to find. In this paper, we show how staggered wake-ups achieve this. We call this low-delay and low-power duty cycle management scheme *CUPID* because its parameterization is based on the expected communication patterns in the network, duty-cycle and latency requirements. We show by simulations and real-world experiments with more than 150 nodes that our scheme significantly reduces the packet delay for low duty cycle settings, especially in large networks.**

***Keywords*-Wireless Sensor Network, Duty Cycling, Low-delay event reporting, Attack-resilience, Real-world measurements**

I. Introduction

It is envisioned that in the future, all kinds of devices ranging from resource-constraint wireless sensor nodes to powerful server-class computers will interact to form an Internet of Things (IoT). In such a setting, tiny devices will extend the Internet to the physical world and allow for a completely new class of applications. Since the introduction of the IoT vision over a decade ago, it has evolved tremendously since the underlying technologies are maturing. Envisioned application scenarios include environmental monitoring, personal health monitoring, monitoring and control of industrial processes, public safety, smart spaces and increasingly smart cities.

In the public safety domain, the surveillance of areas (e.g., for trespasser detection) is a typical application domain of wireless sensor networks. Examples are [2], [3] and the FleGSens project [4], which realize trespasser detection and localization systems for borders and critical areas. Smart cities have recently gained momentum and examples of smart cities are projects such as CitySense [5], Oulu Smart City [6], T-City Friedrichshafen [7] and the recently started EU FP7-funded SmartSantander [8] project. I such city-scale deployments dozens of thousands devices are being or going to be deployed and will offer smart services to improve the lives of a city's inhabitants.

Despite their differences, both application scenarios share many underlying challenges in terms of realization. These networks are comprised of a large number of wireless sensor nodes that are deployed in a large-scale environment. For the majority of nodes, mains power supply is not an option due to cost or unavailability and hence the nodes must operate on batteries. Another shared property is that in case of an observed event (e.g., a trespasser or a bus approaching a station), the event must be reported as fast as required to a base station that provides access to a backbone network such as the Internet. Compared to the number of nodes, the number of base stations will be small and nodes must forward data in a multi-hop fashion towards the nearest base station. In addition, both networks operate in a potentially hostile environment where they are subject to attacks.

In such networks, important factors are network lifetime, the time to report an event, and resilience. As the replacement of batteries is disproportionately expensive because of inaccessibility and the large number of nodes, saving energy is imperative – typically by using an very low duty cycle ($< 1\%$) – to achieve a long network life time. Simply decreasing the duty cycle results in high multi-hop latencies as nodes must wait for their neighbors to wake up and be ready-to-receive before they can forward a message but especially for critical messages (e.g., alarms) low latencies are crucial. To save energy, approaches such as Low Power Listening (LPL) are frequently used where nodes periodically sample the radio interface for preambles that indicate that a packet is to be received and switch to active mode for reception. Low Power Listening is vulnerable to so-called *sleep deprivation* where attackers drain the nodes' batteries by wasting energy, e.g., by sending messages or preambles that cause nodes to stay awake.

This goal conflict requires a careful trade-off and is nowadays often custom-tailored for each individual application. In this paper, we present a novel delay and communication pattern optimized duty cycle management scheme for sensor networks called CUPID (CommUnication Pattern Informed Duty Cycling in Sensor Networks). Its focus is to minimize the latency in multi-hop networks, which use extremely low duty cycles. We picked up the idea of *staggered wakeups* [9], [10], [11], [12], which we have extended and generalized. In this approach, nodes wake up subsequently in the direction where packets are forwarded. In contrast to existing work, CUPID copes with unreliable communication links and hardware constraints, no node redundancy, and with different communication patters (instead of only allowing network-to-sink communication). In addition to the aforementioned properties, CUPID is resilient against sleep deprivation attacks and can therefore be used in security related applications. The work presented in this paper is an extended version of [1].

The remainder of this paper is organized as follows. In the next section, we review related approaches. Following, Section III introduces CUPID in detail. Then, Section IV provides a simulative and Section V an experimental evaluation. Section VI concludes the paper with a summary and future work.

II. Related Work

In general, there are different types of power management techniques. One of the earliest proposals is *Low Power Listening* (e.g., used in Wisemac [13]), in which nodes are unsynchronized and periodically activate their radio for a fixed duration. A sender transmits short preambles before sending the actual message causing all addressed receivers activating their radio at the respective point in time. However, this implies vulnerability to sleep deprivation attacks.

In *MAC layer schemes* like S-MAC and T-MAC media access during synchronized active slots is coordinated using Ready-To-Send/Clear-To-Send mechanisms, which implies that only unicast communication is possible as well as vulnerability to sleep deprivation attacks.

K-coverage approaches assume that sensing is only possible when a node is active. We do not consider these approaches here, since we assume that sensors have the capability of waking up the node. Also, we don't compare to *on-demand wakeup* radios [14] where ultra-low-power radio interfaces wake up the main radio interface to receive a message. The special hardware required for the first radio significantly increases cost.

Scheduled wakeup schemes do not integrate the duty cycling into the MAC layer but they organize their sleep cycles using predetermined or random schemes. These can be further divided into mechanisms with and without time synchronization. Asynchronous wakeup methods (e.g., [15]) do not require synchronized clocks, but they increase the end-to-end delay especially over long distances. It is therefore common practice to assume (loosely) synchronized clocks. Examples for this approach are *staggered wakeups* where nodes wake up subsequently in the packet forward direction. Stankovic et al. [10] call it *streamlined wakeup*, but assume reliable communication links and high node redundancy. Moreover, they explain their end-to-end optimization only in a short overview and focus on minimizing the detection delay of one sensor event instead.

In [9], Li et al. call the approach *fast path algorithm*. Applying additional wake phases they establish a single path between source and sink, but this is independent from a global duty cycle and from other paths. As a node can handle only a few paths their approach is only appropriate if few nodes send data to few other nodes. In contrast to them we concentrate on a solution for the whole network to allow for network-to-sink communication and vice versa.

Bisnik et al. [11] convert the network into a graph, vertices representing nodes and edges representing communication links. The goal is now to assign (awake) slots to all nodes so that the maximum end-to-end communication delay is minimal. Using the graph-theoretical abstraction the authors prove that this optimization problem is NP-hard given all data is available at a central station. They derive an optimal solution for a rectangular topology where nodes are adjusted in a grid and have exactly four neighbors. However, such a topology can hardly be produced in reality and also assumes reliable communication links. Moreover, the smallest considered duty cycle is 5% if no messages are sent. Nodes sending a message must have knowledge about the wake phases of all destination nodes, which are not necessarily at the same time, and must be additionally awake during these times. By contrast, we investigated constant duty cycles of up to $0{,}1\,\%$ while enabling broadcast communication at the same time.

The authors of [16] describe a wave based communication mechanism to deliver sensor information to and from a designated node without the need of centralized controls. However, they do not consider the resulting duty cycle whereas we meet the requirement of a constant duty cycle. In addition, they base there results exclusively on simulations.

Keshavarzian et al. propose two methods they call *crossed-ladders* and *multi-parent scheme* [12], where nodes are divided into groups depending on their hop based distance to a reference node. Each group is awake during one slot. But their theoretical considerations also assume reliable communication links and disregard the hidden-terminal-problem. Additionally, the authors consider only small networks of at most 4 hops. Their second approach does not support broadcast communication as the network is divided into several partitions, which work independently from each other. Simulation results refer exclusively to the generation of the network partitions. Experimental results are not presented.

To summarize, MAC layer schemes are inefficient, vulnerable to sleep-deprivation attacks or do not support broadcast communication, k-coverage protocols make different assumptions, and on-demand wakeup radios require expensive additional hardware. The techniques closest to our approach are scheduled wakeup schemes. Here, we point out that other approaches do not scale or do not support periodic or broadcast communication. Many authors base their work on unrealistic assumptions and it remains open in which way varying communication ranges influence their approaches. In contrast to these, we investigate the influence of medium access and the hidden terminal problem, which is unavoidable when nodes residing in the same level cannot hear each other. Moreover, CUPID guarantees that all neighbors of a sending node are ready-to-receive so that arbitrary communication patterns are supported and nodes are not obliged to keep track of forwarding directions or sleep cycles of parent nodes.

III. CUPID - Communication Pattern informed Duty Cycling

In this section, we present CUPID, our communication pattern informed duty cycle management scheme. The major goals of CUPID are

1) conservation of energy,
2) resilience against sleep deprivation attacks,
3) minimization of end-to-end message delay,
4) support for broadcast communication, and
5) reduction of collision probability.

Depending on the requirements of the application, CUPID can be parameterized in order to meet them.

The basic idea of our scheme is that groups of adjacent nodes wake up and enter sleep mode subsequently. The idea is not completely new, but for practical use, several problems must be taken into account and the user must be aware of advantages and disadvantages. All nodes in one group follow the same sleep/wake-cycle. One after the other group wake up (depending on their hop-based distance to a reference node, e.g., a sink), stay awake for the same period of time and go asleep one after the other again. On a global level, these wake phases form a *wave* from the sink to the farthest nodes and vice versa.

We consider a line-shaped network topology as shown in Figure 1(a) where nodes are arranged in a 1-dimensional way and the generalized case shown in Figure 1(b). The lines limit the hop-based range to the reference node (black dot). The particular property (discussed in Section IV-B) of the rectangle topology with regard to CUPID is that not all nodes of the same group are within each other's communication range.

Figure 2 shows three groups from a line-shaped network: Group $n-1$, n and $n+1$ where n means n hops away from the reference node. The boxes indicate different states, e.g., Group $n+1$ wakes up at T_0 and T_2 and goes to sleep at T_1 and T_3. During a wake period, a node is in receiving mode (indicated by Rx) and may only transmit during a certain interval (indicated by Tx). Like this, the nodes from Group n can transmit to the nodes of both neighboring groups at the same time.

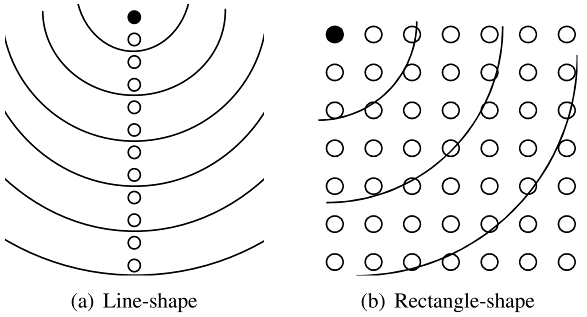

(a) Line-shape (b) Rectangle-shape

Figure 1. Considered network topologies and the hop-based distances to the reference node (black dot)

The major assumption of our scheme is that some kind of time synchronization makes sure that the difference between two clocks stays below an upper bound (cf. Section III-A1). In addition, we assume the presence of at least one reference node (typically a sink), from which all other nodes calculate their hop-based distance. For estimating the network diameter another reference node is needed, which is furthest to the sink or the user must estimate the diameter. The scheme is based on fixed node positions. A looser assumption is that data flows from or to a reference node. CUPID is optimized for this kind of data flow. However, multi-hop communication orthogonal to the wave takes disproportionately more time.

A. Design

This section introduces our considerations when CUPID is applied on real network. We assume that application designers have a fixed application scenario where some parameters are known a priori (e.g., the network's diameter) and some are desired (e.g., maximum end-to-end delay or duty cycle). Table I lists the parameters relevant to CUPID.

Abbreviation	Unit / Range	Name
ND	$[Hops]$	Network Diameter
EED	$[sec]$	max. End-to-End Delay
DC	$\in [0,1]$	Duty Cycle
$d_{slotlen}$	$[sec]$	Slot Length
d_{sil}	$[sec]$	Silence Length
d_{SFL}	$[sec]$	Super Frame Length
d_{tol}	$[ms]$	Tolerance Length

Table I
PARAMETERS RELEVANT TO CUPID

The *Network Diameter* $ND[hops]$ is the maximum number of hops from the reference node. The *End-to-End Delay* $EED[sec]$ describes the maximum amount of time messages need to travel from source to sink. The *Duty Cycle* DC is the percentage of time, where nodes are awake, i.e., the radio interface is active. The remaining parameters are shown in Figure 3. Each group shares a so-called *Sending Slot* of

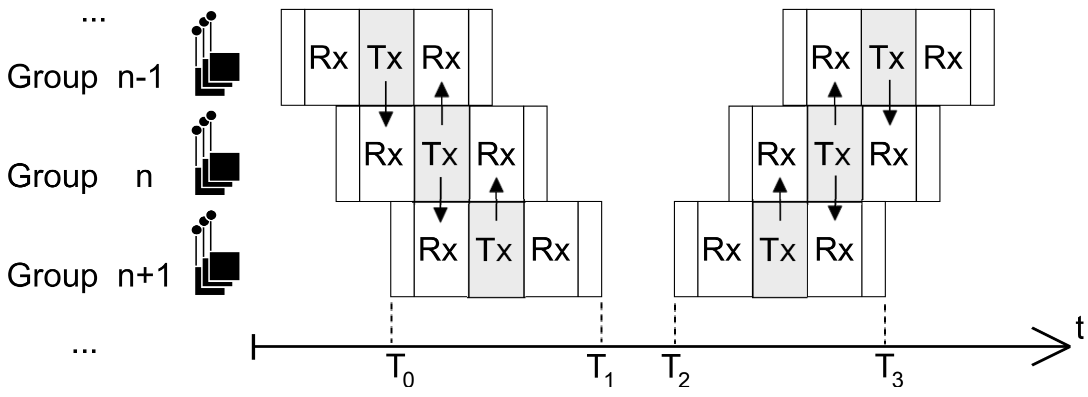


Figure 2. Overview of CUPID's sleep/wake-cycle scheme

length $d_{slotlen}[sec]$, i.e., during this time they access the medium via CSMA/CA. The overall number of sending slots equals the Network Diameter. After the last sending slot, there is a *Silence Period* of length $d_{sil}[sec]$ where no node is awake. In addition, we define the *Super Frame Length* $d_{SFL} = ND \cdot d_{slotlen} + d_{sil}$.

The designer can choose the proportion of waves towards and away from the reference node to support the expected data flow. For instance, a data collection application forwards much data towards a sink, so that the wave towards the sink should be employed much more often than the other one.

1) Coping with clock drift: Coordinated sleep/wake-cycles require synchronized clocks, which can – to a certain extent – be achieved using an appropriate protocol. However, between two clock synchronizations, the hardware clocks of the nodes drift unpredictably, but within known upper bounds. In order to prevent message loss because of sleeping neighbors, there are two different approaches: The first one is to let the nodes wake up d_{tol} ms earlier and go asleep d_{tol} ms later so that the receivers compensate for possible clock drift. This *Tolerance Length* d_{tol} is shown in Figure 3. The second option is to shorten the duration of each sending slot so that the senders compensate for clock drift.

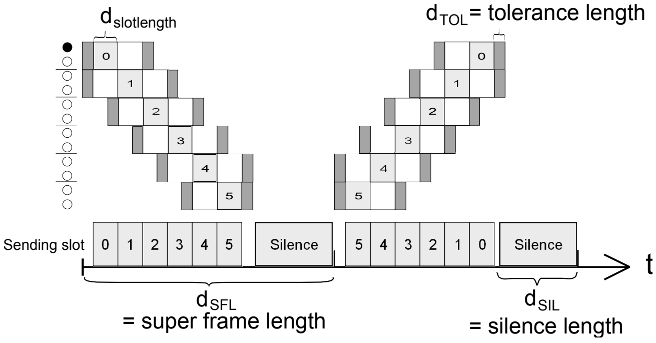


Figure 3. CUPID in detail

We decided to use the first option because the overhead of option two is $3 \cdot 2 \cdot d_{tol}$ ms in the worst case, where the tolerance period remains unused. This is three times larger than the other overhead of $2 \cdot d_{tol}$ ms. However, we must keep in mind that this decision results in slots directly following each other and unsynchronized clocks result in overlapping slots, which cause additional contention for medium access. The scheme is adaptable to unsynchronized clocks by adjusting the duration of the tolerance phase.

2) Mutual Parameter Dependencies: Not all combinations of Network Diameter, Duty Cycle, Slot Length and Super Frame Length (i.e., end-to-end delay) are possible. We show, which parameters have mutual dependencies and which value combinations are feasible.

As each node stays awake three sending slots – its own sending slot as well as the sending slots of preceding and succeeding group - the duty cycle DC results to

$$DC = \frac{3 \cdot d_{slotlen} + 2 \cdot d_{tol}}{d_{SFL}} \cdot 100. \qquad (1)$$

Imagine a required EED of 300 ms and a Network Diameter of 30 hops (and therefore 30 sending slots). Then the maximum value for Slot Length is 10 ms with no Silence Period ($d_{sil} = 0$). Hence, $\frac{d_{SFL}}{ND}$ is the upper bound for Slot Length. Reducing the Slot Length for a given Network Diameter decreases the End-to-End Delay. The lower bound for Slot Length is the time required to transmit at least one single message. However, the Slot Length should allow for sending all desired messages. The more messages nodes have to send or forward, the longer the slots must be.

In addition, depending on the communication pattern it may be beneficial not to use equally long Slot Lengths but to adapt their value depending on the location of a node. For instance, if every node wants to send data to the sink without data aggregation, the message density increases exponentially with the proximity to the reference node. Hence, it can make sense to prolong sending slots (when the wave goes back to the sink) the closer the nodes are to the sink node, so that they have more time for message forwarding.

The Network Diameter also limits the maximum possible value for the Duty Cycle for a given EED. To obtain this, it is again $d_{sil} = 0$. In the case of perfectly synchronized clocks ($d_{tol} = 0$) using Equation 1 we get:

$$DC = \frac{3 \cdot d_{slotlen} + 2 \cdot d_{tol}}{ND \cdot d_{slotlen} + d_{sil}} \cdot 100 \quad (2)$$

$$= \frac{3}{ND} \cdot 100. \quad (3)$$

For instance, with $ND = 5$, the maximum value of Duty Cycle is 60%, while for $ND = 100$ this upper bound goes down to 3%. For some applications it might be necessary to further reduce the Duty Cycle. This can be achieved by decreasing the Slot Length, which automatically increases the Silence Period since the Super Frame Length is fixed by the selected End-to-End Delay.

B. Arbitrary communication patterns

Until now, an implicit assumption has been that data flows from or to a reference node and therefore along the directions of the wave. Our scheme is optimized for this kind of data flow. In the directions of the wave, the delay for a message scheduled at random point of time is nearly equal for all nodes in the network and only depends on the path length to the reference node.

However, if two arbitrary nodes want to exchange messages, CUPID causes additional delay compared to other schemes (such as synchronous cycling). Consider two neighboring nodes N_1 of group 1 and N_2 of group 2. N_2 sends a message MSG_1 to N_1, which is supposed to respond with message MSG_2. In the inconvenient case, the wave goes from group 1 to n and on reception of MSG_1 N_1's sending slot is already over. Hence, MSG_2 has to wait for the duration of d_{SFL} until it will reach N_2. CUPID is especially suited for communication in two (preferred) directions, towards or from a reference node. Multi-hop communication orthogonal to the wave takes disproportionately more time.

C. Complex network topologies

More complex network topologies, as shown in Figure 4, lead to waves moving towards each other, which increases the probability of message loss due to hidden terminals. Consider the example: Here, two wave fronts moving away from the sink collide at node A and two wave fronts moving towards the sink collide at node B.

In complex topologies with vertices and holes CUPID can be applied, but some settings systematically lead to collisions. Representing the topology by a graph, we observe that cycles result in these settings and vertices if the wave moves towards the sink.

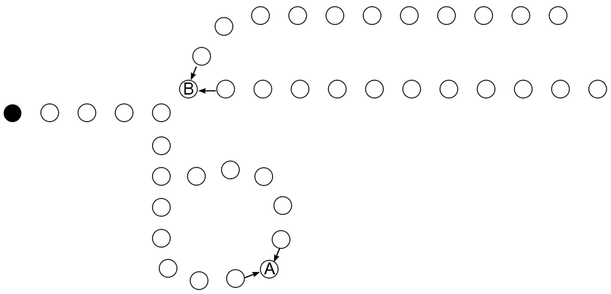


Figure 4. Example of a more complex topology

D. Installing CUPID at run-time

CUPID is divided into the start-up phase that is executed after network deployment and the subsequent operating phase describe above. We assume that the values for Duty Cycle and Tolerance Period are known at compile-time. However, for a node to know the wake-up, transmit and sleep times, it must determine (during the start-up phase) the number of groups and its group number. We briefly present one possibility to obtain the values.

- **Group numbers:** A way to assign the group number to each node is to use shortest paths to the reference node. Therefore, the reference node initiates a tree construction. If all messages contain their hop based distance, each node knows its distance to the reference node. Figure 5 shows an example how S_1 initiates a tree construction, how a possible constructed tree looks like, and how the nodes assign their respective group numbers.

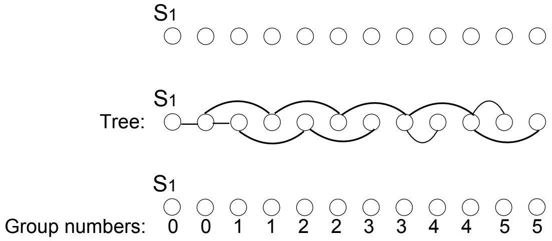


Figure 5. Assignment of group numbers

- **Overall number of groups:** If the network diameter cannot be estimated, each node sends a message to its parent node in the tree containing the maximum of its own group number and all previously received group numbers. The parent node forwards the message if the included value is greater than the actual local maximum. Like this, the maximum is forwarded to the reference node. In order to save messages, the further a node is located from the root of the tree the earlier it should send. The reference node then increments the maximum by one to obtain the overall number of groups and floods a message containing this number to inform all nodes. If the diameter is estimated by the user, the estimation should be greater than the real diameter by any means to ensure that every node can use its slot number. An overestimation simply increases the silence period, but does not invalidate slot numbers.

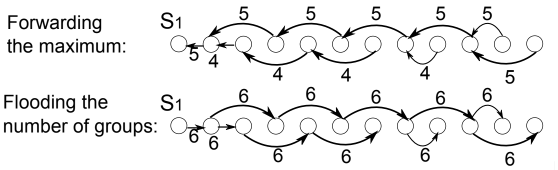


Figure 6. Detection of the number of groups

Algorithm 1 shows the computation of the slot and silence lengths during the start-up phase. First, the greatest possible slot length is determined as well as the resulting duty cycle. If this duty cycle is greater than the desired duty cycle, the slot length is recomputed using the desired duty cycle. Finally, the length of the silence period is computed.

Algorithm 1 Computation of CUPID's parameters

```
slot_len := (EED – 2*d_tol)/ND;
DC := (3*slot_len + 2*d_tol)*100/EED;
if DC > DC_DESIRED then
    slot_len := EED*DC_DESIRED/100;
    if slot_len > 2*d_tol then
        slot_len := slot_len – 2*d_tol;
        slot_len := slot_len / 3;
    else
        //Error: Invalid parameter combination!
        slot_len := 1;
    end if
end if
d_sil := EED – slot_len*ND – 2*d_tol;
```

In order to switch to the operation phase, the reference node floods another message containing the switching time T_C. In case of lower densities, but of higher risk of packet loss, additional broadcasts should increase the probability of a reliable message distribution. If new nodes join the network, they can either listen to the messages of their neighbors to learn their group number or the start-up phase could be repeated.

IV. Simulative Evaluation

To evaluate our approach, we conducted a set of simulations in the simulator Shawn [17] and compared CUPID to two other schemes representing the most commonly employed classes of duty cycling protocols. The first benchmark scheme called *SC-CSMA* (Synchronous Cycling with CSMA) is a simple synchronized duty cycling of all nodes, which access the medium by CSMA/CA during wake phases. Directly after waking up and before going to sleep, there is the same tolerance period like in CUPID to safeguard against clock drift. We call the second benchmark *SC-TDMA* (Synchronous Cycling with TDMA). Here, nodes are awake synchronously with the same tolerances like in both other schemes, but access the medium exclusively using a two-hop wide unique slot. To establish these slots we used the DRAND-algorithm [18].

Our goal was to model the real world as good as possible. The sensor nodes used for the real-world experiments are iSense nodes [19]. Software written for these nodes can be compiled for execution in Shawn, which provides also an IEEE 802.15.4 compliant CSMA/CA transmission model.

To model communication links, we employed the *stochastic communication model* where the packet reception probability remains constant at p_{max} up to the distance r_1 from the sender and decreases linearly from p_{max} to 0 for $r_1 < d < r_2$. We applied the values for $p_{max} = 98\%$, $r_1 = 28$ m, and $r_2 = 37.5$ m obtained from a set of real-world outdoor experiments.

The accuracy of the oscillators of iSense nodes is 20 ppm and by repeating time synchronization every 5 minutes, an offset of less than 12 milliseconds is guaranteed, so we set $d_{tol} = 12$ ms. To obtain statistically sound results, the results presented here are averaged over 100 simulations using the same parameter set but different random seeds.

A. Line-shaped topology

For our first simulations, we used a linear simulation area where nodes were placed in equal distances. We chose the following parameters of which we varied always one within each simulation set: $ND = 50$ hops, $DC = 1\%$ and the density was set to 10 neighbors on average to obtain a realistic working setup. As the influence caused by the chosen super frame length is expected to vary for different schemes, we altered the super frame length and then picked the best for each scheme.

At opposite ends of the simulation area, we placed two dedicated sink nodes S_1 and S_2 whereas S_1 triggered CUPID's installation as well as the operating phase start. We measured the EED of a single (flooded) packet from S_1 to S_2, which was scheduled at the beginning of one awake phase of S_1.

1) Influence of super frame length on latency: Figure 7 shows the average EED over the super frame length for the different duty cycling schemes. The lines indicate how the latency develops with an increasing super frame length. The vertical bars indicate minimum and maximum occurring value. Note that the larger the super frame length is, the larger is the awake phase, the slot length, and thereby the time for sending or forwarding a packet.

First, we have a look at the results of CUPID. For super frames shorter than 8 s, it depends on the random seed how

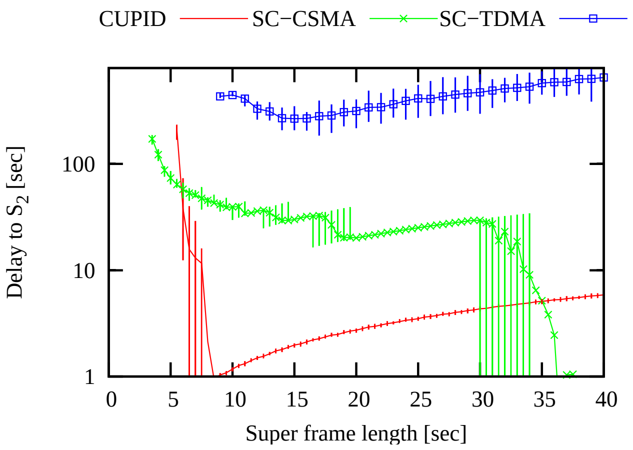


Figure 7. Average latency in the best case (diameter 50)

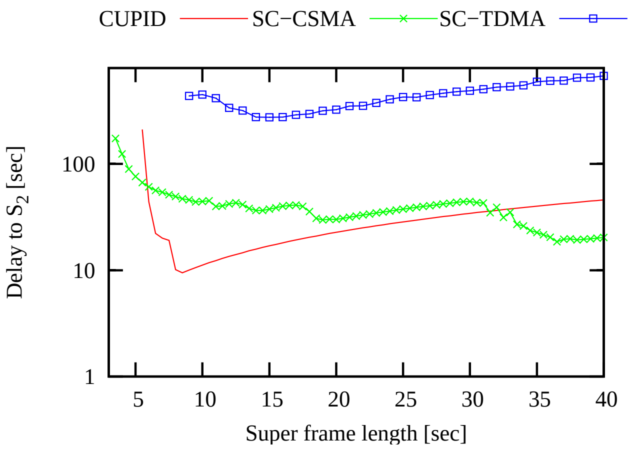


Figure 8. Average latency for an alarm message (diameter 50)

many cycles the flooded packet needs to arrive at the end of the network. At most it takes about 50 s, which corresponds to 7 cycles. The longer the wake phase is, the more likely it becomes that the packet is passed through the whole network within one cycle. 8 s as super frame length is obviously enough for the network diameter of 50 hops. The packet forwarding is done in only 942 ms. Increasing the super frame length leads to larger slot lengths and in particular in later slots (except the first one). This is why the delay of the packet slowly rises with an increasing super frame length.

The curve of the SC-CSMA scheme proceeds in stages because the message needs several awake phase to pass through the whole network and the longer the awake phase is the more hops are bypassed within one cycle. For larger super frame lengths than 40 s, the EED remains low because one awake phase is long enough to forward the packet over 50 hops. While CUPID features a delay between 900 ms and 5 s for super frame lengths between 10 s and 40 s, for SC-CSMA it becomes that low not before a super frame length of 38 s. This is especially harmful if the packet to be sent is an alarm message, which can be scheduled at any point in time. Hence, just in the case of an alarm where a low delay is important such a great super frame length affects the SC-CSMA in a negative way.

Figure 8 shows the average latency of an alarm message. Note that the time needed for forwarding the packet can be relatively short in comparison to the additional latency caused by waiting until neighbors have woken up. For CUPID it is 942ms vs. 10 s, which are caused by considering the random schedule time of an alarm.

SC-TDMA needs significantly more time (at least 260 s) to deliver the packet to S_2 because sending slots are not ordered in the forwarding direction. Hence, in the worst case the packet is forwarded only once during one wake phase. If it is clear, that the principle occurring communication pattern consists of messages in one direction, it is possible to adapt CUPID, so that the respective cycle is processed more frequently than the other cycle and the average EED is reduced further.

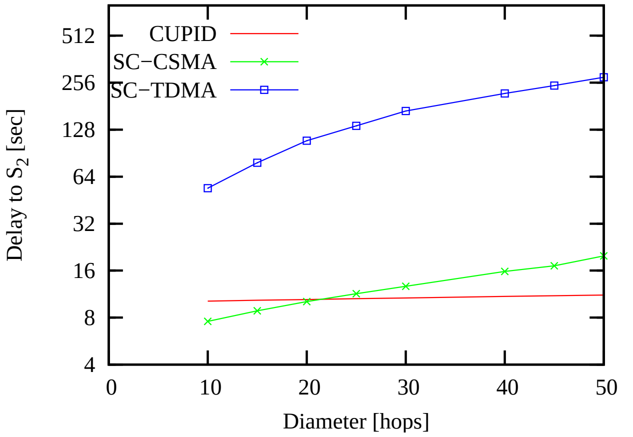


Figure 9. Average latency over diameter

2) Influence of network diameter on latency: Figure 9 illustrates the latency in the average (alarm) case over different diameters. We chose the optimal super frame length here for each scheme and each diameter. For CUPID, we set $d_{SFL} = 8$ s whereas for the other schemes $d_{SFL} = diameter + 20$ s holds.

We notice that in smaller networks with a diameter of less than 20 hops CUPID and SC-CSMA perform similarly, while SC-TDMA in contrast does not achieve the same efficiency. For larger diameters than 20, CUPID outperforms the other schemes.

3) Influence of density on latency: In contrast to different network diameters, other densities than 15 neighbors on average do not influence the latency of the packet very much. We set the density to 6, 10 and 20, but the results are similar

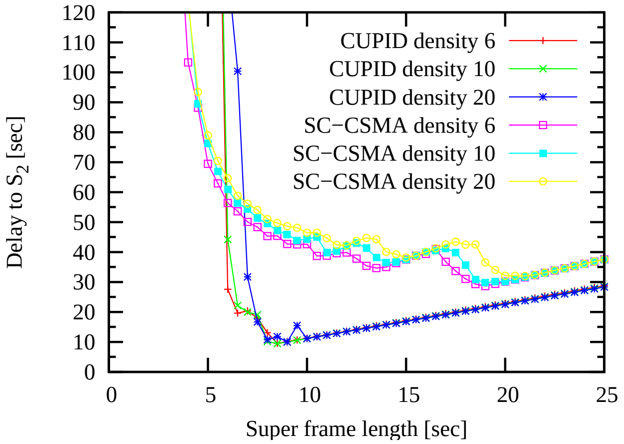


Figure 10. Influence of density

for all schemes. This is barely surprising because a higher density hinders the medium access, but hardly the latency until a packet is forwarded as always at least one node in every hop forwards the message.

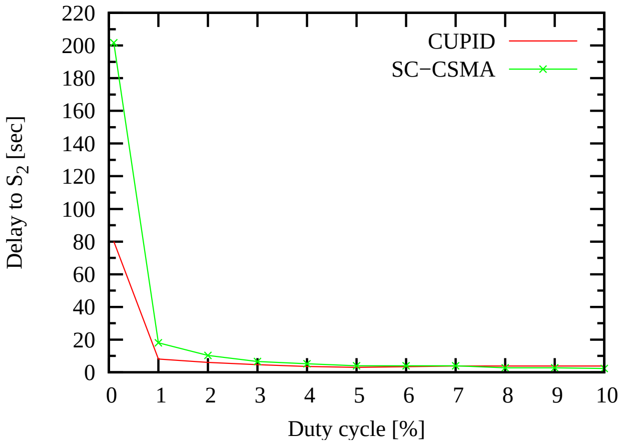


Figure 11. Average latency over duty cycle

4) Influence of duty cycle on latency: Finally, we investigated the influence of the duty cycle. Figure 11 depicts the latency of the packet for different duty cycles. It becomes clear that CUPID is beneficial for duty cycles lower than 2%. If energy is not scarce and a higher duty cycle may be used, it does not make sense to use CUPID, but especially for even lower duty cycles than 1% CUPID is extremely advantageous.

The following section presents the performance analysis of CUPID in the more general case, a two-dimensional topology.

B. Rectangle-shaped topology

The line-shaped topology may occur at borders or dikes, but to evaluate the performance of CUPID in other networks we conducted simulations with the scenario shown in Figure 12 comprising different structures. We placed 900 nodes in equal distances of 15 m in the simulation area so that we obtained about 10 hops in vertical and 30 hops in horizontal direction and about 20 neighbors on average.

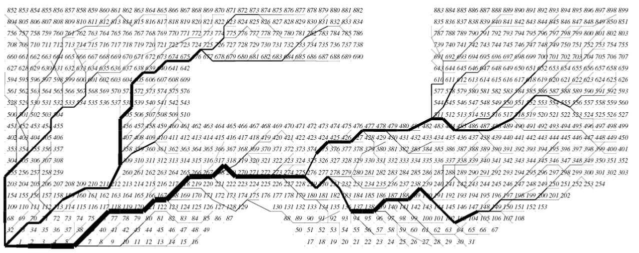


Figure 12. Rectangle scenario

Based on the assumption that sending measured data to a sink is a typical communication pattern in sensor networks, we let nodes send messages to the sink node in the lower left corner (using a tree routing). The resulting message density on each link is indicated through the thickness of the lines in Figure 12. We let different subsets of nodes (from 1 % to 20 %) of all 900 nodes send packets to the sink during the same wake phase. Each node of the subset sends one packet during this special wake phase resulting in 9 to 180 packets. We counted how many of these arrived at the sink and logged the delay. The super frame length was again set to 8 s. Along the tree, single hop acknowledgements are used, where each message is sent at most three times and is discarded after three tries. The lack of multi-hop acknowledgements leads to unreliable multi-hop communication.

The following two sections show the evaluation results concerning delivery ratio and the latency. We do not consider the SC-TDMA scheme in this scenario as it yielded in unacceptable high latencies.

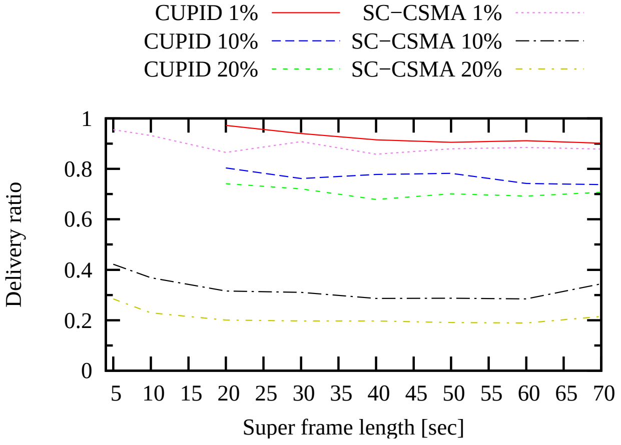


Figure 13. Delivery ratio

1) Delivery ratio: Figure 13 shows the receiving rate at S_1 for the different numbers of sent packets. SC-CSMA provides for all the percentages worse results. When 9 packets are sent, between 90 % and 95 % arrive at their destination. When 90 or 180 nodes send a packet only 30 % to 40 %, respectively 20 % to 30 %, arrive at S_1. The reason for this is that nodes that cannot hear each other send potentially at the same time to the same parent node leading to systematic collisions.

By contrast, CUPID transports 90 % to 98 % of the messages to S_1, if 1 % of the nodes send a message and between 70 % and 80 % if 90 or 180 nodes send a message. The regulated sending increases the delivery ratio, especially in case of high traffic volume.

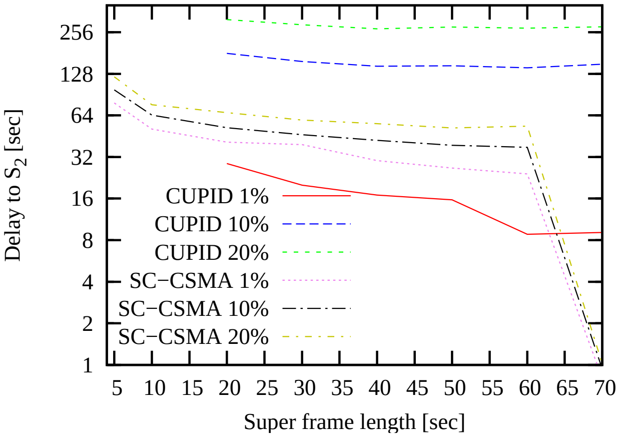


Figure 14. Average latency

2) Latency: The previous section showed that CUPID provides a good delivery ratio. Figure 14 shows the corresponding averaged latencies of the successfully delivered messages. Both schemes provided similar delivery ratios if only 9 messages are sent, but CUPID provides the shorter delays. Additionally, in case of a message scheduled at a random point in time, we must add the half of the super frame length on average. Moreover, shorter super frames allow for an earlier sending of another message. Using SC-CSMA a slot length of 700 ms is sufficient to transport the messages to the sink within one wake phase. As soon as the slot length is smaller the delay increases from 24 to 78 s. CUPID need only between 9 and 28 s.

The smaller values of the SC-CSMA scheme for 90 and 180 sent messages show that after a short time of traffic with many collisions no packets are forwarded or delivered anymore. CUPID delivers messages within up to 180 s, respectively 315 s. Of course, the more messages are sent the longer the delivery takes. However, CUPID reduces the probability of collisions since fewer nodes sent at the same time.

V. Real-world Experiments

In order to demonstrate the correctness of CUPID we conducted a series of real-world experiments using 156 iSense sensor nodes. As a comparison scheme we picked the better one of the benchmark schemes (SC-CSMA).

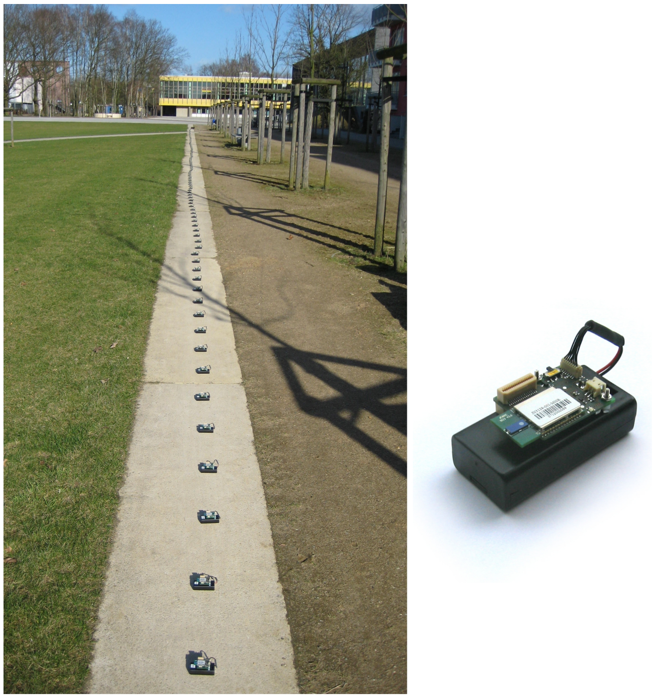


Figure 15. Experimental scenario (Second place)

In order to get a large network diameter, we reduced the sending power of the nodes by -30 db since otherwise, the length of the network would have been more than $1,5$ km. Using this sending power, the nodes were deployed in a row with an equal distance of 40 cm. We deployed the devices at two different places. First place was near a building where we got only 12 hops (due to many reflections and varying ranges), the second deployment was in a free area where we obtained 23 hops. After time synchronization and initialization we let one gateway node flood one packet at the beginning of a wake phase (leading to the best case) and logged its sending and arrival time at the farthest node. Figure 16 shows the results of this EED averaged over 120 packets.

It can be seen that in the first deployment near the building (diameter 12) the SC-CSMA scheme performs better as it can benefit from simultaneous wake phases. Only here are the temporarily larger communication ranges resulting in a smaller network diameter.

In the larger scenario without reflections CUPID performs better: It needs only 500 ms for the delivery of the packet, SC-CSMA needs 900 ms. Taken into account that an average offset of 10 seconds has to be added to these values for the

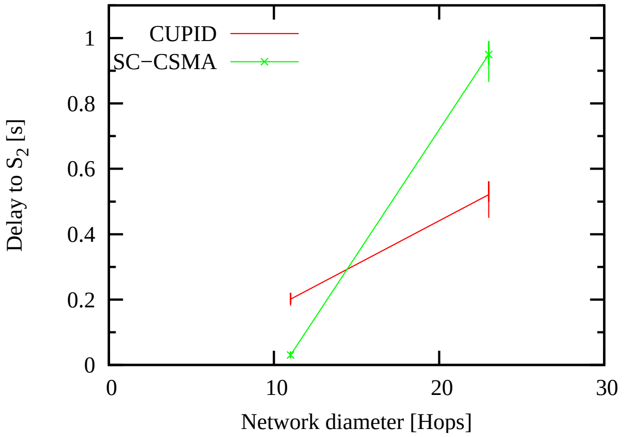


Figure 16. Latency for diameter 12 and 23

average case of an alarm packet, the results match exactly the simulation results shown in Figure 9.

VI. Conclusion

In this paper, we motivated, introduced, and evaluated a novel protocol called CUPID for low-delay, low-power, and attack-resilient operation of wireless sensor networks. Application domains include public safety or smart cities where large-scale, long-term, battery-driven operation, low-delay reporting of events and operation in potentially hostile environments are important. Our scheme is beneficial for any application that requires low message delivery latency and low duty cycles of 1% or even less at the same time.

CUPID is adaptable to different in-network communication patterns by the application designer and can be custom-tailored to application demands. In particular it supports communication from sink-to-network and vice versa, but is adaptable to the predominant communication pattern, duty-cycle settings and latency requirements. The scheme ensures that all neighbors of sending nodes are ready-to-receive while only assuming synchronized clocks to a certain extent. We show by simulations and real-world experiments with more than 150 nodes that our scheme significantly reduces the packet delay for low-duty cycle settings, especially in large networks.

While CUPID deals successfully with changing network structures caused by new or removed nodes as well as with unreliable links, it cannot cope with mobile nodes or very dynamic communication ranges. In future work, we will extend CUPID to semi-mobile scenarios where some nodes are mobile but the majority is static. In addition, we plan to evaluate CUPID in a larger-scale testbed such as SmartSantander, which will eventually support more than 10.000 nodes.

References

[1] D. Krüger, D. Pfisterer, and S. Fischer, “CUPID - Communication pattern informed Duty Cycling in Sensor Networks,” in *The Fifth International Conference on Systems and Networks Communications (ICSNC 2010)*. IEEE Computer Society Conference Publishing Services, 8 2010.

[2] A. Arora, P. Dutta, S. Bapat, V. Kulathumani, H. Zhang, V. Naik, V. Mittal, H. Cao, M. Demirbas, M. Gouda, Y. Choi, T. Herman, S. Kulkarni, U. Arumugam, M. Nesterenko, A. Vora, and M. Miyashita, “A line in the sand: a wireless sensor network for target detection, classification, and tracking,” *Computer Networks*, vol. 46, no. 5, pp. 605 – 634, 2004, Military Communications Systems and Technologies.

[3] C. Gui and P. Mohapatra, “Virtual patrol: a new power conservation design for surveillance using sensor networks,” in *Proceedings of the 4th international symposium on Information processing in sensor networks*, ser. IPSN ’05. Piscataway, NJ, USA: IEEE Press, 2005. [Online]. Available: http://portal.acm.org/citation.cfm?id=1147685.1147728

[4] P. Rothenpieler, D. Krüger, D. Pfisterer, S. Fischer, D. Dudek, C. Haas, A. Kuntz, and M. Zitterbart, “Flegsens - secure area monitoring using wireless sensor networks,” *Proceedings of World Academy of Science, Engineering and Technology*, 2009.

[5] Cambridge (MA) Smart City, “CitySense - an open, urban-scale sensor network testbed,” 2010, http://www.citysense.net [Last accessed: May 17, 2011].

[6] “Oulu Smart City,” 2010, http://www.ubiprogram.fi [Last accessed: May 17, 2011].

[7] German Telekom and City of Friedrichshafen, “Friedrichshafen Smart City,” 2010, http://www.telekom.com/dtag/cms/content/dt/de/217308 [Last accessed: May 17, 2011].

[8] The SmartSantander consortium, “SmartSantander,” 2010, http://www.smartsantander.eu/ [Last accessed: May 17, 2011].

[9] Y. Li, W. Ye, and J. Heidemann, “Energy and Latency Control in Low Duty Cycle MAC Protocols,” USC/Information Sciences Institute, Technical Report ISI-TR-595, August 2004.

[10] Q. Cao, T. Abdelzaher, T. He, and J. Stankovic, “Towards optimal sleep scheduling in sensor networks for rare-event detection,” in *IPSN ’05: Proceedings of the 4th international symposium on Information processing in sensor networks*. Piscataway, NJ, USA: IEEE Press, 2005, p. 4.

[11] N. Bisnik and A. A. Abouzeid, “Delay and capacity in energy efficient sensor networks,” in *PE-WASUN ’07: Proceedings of the 4th ACM workshop on Performance evaluation of wireless ad hoc, sensor,and ubiquitous networks*. New York, NY, USA: ACM, 2007, pp. 17–24.

[12] A. Keshavarzian, H. Lee, and L. Venkatraman, “Wakeup scheduling in wireless sensor networks,” in *MobiHoc ’06: Proceedings of the 7th ACM international symposium on Mobile ad hoc networking and computing*. New York, NY, USA: ACM, 2006, pp. 322–333.

[13] A. El-Hoiydi, J.-D. Decotignie, C. Enz, and E. Le Roux, “Poster abstract: wiseMAC, an ultra low power MAC protocol for the wiseNET wireless sensor network,” in *SenSys ’03: Proceedings of the 1st international conference on Embedded networked sensor systems*. New York, NY, USA: ACM, 2003, pp. 302–303.

[14] L. Gu and J. Stankovic, “Radio-triggered wake-up capability for sensor networks,” in *RTAS ’04: Proceedings of the 10th IEEE Real-Time and Embedded Technology and Applications Symposium*. Washington, DC, USA: IEEE Computer Society, 2004, p. 27.

[15] R. Zheng, J. C. Hou, and L. Sha, “Asynchronous wakeup for ad hoc networks,” in *Proceedings of the Fourth ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, 2003.

[16] Y. Taniguchi, N. Wakamiya, and M. Murat, “A traveling wave based communication mechanism for wireless sensor networks,” *Journal of Networks*, vol. 2, pp. 24–32, 2007.

[17] S. P. Fekete, A. Kröller, S. Fischer, and D. Pfisterer, “Shawn: The fast, highly customizable sensor network simulator,” in *Proceedings of the Fourth International Conference on Networked Sensing Systems (INSS’ 07)*, Jun. 2007.

[18] I. Rhee, A. Warrier, J. Min, and L. Xu, “Drand: distributed randomized tdma scheduling for wireless ad-hoc networks,” in *MobiHoc ’06: Proceedings of the 7th ACM international symposium on Mobile ad hoc networking and computing*. New York, NY, USA: ACM, 2006, pp. 190–201.

[19] coalesenses GmbH, “iSense,” http://www.coalesenses.com [Last accessed: May 17, 2011].

Performance Evaluation of a Disaster Recovery System and Practical Network Applications in Cloud Computing Envionments

Yoichiro Ueno, Noriharu Miyaho, Shuichi Suzuki
School of Information Environment,
Tokyo Denki University,
Muzai Gakuendai, Inzai-shi, Chiba, 270-1382 Japan
e-mail: ueno@sie.dendai.ac.jp,
miyaho@sie.dendai.ac.jp, ssuzuki@sie.dendai.ac.jp

Kazuo Ichihara
Net & Logic. Co.
Daizawa, Setagaya-ward, Tokyo, 155-0032 Japan
e-mail: Ichihara@nogic.net

***Abstract* — This paper presents evaluation results for a high security disaster recovery system using distribution and rake technology. In an experimental evaluation, the encryption and spatial scrambling performance and the average response time have been estimated in terms of the data file size. Discussion is also provided on an effective shuffling algorithm to determine the dispersed location sites. Finally, this paper describes a prototype system configuration for several practical network applications, including the hybrid utilization of cloud computing facilities and environments which are already commercialized.**

Keywords-disaster recovery; backup; metadata; distributed processing; cloud computing; strong cipher; secure video streaming

I. Introduction

Innovative network technology to guarantee, as far as possible, the security of users' or institutes' massive files of important data from any risks such as an unexpected natural disaster, a cyber-terrorism attack, etc., is becoming more indispensable day by day. To meet this need, technology is required that can be used to realize a system that has affordable maintenance and operation costs and provides high security.

For this application, Data Grid technology is expected to provide an effective and economical backup system by making use of a very large number of PCs whose resources are not fully utilized. In particular, Data Grid technology using a distributed file data backup mechanism will be utilized by government and municipal offices, hospitals, insurance companies, etc., to guard against the occurrence of unexpected disasters such as earthquakes, large fires and storms.

However, these methods involve high operation costs, and there are many technical issues to be solved, in particular relating to security and prompt restoration in the event of disasters occurring in multiple geographical locations.

In addition, there is a network infrastructure, which can be used to distribute and back up a great number of data files, and a large number of remote office personal computers, smart phones, or cellular phones, can be readily utilized for this purpose.

In addition to these factors, many people involved in industry and commerce are interested in making use of public or private cloud computing facilities/environments, provided by carriers or computer vendors, to provide temporary storage of and real-time access to their multimedia files, as a means of achieving security and low maintenance and operation costs. However, the guaranteed security level provided by cloud computer services has not yet been established.

In this paper we propose an innovative file backup concept which makes use of an effective ultra-widely distributed data transfer mechanism and a high-speed encryption technology, based on the assumption that we can use a small portion of the storage capacity of a large number of PCs and cellular phones that are in use in daily life, to efficiently realize safe data backup at an affordable maintenance and operation cost [1][2].

Figure 1 shows a comparison of the proposed method with the conventional method in terms of network utilization. The principal differences in the proposed system are as follows. (1) It does not require the use of expensive leased lines. (2) It only utilizes otherwise unused network resources, such as unused network bandwidth and unused storage

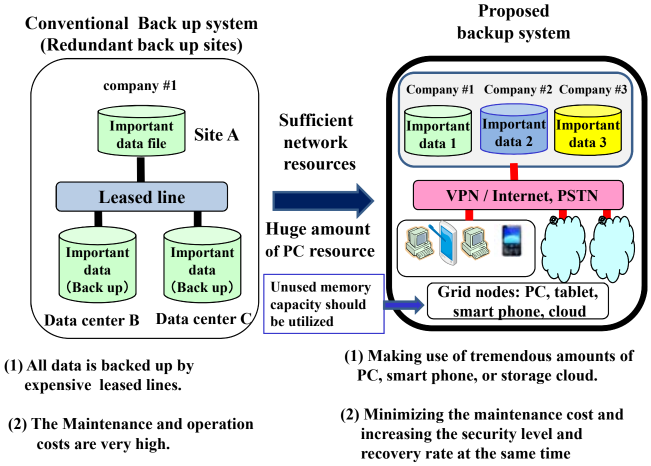

Conventional Back up system
(Redundant back up sites)
company #1
Important data file
Site A
Leased line
Important data (Back up)
Important data (Back up)
Data center B
Data center C
Sufficient network resources
Huge amount of PC resource
Unused memory capacity should be utilized
Proposed backup system
Company #1
Company #2
Company #3
Important data 1
Important data 2
Important data 3
VPN / Internet, PSTN
Grid nodes: PC, tablet, smart phone, cloud
(1) All data is backed up by expensive leased lines.
(2) The Maintenance and operation costs are very high.
(1) Making use of tremendous amounts of PC, smart phone, or storage cloud.
(2) Minimizing the maintenance cost and increasing the security level and recovery rate at the same time

Figure 1. Comparison of the proposed system with a conventional data backup system

capacity in PCs, smart phones and cellular phones, etc. (3) It can utilize cloud computing facilities/environments as one of the remote Grid nodes to obtain a requested amount of storage and a specific security level in accordance with the customer's requirements. (4) It can cipher a number of important data files at the same time using spatial scrambling and random dispatching technology. (5) As the number of user companies increases, the security against being deciphered illegally increases accordingly. (6) The maintenance cost can be drastically reduced. In addition, since it uses a stream cipher, the speed of encryption of data is increased, so it can also be applied to secure streaming for video transmission services.

In general, encryption can use two types of technology, that is, block cipher technology or stream cipher technology. In the case of block cipher technology, the data is divided into successive blocks and each block is processed separately for point-to-point systems; as a result, the encryption speed is quite slow. As the data volume increases, the required processor and memory cost increases in an exponential manner.

On the other hand, in case of the stream cipher, the input data is simply operated on bit-by-bit, using a simple arithmetic operation. Therefore, high-speed processing becomes feasible. These are the fundamental differences between the two cipher technologies [3].

When an ultra-widely distributed file data transfer technology, a file fragmentation technology and an encryption technology are used together, then quite different characteristics arise from a point of cipher strength. It is possible to combine the use of technologies, specifically, the spatial scrambling of all data files, the random fragmentation of the data files, and the corresponding encryption and replication of each file fragment using a stream cipher. The corresponding history data, including the encryption key code sequence, which we call "encryption metadata", are used to recover the original data. This mechanism is equivalent to realizing a strong cipher code, comparable to any conventionally developed cipher code, by appropriately assigning affordable network resources [4].

By making use of the proposed combined technology, illegal interception and decoding of the data by a third party becomes almost impossible and an extremely safe data backup system can be realized at reasonable cost. The proposed technology can also increase both the cipher code strength and the data recovery rate.

To realize the proposed disaster recovery mechanism, the following three principal network components are required: (1) a secure data center, (2) several secure supervisory servers, and (3) a number of client nodes such as PCs or cellular phones. We have previously clarified the relationships between data file capacity, the number of file fragments and the number of replications for the case of disaster recovery [5][6].

In this paper, we briefly describe the related work in Section II, and discuss the basic configuration of the system architecture in Section III, and the performance evaluation in Section IV. The practical experimental system is discussed in Section V. Finally, we provide our conclusions from these studies in Section VI.

II. RELATED WORK

Conventionally, a file data backup system [7] has been realized by duplication of a data center, access lines, etc. However, considering that an earthquake may cause fiber cable failures over a wide area and shut down several communication links, this approach is not fully satisfactory [8].

To take account of this, we have already proposed a disaster recovery concept which can use to realize a reliable file backup system by making use of safe and fast encryption mechanisms, a network distribution mechanism, and a technique for the effective use of secure meta-data containing the history of encryption key code sequences [9][10][11][12]. This technical approach has not yet been implemented elsewhere. Other related studies have included the concept of a distributed file backup system [13][14]. However, in these studies, neither a precise performance evaluation nor practical network service systems are clearly described. In addition, the technological aspects concerning the effective distribution of the fragmented data to establish the required security level are not clearly discussed for conventional systems.

On the other hand, there has been only a little published research on personal disaster recovery systems. A personal disaster recovery system should take account of the cost and the guarantee of security of personal data. So, a personal disaster recovery system may use some kind of cheap storage service for data backup. In recent years, it has become possible to implement data backup using free online storage. However, such low cost storage services are not secure enough for personal information. A persistent file server that should improve security of such storage service has been proposed [15]. This persistent file server system introduced encryption, fragmentation, replication, and scattering. Our proposed system adds more security with spatial scrambling, a second encryption, and an effective shuffling algorithm.

III. BASIC CONFIGURATION OF THE SYSTEM ARCHITECTURE AND ITS VARIATIONS

This section discusses the basic configuration of the high security distribution and rake technology (HS-DRT).

A. *Basic Configuration of the HS-DRT System*

The HS-DRT file backup mechanism has three principal components as shown in Figure 2. The main functions of the proposed network components are Data Center, Supervisory Server and various client nodes, and these can be specified as follows.

The client nodes (at the bottom of Fig. 2) are PCs, Smart Phones, Network Attached Storage (NAS) devices, and Storage Services. They are connected to a Supervisory Server in addition to the Data Center via a secure network.

The Supervisory Server (on the right in Fig. 2) acquires the history data, which includes the encryption key code sequence (metadata) from the Data Center (on the left in Fig. 2) via a network.

The basic procedure in the proposed network system is as follows.

1) Backup sequence

When the Data Center receives the data to be backed up, it encrypts it, scrambles it, and divides it into fragments, and thereafter replicates the data to the extent necessary to satisfy the required recovery rate according to the pre-determined service level. The Data Center encrypts the fragments again in the second stage and distributes them to the client nodes in a random order. At the sane time, the Data Center sends the metadata used for deciphering the series of fragments to the Supervisory Server. The metadata comprises encryption keys (for both the first and second stages), and several items of information related to fragmentation, replication, and distribution.

2) Recovery sequence

When a disaster occurs, or at other times, the Supervisory Server initiates the recovery sequence. The Supervisory Server collects the encrypted fragments from various appropriate clients in a manner similar to a rake reception procedure. When the Supervisory Server has collected a sufficient number of encrypted fragments, which is not necessarily all encrypted fragments, they are decrypted, merged, and descrambled in the reverse order of that performed at the Data Center and the decryption is then complete. Though these processes, the Supervisory Server can recover the original data that has been backed-up.

B. *Secuirty Level of HS-DRT*

The Security level of the HS-DRT does not only depend on the cryptographic technology but also on the combined method of specifying the three factors, that is, spatial scrambling, fragmentation/replication, and the shuffling algorithm.

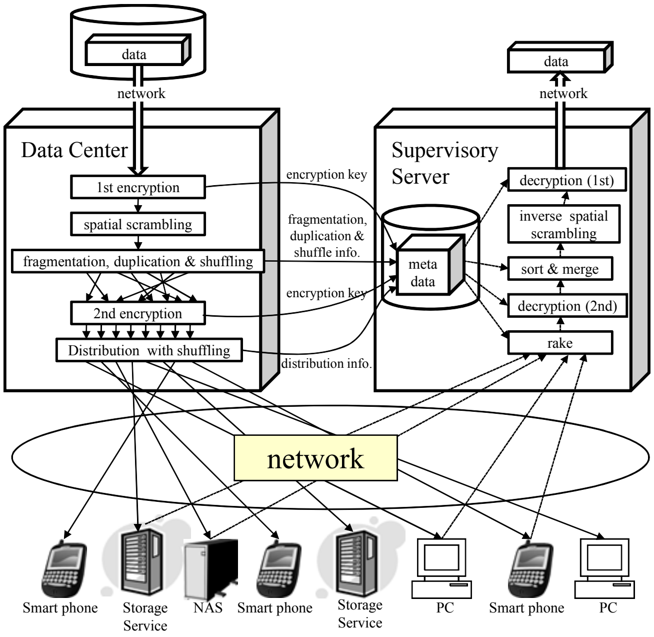


Figure 2. The basic configuration of HS-DRT system

Because of these three factors, nobody is able to decrypt without collecting all relevant fragments, selecting a unique set of fragments, and sorting the fragments into the correct order. Even if some fragments are intercepted, nobody is able to decrypt parts of the original data from such fragments.

1) Spatial scrambling

The spatial scrambling procedure can be realized by executing the simple algorithm illustrated as follows. Using a C-style description, we can write:

```
for(i=1;i<imax;i++){buf[i]=buf[i]+buf[i-1];}
buf[0]=buf[0]+buf[imax-1];
```

The array buf[] is the target data to be scrambled and imax is the size of the buf array. This computation process should be repeated several times. It is strongly recommended that this process be repeated at least six times. To de-scramble, it is only necessary to perform the same operations in the reverse order. By introducing the above mentioned spatial scrambling technology, it is almost impossible for a third party to decipher the data by comparing and combining the encrypted fragments, since uniform distribution of information can be achieved by this spatial scrambling.

2) Fragmentation/replication

One of the innovative ideas of HS-DRT is that the combination of fragmentation and distribution can be achieved in an appropriately shuffled order. Even if a cracker captured all raw packets between the data center and the client nodes, it would be extremely difficult to assemble all the packets in the correct order, because it would be necessary to try about (no. of fragments)! possibilities.

Furthermore, the proposed backup mechanism replicates each fragment and encrypts each copy of the fragment with a different encryption key. Even if a pair of encrypted fragments are the copy of the pre-encrypted fragment with each other, their bit patterns are completely different from each other owing to the different encryption key. Therefore, it is impossible to identify the encrypted fragment with the other. Crackers would require innumerable attempts to decipher the data.

3) Shuffling

HS-DRT mainly uses a shuffling method with pseudo-random number generators for the distribution to the client nodes. When we distribute the fragments of the encrypted data to widely dispersed client nodes, we can send them in a shuffled order, since we predetermine the destination client nodes from the shuffled table in advance. If the shuffled table has a uniform distribution, the table itself is hard for a third part to guess. But, if the shuffled tables are biased, there may exist weak points in the corresponding recovery system.

The result of the significance level, when we divide the data into 100 fragments, is as follows. The significance level for the Mersenne Twister and "Fisher-Yates shuffle" [16][17] with 1 round was 0.4617, and with 3 rounds was 0.8416.

HS-DRT adopts the Fisher-Yates shuffle with 3 rounds as the shuffling method and Mersenne twister as the pseudo-random number generator.

IV. PERFORMANCE EVALUATION

In this section we discuss the encryption performance and the spatial scrambling performance of the core module.

A. *Fundamental data*

We evaluated the performance of three systems. Figure 3 shows the test system, which consisted of two PCs, four Network Attached Storage devices, and three 1000base-T Ethernet networks. Table I shows the test environment adopted in each PC.

We examined the fundamental performance as follows.

By using Mersenne twister (mt19937ar), it was possible to generate a pseudo-random number in 7.35 nsec on PC1. By using the Fisher-Yates shuffle algorithm with 3 rounds and Mersenne twister, it was possible to generate 128 entries for a uniform distribution table in 18.0 μsec on PC1.

By using the 1000base-T (MTU=1522) network and a TCP stream, it was possible to transfer data from PC2 to PC1 at 112 MB/sec. In this paper, we define "MB (MegaByte)" as $(1024)^2$ bytes and "GB (GigaByte)" as $(1024)^3$ bytes, except for HDD capacity. Due to hardware limitations, we did not examine the performance using the jumbo frame size. By using the 1000base-T and FTP protocol, we found that we could achieve 33.9 MB/sec from PC1 to NAS for writing. By using 1000base-T, where the MTU size was 7422, we could achieve 42.6 MB/sec from PC1 to NAS for writing.

B. *Highest performance*

The HS-DRT encryption core module consists of three threads and four buffers. To examine the highest performance, the three threads, that is, the receiver, encryption/scrambling, and sender processes, were as follows. The receiver thread read the dummy data from "/dev/zero". This operation means that the receiver thread reads all zero dummy data without reading from the HDD or the network. The encryption/scrambling thread encrypts and scrambles the data. Then this thread divides the data into 128 fragments, and encrypts them in the shuffled order. The sender thread writes the 128 fragments in the shuffled order to "/dev/null". This operation means that the sender thread abandons all encrypted fragments.

Table II shows the performance of the HS-DRT encryption core module. The size of the buffers was 1GB each. "Time" means the actual elapsed time between the starting time of the first receiver thread and the finishing time of the final sender thread. The HS-DRT encryption core module was able to achieve 150MB/sec or more. This performance was better than that of the 1000base-T network interface of 119MB/sec.

C. *Practical performance*

The results obtained by examining the practical performance of the receiver/sender thread processes are as follows. The receiver thread reads the dummy data from the TCP stream that was transmitted from PC2. The sender

Table I. System environment

	PC1 Encryption core module	PC2 Data generator
CPU	Core2 Quad Q6600 2.40GHz	
Memory	8GB (DDR2-800)	4GB (DDR2-800)
HDD	RAID 0(striping) SATA 500GB 7200rpm×4	SATA 250GB 7200rpm
OS	Fedora 12 x86_64	Fedora 10 i686
Kernel	2.6.31.5-127.fc12.x86_64	2.6.27.5-117.fc10.i686.PAE
gcc	gcc(GCC) 4.4.2 20091027 (Red Hat 4.4..2-7)	gcc (GCC) 4.3.2 20081105 (Red Hat 4.3.2-7)
libc	glibc-2.11-2.x86_64	glibc-2.9-2.i686

Table II. Highest performance of HS-DRT encryption core module

Processed Data size [GB]	Time [sec]	Performance [MB/sec]
1	6.76	151
2	12.8	160
4	25.0	164
8	49.3	166
12	73.9	166
16	97.9	167
24	146	168
32	195	168
48	292	168
64	390	168

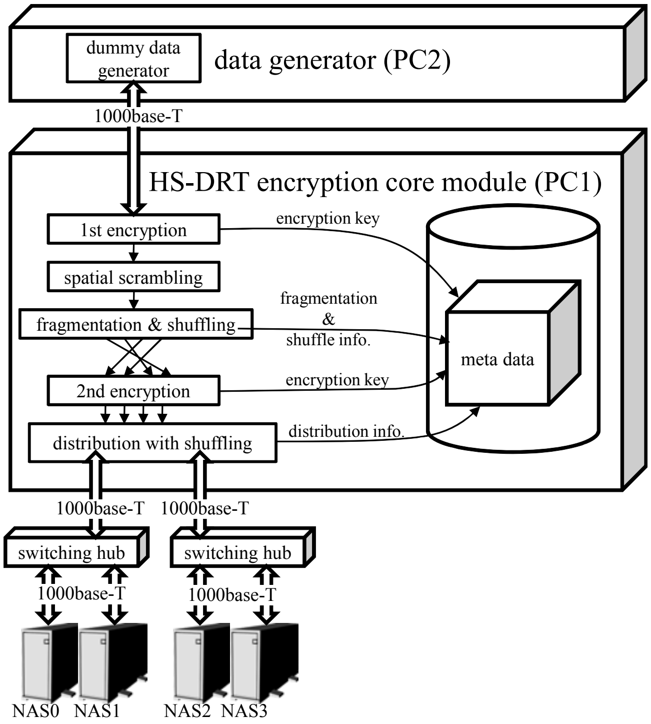


Figure 3. The performance evaluation set-up for the HS-DRT module

thread writes 128 fragments in a shuffled order using a built-in FTP client to four NAS devices. Since the sender thread creates new four threads, the HS-DRT encryption core module can write the encrypted fragments to the NAS devices in parallel.

Although most access lines to the Internet are slower than 100base-T, we used 1000base-T networks in the practical performance evaluations. There are two reasons. First, we hope to use gigabit access lines to the Internet in the near future. The second reason is the system requirement that the backup process should be completed immediately and should not be frustrated by a slow access line. So, the practical HS-DRT system should consist of two stages. In the first stage, the system processes are the 1st encryption, spatial scrambling, and fragmentation, carried out as fast as possible with local area high speed networks. After the first stage, the computer which has been backed up would resume its normal task. Then, in the second stage, the system processes are replication, shuffling, and distribution according to the speed of the access line. In our practical performance evaluation, we focused on the first stage performance.

Table III shows the practical performance of the HS-DRT encryption core module. “Sending Time” means the actual elapsed time for the data transfer from PC2. “Processing Time” means the actual elapsed time between the starting time of the first receiver thread and the finishing time of the final sender thread at PC1. The size of the buffers was 1GB each. “Latency” means the difference between “Sending Time” and “Processing Time”.

The Performance of PC1 changes according to the Processed Data size, and depends on the Latency. As the Latency consists of the final encrypting/scrambling time and the final sending time, it retains a constant value. Figure 4 shows the performance comparison of throughput on PC1. In Fig. 4, the X-axis shows the processed data size in Gigabytes and the Y-axis shows the performance in billions of bits per second. The upper line indicates the throughput of the encryption core module on PC1 under conditions for the highest performance and the lower line shows the same throughput under the practical performance conditions.

Table III.   practical performance of HS-DRT encryption core module

Processed Data size [GB]	Sending Time [sec]	Processing Time [sec]	Latency [sec]	Performance [MB/sec]	
	PC2	PC1		PC2	PC1
1	9.10	22.0	12.9	113	46.5
2	18.2	31.6	13.4	112	64.8
4	36.5	49.7	13.2	112	82.4
8	73.0	86.2	13.2	112	95.1
12	109	123	13.1	112	100
16	146	159	13.0	112	103
24	219	232	13.2	112	106
32	292	305	13.2	112	107
48	438	451	13.2	112	109

The performance degrades at smaller sizes of processed data because of the latency of processing and sending one GB of data. However, this latency becomes negligible at larger processed data sizes. Under the practical performance conditions, the throughput of the encryption core module on PC1 becomes saturated at one billion bits per second. The difference between this one billion bits per second and the total throughput on PC1 may be due to the overhead of Ethernet frame handling.

Figure 5 shows the CPU usage of PC1 and the threads execution timing chart when the HS-DRT encryption core module executes such operations as receiving, processing, and sending 4GB data. In the CPU usage graph in the upper half of Fig. 5, the X-axis indicates the elapsed time and Y-axis indicates the percentage CPU usage. As PC1 has a Quad-Core CPU, 100% of CPU usage means four or more processes/threads running in parallel, and 25% of CPU usage means only one process/thread running. The “IDLE” area shows the percentage of time that the CPU was idle. The “SOFTIRQ” area shows the percentage of time spent by the CPU in handling Soft-IRQs. The “SYS” area shows the percentage of CPU utilization at the kernel level. The “USER” area shows the percentage of CPU utilization at the user level. The total graph area, except for the “IDLE” section, indicates the total CPU workload. Any other parameters, such as the “NICE” value, are omitted from this graph as these values are negligibly small in this performance evaluation.

The “USER” area is the workload of the HS-DRT encryption core module, and the “SYS” + “SOFTIRQ” areas are the workload of the Operating System for receiving and sending.

The chart in the lower half of Fig. 5 shows the timing of the execution of the threads for the CPU usage graph. In this chart, there are twelve rectangles. The width of each rectangle represents the processing period of a thread. As mentioned above, the HS-DRT encryption core module consists of three threads, the receiver thread, the encryption/scrambling thread, and the sender thread. In this chart, the four rectangles that are marked “R#” show the

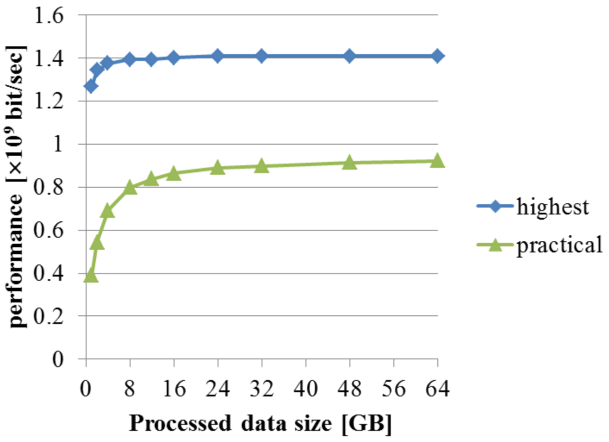


Figure 4. Comparison of throughput for PC1

processing period of the receiver threads, the four rectangles marked "E#" show the processing period of the encryption/scrambling threads, and the four rectangles marked "S#" show the processing period of the sender threads.

For example, the receiver thread received the first 1GB data from the network during "R1" period. Then the receiver thread passed the received 1GB data to the encryption/scrambling thread. The encryption/scrambling thread encrypted the first 1GB data during the "E1" period, and passed this 1GB of data to the sender thread. Finally, the sender thread sent the first 1GB data to the network during the "S1" period. So, the "R1", "E1", and "S1" rectangles form a row.

Looking at the process from a different perspective, the receiver thread received the first 1GB data from network during "R1" period, and passed the received data to the encryption/scrambling thread. Then, the receiver thread received the second 1GB data from network during the "R2" period, and so on.

In the processing period of the encryption/scrambling threads, the "USER" area of the CPU usage graph remains at about 25%. In contrast, the first (leftmost) processing period of the receiver thread and the last (rightmost) processing period of the sender thread consist of "SOFTIRQ" and "SYS" in the CPU usage graph.

From this CPU usage graph and the execution timing chart, it can be seen that the I/O ("SOFTIRQ" and "SYS") of the receiving and sender threads form the main bottleneck of the HS-DRT encryption core module. The proposed encryption core module can achieve the wire speed of gigabit Ethernet with half of 2.4GHz Quad-Core CPU power. So, the proposed method can realize a low cost encryption system with a commercially available cheap Dual-Core CPU.

We also examined the MPEG-2 (8Mb/s) video data transmission performance of the HS-DRT by using a 2.4 GHz Core2 Quad processor. In this case we assumed that the number of GOPs (Group Of Picture) is 15, and 30 frames/s video data transmission (which is equivalent to 512 KBytes per GOP) is imposed. We confirmed that the required encoding time including the encryption and spatial scrambling is 3ms, and the required decoding time including the decryption and descrambling is also 3 ms. This means that the total processing time of 6 ms, which corresponds to the total transmission latency, is sufficiently small and is within the specified standard field time (for example 16 ms).

V. PRACTICAL EXPERIMENTAL SYSTEM

In this section, we describe an application for secure video streaming and an application example of the HS-DRT core module used with a cloud computing system.

A. Application of HS-DRT for secure video streaming

HS-DRT is a versatile technology for secure data transmission. One of the network applications of HS-DRT is a secure video streaming service. Figure 6 shows an implementation example of a secure camera monitoring system using HS-DRT. In the streaming sender (on the left-hand side of Fig. 6), the picture frames or GOP of the MPEG video from the camera are encrypted, scrambled, and divided into "m" pieces. Moreover, each of the "m" pieces is further divided into "n" fragments. These (m×n) fragments are sent via (m×n) TCP/UDP streams. At the time of sending, each fragment is assigned the appropriate destination TCP/UDP flow port number by using the shuffled table. In the streaming receiver (on the right of Fig. 6), the "m" pieces are assembled by sorting and merging (m×n) fragments. The receiver merges and descrambles them in the reverse order,

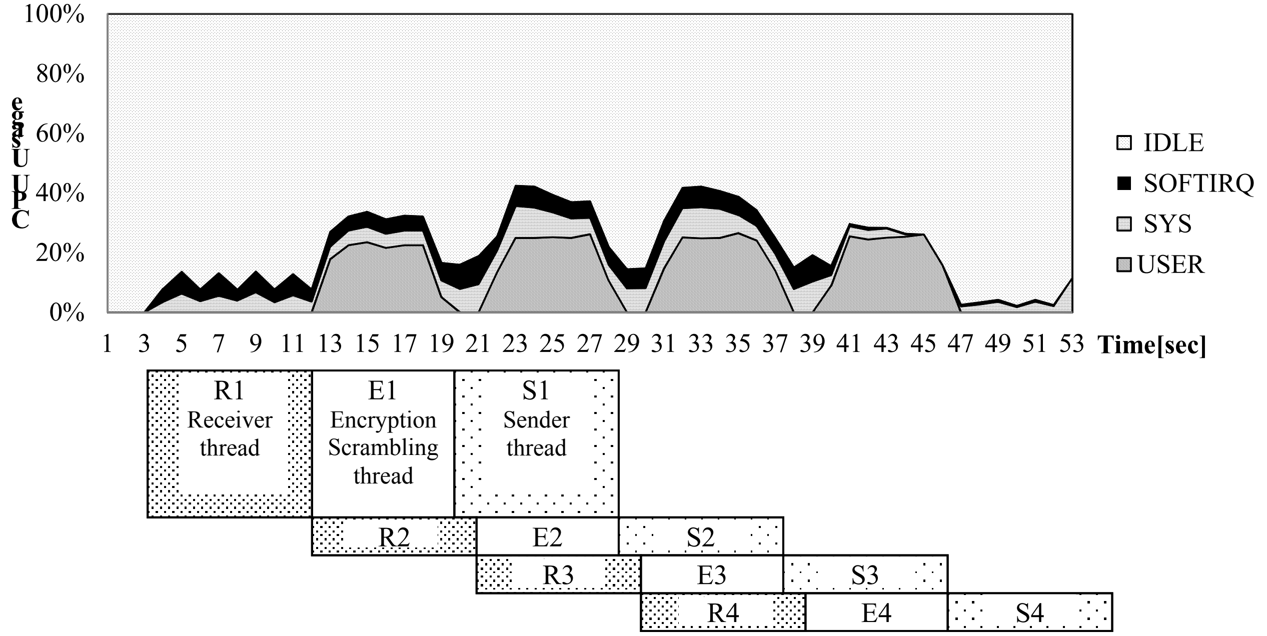


Figure 5. CPU usage on PC1 under practical performance conditions with 4GB data and thread execution timing chart

and decrypts "m" pieces to recover the original captured data. This implementation is considered to be a special case, because the grid nodes and the supervisory server was integrated in the streaming receiver on the right. In this implementation, the sending side and the receiving side have to share secret keys. These shared secret keys consist of the encryption key, the information regarding fragmentation, and the shuffle tables.

B. *Integration with a cloud computing system*

1) *System Implementation*

Figure 7 shows the configuration of one of the practical experimental systems to realize a hybrid HS-DRT processor by making use of a cloud computing system at the same time. As shown in Fig. 7, the system mainly consists of the following four parts: thin clients, a web applications server (Web-Apps Server), an HS-DRT processor, and Storage Clouds. Thin Clients are terminals which can use the web applications in the SaaS (Software as a Service) environment. Thin Clients can make use of the application services which are provided by the web applications server. The HS-DRT processor is considered to be a component of the hybrid cloud computing system, which can also strengthen the cloud computers' security level at the same time. The data center and the supervisory server can be integrated in the HS-DRT processor. The HS-DRT processor can effectively utilize the storage clouds as grid node facilities.

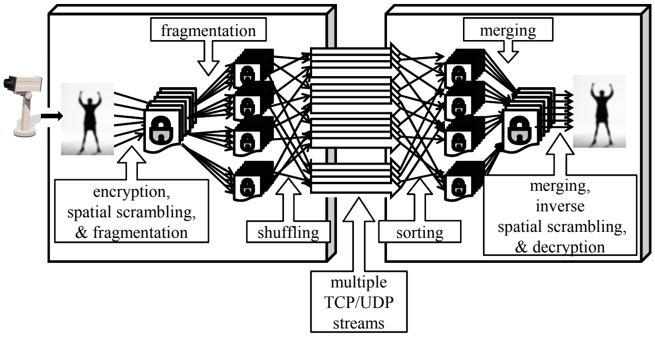


Figure 6. Secure video streaming with HS-DRT

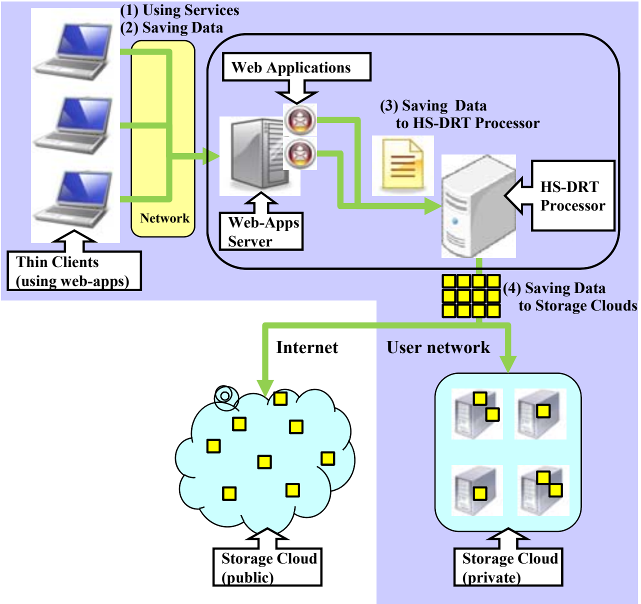


Figure 7. System implementation of HS-DRT processor with cloud computing system

For example, when a user wants to make use of the Web application function for storing specific individual items of data automatically, the user selects the corresponding function, such as "store", or "automatic store", as shown in (2) in Fig. 7.

When the user wants to store the data using the specific HS-DRT processor, the corresponding operation can also be handled by the corresponding user operations, by using well defined GUI operations as shown in (3) in Fig. 7.

In the HS-DRT processor, the HS-DRT-engine, which also has a function related to a web application, executes the encryption, spatial scrambling, and fragmentation of the corresponding files. It sends the corresponding encrypted data fragments to the public or private storage cloud computing system, where they are stored. The choice of whether to use a private cloud or a public cloud follows pre-determined criteria depending on the type of the web application.

2) *HS-DRT Processor structure*

Figure 8 shows the data processing model inside the HS-DRT processor. In Fig. 8, the HS-DRT processor is connected to web applications server on the left, and connected to some cloud computing storage systems on the right. The saved data flows from left to right via the HS-DRT encryption core in the upper half of the HS-DRT processor, and the user data provided in response to a load request flows from right to left via the HS-DRT decryption core in the lower half of the HS-DRT processor.

The data from the web application is first received at the engine, which is equipped with a distributed cache. This cache enhances the response time of the HS-DRT processors to the web applications server in both saving and loading operations. The data which is to be stored is sent to the HS-DRT encryption core and thereafter, encrypted, spatially scrambled, and divided into fragments. Following this, the HS-DRT distributes the data by transmitting the fragments.

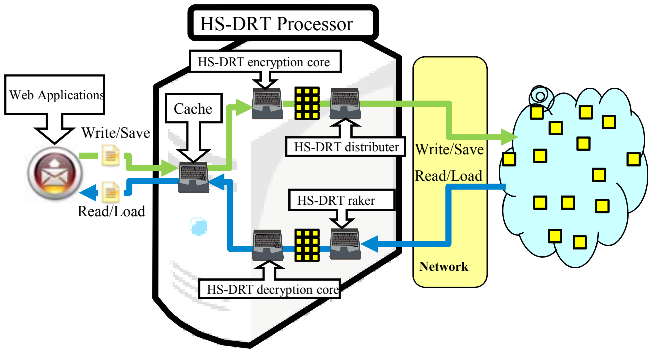


Figure 8. HS-DRT Processor structure

When there is a request from the web applications to load data, the cache engine first processes the data. The HS-DRT decryption core engine indicates that the HS-DRT-raker should collect back the distributed data from the relevant storage clouds. The engine then sends the collected data, which can be used for file data recovery if required.

3) Characteristics of HS-DRT processor integrated with the cloud computing system

It is very important to note that the processing efficiency of the HS-DRT processor can easily be improved by increasing the amount of the web cache memory.

We need to consider the scalability of the engine, since it may become a bottleneck in a very large system, owing to the number of clients and the amount of storage. In such cases, the HS-DRT processor may use a key-value database. As the HS-DRT processor can easily work with other HS-DRT processors, the system can be extended. The secrecy of the system can easily be assured because there is no plain raw data stream appearing anywhere in the entire data processing procedure.

On the other hand, we should take the following disadvantageous points into account.

- In order to fully utilize the HS-DRT processor, the web applications need to be well adjusted to be used by the HS-DRT engine itself.
- If the number of replicated copies of file data increases, the corresponding processor performance for executing the web applications will be degraded accordingly.

As for the HS-DRT processor, it can include the function of the web application server in addition to the encryption, spatial scrambling, and fragmentation of the corresponding file data. Scalability can also be easily attained by making use of conventional technologies for load balancing and multiplexing of the processor.

VI. Conclusions

We have presented an experimental evaluation of the encryption and the spatial scrambling performance of the proposed data recovery system, and found that the average response time in terms of the file data size is sufficiently practical to realize the corresponding network services.

Discussion has also been provided on an effective shuffling algorithm using Mersenne Twister and "Fisher-Yates shuffle" to determine the dispersed location sites.

Finally, this paper has described a prototype system configuration for several practical network applications, in particular, implementing a hybrid structure by making use of cloud computing facilities and environments which have already been commercialized.

Further studies should address the optimum network utilization technology. We are planning to verify the essential characteristics necessary to fully utilize the network resources to commercialize an ideal disaster recovery system.

Acknowledgments

This work has been partially supported by the Grants-in-Aid for Scientific Research (Issue number: 21560414), the Research Institute for Science and Technology of Tokyo Denki University (Q09S-10) and a grant of Strategic Research Foundation Grant-aided Project for Private Universities from Ministry of Education, Culture, Support, Science, and Technology, Japan (MEXT), 2007-2011.

References

[1] Y. Ueno, N. Miyaho, S. Suzuki, and K. Ichihara, "Performance Evaluation of a Disaster Recovery System and Practical Network System Applications," ICSNC 2010, pp. 195-200, Aug., 2010.

[2] N. Miyaho, Y. Ueno, S. Suzuki, K. Mori, and K. Ichihara, "Study on a Disaster Recovery Network Mechanism by Using Widely Distributed Client Nodes," ICSNC 2009, pp. 217-223, Sep., 2009.

[3] S. Suzuki, "Additive cryptosystem and World Wide master key," IEICE technical report ISEC 101(403), pp. 39-46, Nov., 2001.

[4] N. Miyaho, S. Suzuki, Y. Ueno, A. Takubo, Y. Wada, and R. Shibata, "Disaster recovery equipments, programs, and system," Patent. publication 2007/3/6 (Japan), PCT Patent :No.4296304, Apr. 2009.

[5] Y.Ueno, N.Miyaho, and S.Suzuki, "Disaster Recovery Mechanism using Widely Distributed Networking and Secure Metadata Handling Technology," Proceedings of the 4th edition of the UPGRADE-CN workshop, Session II: Networking, pp. 45-48, Jun., 2009.

[6] K. Kokubun, Y. Kawai, Y. Ueno, S. Suzuki, and N. Miyaho, "Performance evaluation of Disaster Recovery System using Grid Computing technology," IEICE Technical Report 107(403), pp. 1-6, Dec., 2007.

[7] NTT-East "Wide area disaster recovery services", <http://www.ntt-east.co.jp/business/solution/security/dr/> 23.05.2010

[8] Y. Kitamura, Y. Lee, R. Sakiyama, and K. Okamura, "Experience with Restoration of Asia Pacific Network Failures from Taiwan Earthquake," IEICE Transactions on Communications E90-B(11), pp. 3095-3103, Nov. , 2007.

[9] S. Kurokawa, Y. Iwaki, and N. Miyaho, "Study on the distributed data sharing mechanism with a mutual authentication and meta-database technology," APCC 2007, pp. 215-218, Oct., 2007.

[10] J. Yamato, M. Kan, and Y. Kikuchi, "Storage Based Data Protection for Disaster Recovery," The Journal of the IEICE 89(9), pp. 801-805, Sep. , 2006.

[11] Shanyu Zhao, Virginia Lo, and Chris GauthierDickey, "Result Verification and Trust-Based Scheduling in Peer-to-Peer Grids," p2p, pp. 31-38, Fifth IEEE International Conference on Peer-to-Peer Computing (P2P'05), 2005.

[12] K. Sagara, K. Nishiki, and M. Koizumi, "A Distributed Authentication Platform Architecture for Peer-to-Peer Applications," IEICE Transactions on Communications E88-B(3), pp. 865-872, 2005.

[13] S. Tezuka, R. Uda, A. Inoue, and Y. Matsushita, "A Secure Virtual File Server with P2P Connection to a Large-Scale Network," IASTED International Conference NCS2006, pp. 310-315, 2006.

[14] R. Uda, A. Inoue, M. Ito, S. Ichimura, K. Tago, and T. Hoshi, "Development of file distributed back up system," Tokyo University of Technology, Technical Report, No.3, pp. 31-38, Mar 2008.

[15] Y. Deswarte, L. Blain, J.-C. Fabre, "Intrusion tolerance in distributed computing systems," Research in Security and Privacy, 1991. Proceedings, 1991 IEEE Computer Society Symposium, pp.110-121, 20-22 May 1991

[16] R. A. Fisher and F. Yates, Statistical tables for biological, agricultural and medical research (3rd ed.). London: Oliver & Boyd. pp. 26–27, 1948.

[17] D. E. Knuth, The Art of Computer Programming, Volume 2: Seminumerical algorithms., 3rd edition, Addison Wesley. pp. 142-146, 1998.

Campus-Wide Indoor Tracking Infrastructure

Heinrich Schmitzberger and Wolfgang Narzt
Department of Business Informatics - Software Engineering
Johannes Kepler University Linz
Altenberger Straße 69, 4040 Linz, Austria
{heinrich.schmitzberger, wolfgang.narzt}@jku.at

Abstract — **In the context of large-scale indoor spaces location-based services still lack a feasible technology for localization and tracking in terms of ubiquitous operation. In this article, we present a tracking system based on wireless local area network infrastructure that is capable of simultaneously tracking numerous and diverse mobile clients (i.e., cell phones, laptops, personal digital assistants and alike) in multistory buildings within a campus facility at a near real time resolution and without client software to be installed. In the course of a real-life project at the campus of the University of Linz we have been studying deployment issues and environmental influences on infrastructure-based tracking in a large-scale setup that comprises various types of building structures and architecture. We contribute our findings regarding effects of arbitrary environmental conditions on radio signal based person tracking and present our current results. Furthermore, we demonstrate the feasibility of integrating infrastructural tracking technology into a location-based services platform called "Digital Graffiti" that handles user and privileges management while sustaining the privacy of the individual.**

Keywords-indoor infrastructural wlan tracking; scalability; large scale; real time; location-based services.

I. Introduction

At the core of mobile computing is the most prominent context information about the user: location. The user's current location as well as the awareness of the location of friends and things of interest have been a decisive driver for the ongoing trend towards smart phones and mobile applications up to date. The integration of the Global Positioning System (GPS) into commercially available mobile devices (smart phones and personal digital assistants for instance) carved the way for a broad variety of location-based services (LBS) covering outdoor spaces and public places in the form of navigation systems, location-based social networks and city guide applications. However, still no convincing counterpart regarding indoor location acquisition has prevailed in real-life LBS scenarios. In the last decade, several approaches towards realizing a ubiquitous indoor localization technology that could compete with the quality and reliability of outdoor GPS have been presented. The underlying sensor technologies were manifold (Bluetooth, Infrared or Ultrasonic, just to name a few). Concluding from the GPS story of success, a key factor of influence is the broad availability of the respective technology in a common mobile device. As we demonstrated in [1], WLAN (Wireless Local Area Network) technology seems to have the most promising potential in this context. We pointed out that the approach of exerting tracking infrastructure for the localization of a large number of concurrent users has shown feasible, even in terms of large-scale setups covering vast building complexes (e.g., the campus of a university). Using WLAN in this context offers certain benefits concerning costs, accuracy, scalability and deployment compared to other popular radio localization technologies, not least because of its availability in modern mobile phones [2].

Recently, commercial WLAN localization products have been introduced [3][4] following a localization method having the mobile device acting as sensor or even estimating its position itself (client-based). In this article, we propose a converse system where sensor hardware as well as position estimation is decoupled from the client, but achieved by a backend server combined with WLAN infrastructure within a complex of buildings (infrastructure-based).

This setup allows us to look into the subject of indoor localization from a different point of view, being able to support a vast range of client devices. Instead of providing software for different client platforms and sensor arrangements our infrastructure comprises a network of homogeneous, permanently active sensors that assure accurate measurements for convenient location estimation of numerous clients operating on several platforms. By this means we avoid forcing CPU prerequisites since infrastructure bears the processing load. Consequently, power consumption is reduced on the client side.

Another benefit of our setup is its robustness with respect to a constantly changing WLAN environment, a disadvantage that affects accuracy in client-based setups because they mainly rely on stable signal strength fingerprints of access points (APs) in the vicinity. Since the system solely depends on measurements of client signals additional WLAN signal sources have no consequences on the position estimation process.

The system presented in this article meets the following prerequisites: first of all, localization accuracy of room-level granularity is reached. The system is able to provide a mapping of geo-coordinates to symbolic names at detailed spatial level (such as "*Office 306, Management Center, University of Linz*"), which facilitates users' perception of location information. In order to attain a ubiquitous user experience, the system operates 24 hours a day, 7 days a week and is open to public access from within the wireless campus network. Accordingly, our architecture has to attend to scalability in terms of providing a near real time service to a potentially unbounded amount of users. Furthermore, we have developed a service based on our LBS framework that enables the finding and tracking of friends given their anytime revocable permission. In this context we refer to the notion tracking almost similarly as to localization, with one distinctive exception: localization describes the process in which the current location of a certain client is estimated. Tracking on the other hand uses previously calculated location estimates to create a traceable path that is further used to render the client's position more precisely and to conclude to the client's course.

The rest of this article is organized as follows. Section 2 discusses related work to the broad topic of indoor localization with respect to WLAN signal strength based approaches concentrating on infrastructure. In Section 3 we describe the architecture of our system in detail, focusing on its 3 main components in particular. Subsequently, Section 4 deals with our real-life setup in the course of the *Smart Information Campus* project at the University of Linz, discussing detailed facts of campus-wide deployment as well as the users' view of the system. In Section 5, we compare tracking results surveyed in different types of buildings at the campus. Finally, Section 6 concludes with a discussion on our efforts to realize the described system and gives a brief outlook on future research questions in context of our setup.

II. Related Work

In the last years, numerous contributions have discussed the subject of indoor localization and tracking on WLAN basis as an alternative to obtaining positions using GPS. Most of them concentrated on signal strength localization algorithms that emphasized a client-based application, i.e., the client device is operated as sensor collecting signal strength information of nearby APs. These systems broadly depend on special hardware in the form of tags [4][5] or native libraries supporting certain network interface card (NIC) features at client-side [6][7][8].

We envision a client-device independent setup in order to not constrain the usage of our system to certain hardware. To this end we have studied an infrastructure-based approach to achieving WLAN signal strength localization. In this context the notion "infrastructure-based" refers to a setup that comprises stationary sensor nodes for measuring client radio transmissions [9][10][11].

Early contributions to this approach are presented in [12]. The RADAR system uses a setup of ordinary PCs as stationary signal strength sensors. This experimental setup showed the feasibility of WLAN for indoor localization. In [9], the LEASE system is proposed, an infrastructure-based framework using sniffers and reference-emitters. The sniffers are constructed as embedded sensor platforms. The emitters were used as signal strength references to constantly rebuild an active radio profile. The main focus of this work concentrated on deployment issues and consequences of radio propagation for location estimation in potential real life scenarios. A detailed description of the sniffers used for LEASE is given in [13]. In this article, the issues of capturing conventional WLAN traffic as well as the placement of sniffer nodes for convenient location estimation are explored.

Embedded devices have also been the basis for Pinpoint [14], a system that uses the Time-Of-Arrival localization method. In order to achieve accurate estimations, the main functionality of their devices was to maintain high precision clock synchronization. The usage of Linksys WRT54 (the previous generation of the devices used in our work) as platform for localization applications has been a matter of research as well. In [15], customized kismet software is used on a Linksys device to report signal strength measurements. The authors report experiments in a calibration-free localization setup. In [16], another calibration-free fingerprinting system is proposed that applies probabilistic methods for constructing radio profile model and position estimation. Due to heavy computational load it uses PCs as sniffers. In this work another focus lies on WLAN channel characteristics and fluctuations in signal strength.

However, all the systems depicted above discuss the infrastructural localization on a prototypical basis. Most of them solely cover a 2-dimensional area of interest; often a dedicated test bed is constructed to make a proof of concept. The work presented in this article reports experiences of a campus-wide real life setup comprising multistory buildings of various characteristics at the *Johannes Kepler University* in Linz. In this context we experimented with commercial products as well. Since the campus-wide WLAN network uses Cisco Aironet 1250 devices we evaluated a trial setup of the Cisco Wireless Location Appliance (Version 3.1.35.0) based on RSS fingerprints [17]. Our findings were that the update frequency of the clients' location was too low (one estimate in 30 seconds up to 1 minute) for our purposes, since it solely listens for client probe request frames. Our system in contrary uses a near real time resolution (updated every 3 seconds). Furthermore, we strongly focus on a scalable, robust system addressed to a public audience in order to provide a convenient LBS experience.

III. WLAN-based Tracking Infrastructure

Our main design objective was to implement a system capable of concurrently tracking multiple mobile clients within multistory buildings across a campus areal. We aim at serving a vast variety of mobile devices not requiring specific hardware but a WLAN interface. This had two profound implications on system architecture. First of all, the sensor component of the system is implemented infrastructural to avoid heterogeneity of measuring data. And second, the client's radio communication with the system can be reused as signal data for position estimation.

Even though indoor WLAN localization systems have been widely studied in the last decade most contributions concentrated on a client-based approach. With few exceptions (cf. Section 2) infrastructure-based WLAN localization has been relegated to a niche existence [13]. This is mainly due to the cost for an area-wide infrastructure deployment (i.e., purchasing sensor hardware, permanent power consumption, setup and maintenance costs and the alike) on the one hand [11], as well as privacy and security concerns on the other hand [18]. In fact, it is easier to sustain user privacy in a client-based localization setup since the respective information is computed on the client side. But if client-side contextual data is transmitted to an untrusted server for LBS consumption (as in most public application scenarios), this advantage does not remain existent.

Another important concern is system scalability. If location estimation for each client is calculated on a central backend server and not on each client device individually, the computation load poses an obvious bottleneck. A main focus of our research lies on this aspect. To overcome the scalability issue we explored the potential of the sensor infrastructure itself, further discussed in Section 3B.

Opposing these difficulties that arise as a consequence of our design decisions we have to point out some benefits as well. First and most of all, the system is able to operate without any client pre-requisites but a WLAN communication interface. At the moment we solely support WLAN 802.11b/g/n (802.11a/n support is a work-in-progress at the moment). This implies that every mobile device equipped with such an interface currently on the market is able to use our tracking service. Since consuming the localization service can basically be done via a web request, it poses an energy saving alternative compared to GPS for instance. Furthermore, no additional software needs to be installed at client-side. The system presented in this article consists of three main components (cf. Fig. 1) discussed in the following sections.

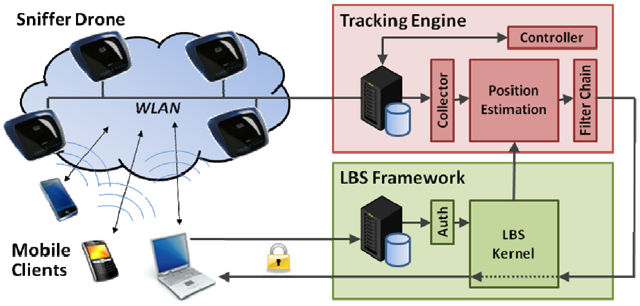


Figure 1. System Architecture

A. Tracking Engine

Our architecture bases on a two-phase signal strength fingerprinting system implemented in Java and hosted on a server backend. Every physical position used for location estimation is represented as a vector consisting of several tuples of a certain signal strength paired with a MAC address (cf. Table 1).

TABLE I. POSITION VECTORS

Position			Client Fingerprints	
LON	*LAT*	*ALT*	*Sniffer MAC*	*RSS*
14.3182073	48.3363918	5.0	00:23:69:3B:2C:A7	-42
			00:23:69:3B:2C:FF	-62
			00:23:69:3B:2E:E7	-78
14.3180591	48.3363577	5.0	00:23:69:3B:2C:FF	-58
			00:23:69:3B:2E:AF	-62
			00:23:69:3B:6D:77	-81

Traditionally, the MAC address identifies an AP that is observed by the client. Since the infrastructure-based setup estimates positions observing client transmissions on the contrary, the MAC address identifies the sniffer device that reads the client's signal strength. A database holding all of these positions (candidate points) is created in the initial training or offline phase, representing a radio profile for the targeted physical space. Fig. 2 (left) shows a clipping of an office floor with its respective candidate points. The colors of the candidate points indicate the quality of the signal level (green: ≥ 5 sniffer receptions, blue: ≥ 3, red: < 3). As the figure illustrates, each office comprises 2 to 4 candidate points with varying signal quality. To cope with several different transmitter characteristics (antenna properties, transmitting power) we created an individual radio profile for each supported device type (laptop, cell phone, PDA).

In the online phase, location estimation is computed with the commonly used Nearest-Neighbor-In-Signal-Space (or kNN) algorithm [12] that queries the database for the *k* best matches with the least Euclidean distance to the client's current signal vector. This approach has been discussed and proven feasible for position estimation in numerous publications [19][20][21] and won't be explained further. An important factor for the estimation quality especially when dealing with large-scale setups in this context is the accurate weighting of both the client's signal vector entries and the candidate points vector entries, which decisively accounts for the localization result. This weighting has to reflect the density of sensors in the vicinity of the candidate point. Consequently, it uses *(i)* the amount of different sniffer entries forming one fingerprint vector, *(ii)* the averaged signal level of the fingerprint and *(iii)* the actual strength of each entry in the vector to reflect the probability of the appearance of the signal at the respective position. If a vector entry is missing in comparison with the database, it has to be taken into account as well.

Since system scalability is of most importance we don't use probabilistic localization approaches that might provide better accuracy for account of CPU load, as the HORUS system [8] does for instance.

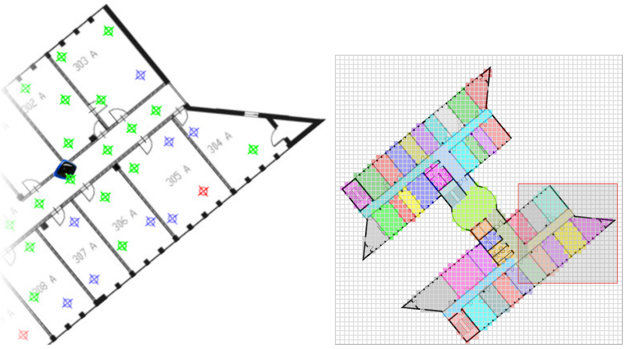


Figure 2. Candidate points (left), floor sectors (right)

Using signal strength (i.e., the RSSI value of the sensor device's WLAN NIC) for client localization has practical implications on the tracking engine. First of all, signal strength measurements underlie fluctuations. As [12][16] reported, multipath signal propagation and other propagation effects such as reflection, refraction, and scattering falsify the signal strength measurement. If not filtered (cf. Section 3B) or handled using appropriate mathematical models as the Wall Attenuation Factor (WAF) model [12], these effects can produce grave estimation outliers that have to be compensated later in the processing chain. Additionally, a person or a group of people between the measurement device and the transmitter can pose a dynamic signal attenuation component in the measuring process affecting especially radio signals at 2.4GHz frequency. Consequently, a daytime-dependent fluctuation in public spaces can be observed [22]. We are currently developing a mechanism to compensate such fluctuations by monitoring AP beacon signals.

Within an interval of 3 seconds (which corresponds to two channel hop cycles, cf. Section 3B) the engine calculates a position estimate. We represent a position according to the World Geodic System WGS84 as spherical polar coordinates in longitude, latitude and altitude. The estimate then passes a chain of filters. The first filter evaluates the plausibility of the estimate by testing if the position is within an accessible area of a building modeled in the database. Each building has at least one floor represented by a north aligned floor map and partitioned into several sectors (cf. Fig. 2 right). Each sector models a distinct type of an enclosed area, as rooms, hallways, stairs or elevators. If the position estimate is within a sector modeled as invalid, the estimate is identified an outlier and is not considered further.

The next filter bases on a set of rules that specify possible transitions from one sector to another. The rule set takes into account the building's characteristics, modeling direct paths, sector connections and next hop neighborhoods. Changing floors is only possible within a stairway or elevator sector. At this point, the track of the client's last positions is considered for stabilizing purpose as well, i.e., to ensure that a client is not considered moving if he is not.

Alternatively, we experimented with a particle filter algorithm [11][23]. Results showed that the filter did not perform much better in compensating outliers. Due to a large increase in computational demand we consider this alternative as not feasible for a large scale setup.

B. *Sniffer Drones*

The work presented in this article emphasizes enhancing WLAN infrastructure with a sensor overlay consisting of of-the-shelf network devices. In related publications these devices are often referred to as sniffers [9][16] or sniffer drones [15]. Our architecture comprises a network of such sniffer drones, forming the core component of the system. In general, sniffers can be denoted as passive components, meaning that they do not emit radio signals themselves since they use an Ethernet backbone to drain their measurement data. Currently, we're running a campus wide sniffer network consisting of custom Linksys WRT610N APs with some modifications to firmware and software.

Basically, these devices can be considered as embedded systems, operating a MIPS32 platform at 466MHz with 8MB RAM and providing two separate WLAN interfaces to support both 2.4GHz and 5GHz radio. A Linux kernel 2.6 [24] allows developing applications at system level. Our sniffing software (implemented in C) uses the low-level packet capturing library *libpcap* [25] that addresses a feature of the WLAN interface driver to collect signal strength measurements, the so called monitoring mode. Monitoring mode describes an alternative mode of operating a WLAN interface (such as the master mode for access point operation or the managed mode for client operation) and is a mandatory feature that facilitates using access points as sensors. This way sniffing the wireless medium in real time is possible, which is crucial to our architecture. First experiments we conducted made use of SNMP (Simple Network Management Protocol) for collecting signal strength measurements, as demonstrated by [26], for instance. A benefit was to be able to abstract from AP hard- and software as long as they supported SNMP. This approach however did not satisfy our demand for conducting real time measurements (cf. Section 2).

As the sniffer drones network is conceived as an overlay to existing WLAN infrastructure, we consequently have to deal with the usage of three non-overlapping channels in the 2.4GHz frequency band (typically the channels 1, 6 and 11 or 2, 7 and 12) - the 5GHz band will not be covered in this article. Hence, we can't assume the transmission channel of a mobile client precisely. Therefore, all possible channels are subsequently iterated while monitoring each prospective channel for a certain period of time. In order to assure collection of sufficient measurements per channel 500 milliseconds of channel dwell time have proven applicable. A completed iteration will further be addressed as a channel hop cycle.

In Section 3A we pointed out that a measurement vector consists of several entries, one for each drone that detects the client's signal at a certain place. To calculate a position these entries are proportionally weighted according to several properties, such as the signal intensity. Hence, the absence of a presumably intensive drone's measurement in this vector can lead to a grave estimation error. If it is not assured that adjacent drones concurrently listen on the same channel it is likely that the measurement vector is incomplete. To avoid this effect we implemented a synchronization mechanism that concurrently triggers restarting of the channel hop cycle at every sniffer drone in the network. The triggering component resides on the server backend, centrally orchestrating the sniffer network by sending a UDP restart broadcast every 60 seconds. In this context, we experimented with an alternative approach to avoid a hop cycle restart every minute. The Precision Time Protocol (PTP), as defined by the relevant IEEE 1588 standard [27], provides clock synchronization accuracy of less than one microsecond. Our embedded hardware platform on the contrary offers a system timer resolution (often referred to as *Jiffy*) of just 10 milliseconds. If we applied a predefined, hard coded hop cycle schedule on each Sniffer Drone along a daemon process updating the system clock via PTP on a daily basis, the maximum drift between the networked Sniffer Drones could be reduced to these 10 milliseconds, in theory. In practice, this has turned out not to work satisfactorily. Due to a varying workload on each separate sensor device correlated to differing radio environments, a measureable internal clock drift within the whole sensor network might appear already after a few minutes, especially between idle sensors and busy ones. Considering the protocol overhead produced by the PTP synchronization, an every minute clock update is not a better choice compared to the UDP restart solution.

During normal sensor operation, each single sniffer continuously reports its measurement data to a collector process within the tracking engine while constantly switching through the channel hop cycle. The sniffer is able to apply packet filters to reduce the subsequent processing load for the engine. System architecture implies that a user has to request a position from a frontend (cf. Section 3C) that operates on a special high port. This port is used as indicator for WLAN traffic to be tracked; traffic on other ports is filtered out. Filter functionality can also be activated remotely by a controller process within the tracking engine, for instance if a certain MAC address has to be blacklisted. As another additional load reduction effort, each sniffer groups measurement data by MAC addresses and averages them before sending.

C. *LBS Framework*

In the previous sections we explained how an infrastructural sensor overlay is used to estimate a user's location. This section deals with how to deliver this information to the user to achieve additional benefit. For these purposes we use the Digital Graffiti system.

Digital Graffiti is a stand-alone framework for location-based services developed in the course of a research project between Siemens Corporate Technology Munich, the University of Linz and the Ars Electronica Futurelab Linz. Conceived as a system to manage and visualize localized information within the context of a mobile user (respectively a mobile device) [28], it has been enhanced with functionality to fulfill the demands for a social network system as well. It comprises a map server, an elaborated user and privileges management concept that additionally handles communication encryption and a messaging component.

Similar to conventional cellular telephony the system uses a distributed provider model for the server-side component where users all over the world can join the provider of their choice in order to take part in the mobile location-based information service. This proven model distributes the load ensuing from (asynchronously) communicating users and guarantees scalability of the service all over the world as each provider only handles a limited number of clients. Information elements (graffiti) are stored in corresponding databases at the providers.

The clients are supposed to be executed on any mobile platform, either as a native application particularly designed for the device or as a web application (utilizing the novel W3C standard and HTML5 for accessing GPS out of a browser). In the context of infrastructure-based indoor tracking, we accentuate the web application, for it complies with our requirements of a bare device without the needs of installing client software. Fig. 3 illustrates the architectural layers of the Digital Graffiti client framework.

The framework utilizes Java for maximizing the variety of potential target platforms (e.g., Symbian, Android). For Java-incompatible systems (such as the Apple iPhone) the framework comprises a Java-based proxy (built upon the same kernel, although without a user interface) which runs on a web server and dynamically transmits data via Ajax to a web browser for display. Thus the service can also be consumed on platforms that are not natively supported, and it abstains from tedious download and installation procedures.

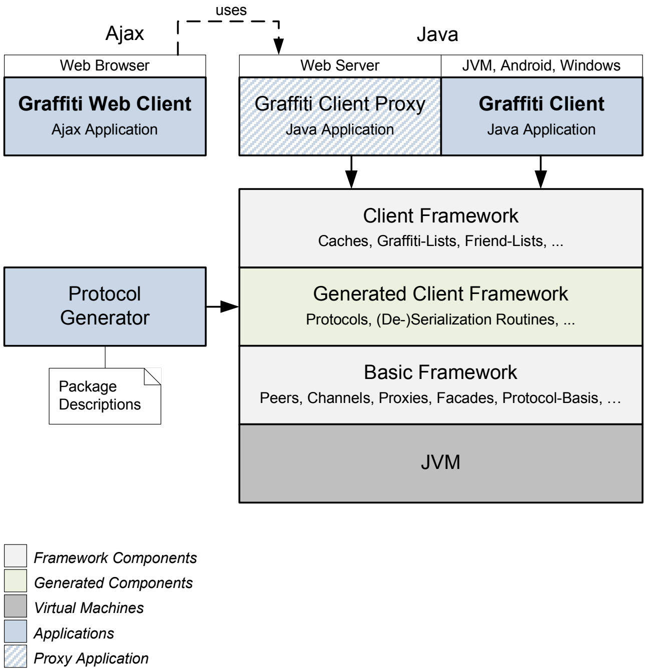


Figure 3. LBS Framework architecture

The architecture envisioned in this article employs the Digital Graffiti framework as third component. The framework acts as frontend for the tracking system, allowing any device equipped with an 802.11b/g/n interface and a web browser to consume location. Once registered and logged in, the user is visualized as an avatar at his exact residing position in front of a map and his geographical position is textually resolved into a human readable address. Digital Graffiti therefore provides a "*spatial coding*" component to map geographical addresses to corresponding names at detailed spatial levels (i.e., buildings, floors and even distinct rooms). Alternatively to indoor WLAN localization the system supports a seamless transition from and to outdoor GPS tracking as well. The position of the user is updated at a near real time frequency (due to the hop cycle length every three seconds; cf. Section 3B). Alongside user's own position, the system also offers to track the position of the user's friends, provided that the respective friend has granted permission. To sustain privacy this permission can be revoked by one click in the user interface.

IV. Application

The applicability of the infrastructural approach of WLAN tracking using the Digital Graffiti framework as a user frontend is being demonstrated within a campus-wide live system at the University of Linz, called the Smart Information Campus (SIC) Project [29]. It covers an area of about 800x300 meters campus space (cf. Fig. 4) including 15 multistory buildings varying from 1 to 11 stories equipped with the proposed sniffer technology, which in total results in 320 access points (Cisco Aironet 1250) and the same amount of co-located sniffer drones (Linksys WRT610N). Co-locating the sensor drones has been decided due to practical reasons such as the availability of power and Ethernet connections, but is still a matter of discussion [13] in terms of localization accuracy.

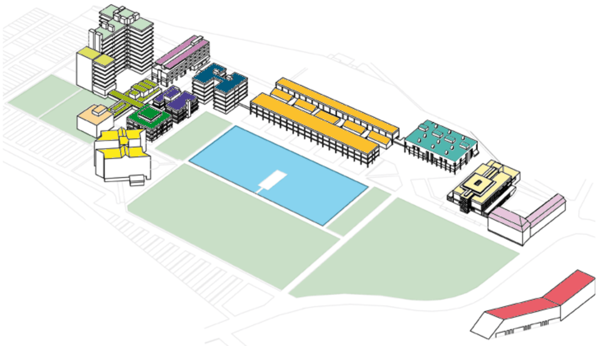


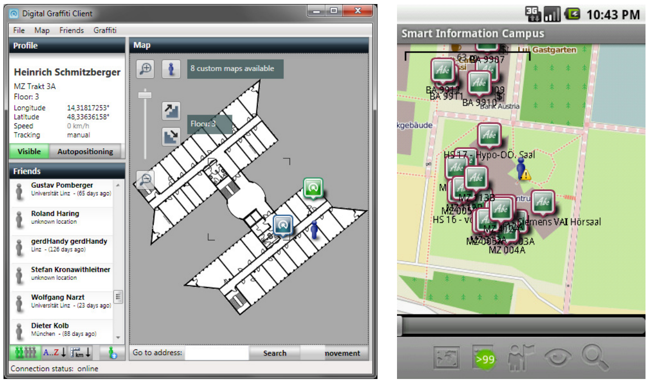
Figure 4. Overview of the building complex at the JKU campus

Table 2 gives an overview of the complete indoor areal of the campus site, showing the abbreviations of each building and a description of the respective area correlated to deployment numbers of the tracking infrastructure (amount of deployed sensor nodes, total number of calibrated candidate points, costs measured in working hours). Due to the system rollout procedure, some details are still to be determined at the time of this writing. As on every typical university campus, these buildings host office spaces, laboratories, meeting rooms and lecture halls, varying in room dimensions, furnishing and technical equipment. The campus has been built starting in the year 1964 and is continuously expanding since then. Consequently, the structural design and building characteristics reflect the ideology of the decade in which each respective building has been realized. In terms of indoor tracking, this has two important implications. For one, the amount of sniffer drones to be deployed varies correlated to wall construction substances since each substance shows its own attenuation characteristics. To assure an appropriate coverage, a traditional radio site survey gives a first guide number for planning the deployment of tracking infrastructure. The other implication for the campus-wide tracking setup is the divergence of tracking accuracy with respect to wall attenuation and the respective room layout. In Section 5 the tracking accuracy within three exemplary buildings on the campus are compared to highlight the effects of differing building design and structure on infrastructural tracking.

TABLE II. Campus Coverage Statistics

Building	Floors	Area (m^2)	Nodes	Candidate Points	Effort (h)
MZ	5	10.500	26	517	39
KE	3	19.293	50	818	47
HF	5	6.650	26	424	29
ScP	7	64.232	32	978	40
SL	3	3.255	8	172	7
UC	4	6.600	11	275	15
BI	3	6.720	12	386	22
HP	4	2.600	8	213	10
ME	1	450	3		
HT	3	2.925	3		
BA	4	3.840	10		
PH	4	3.844			
JU	5	11.970			
Turm	11	18.810			
KG	7	4.256			
		165.945	***189***	***3.783***	***209***

The Smart Information Campus system offers several means of consuming the provided location-based services on a broad variety of mobile platforms. To this end, we focus on four basic directions of implementation. We offer a native Windows desktop application for laptop and netbook service consumption based on *Microsoft WPF*. Java compatible mobile phones (such as Symbian based phones, etc.) are supported by the J2ME client implementation using the Kuix UI framework [30]. For Android based smart phones we additionally implemented a touch screen application utilizing the sophisticated features this modern platform is offering. To provide access for the remaining mobile platforms (e.g., Apple iPhone and iPad, Blackberry, Linux, etc.) we implemented a web application client based on AJAX. Fig. 5 and 6 show screenshots of all four client variants.



Digital Graffiti Client

Profile

Heinrich Schmitzberger
MZ Trakt 3A
Floor: 3
Longitude 14,31817253°
Latitude 48,33636158°
Speed 0 km/h
Tracking manual

Friends

Gustav Pomberger – Universität Linz - (65 days ago)
Roland Haring – unknown location
gerdHandy gerdHandy – Linz - (126 days ago)
Stefan Kronawithleitner – unknown location
Wolfgang Narzt – Universität Linz - (23 days ago)
Dieter Kolb – München - (88 days ago)

Go to address:

Connection status: online

Smart Information Campus

Figure 5. Client Snapshots (Desktop and Android)

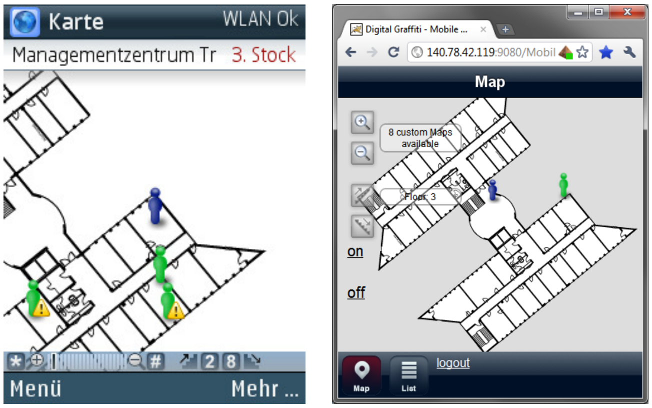


Figure 6. Client Snapshots (J2ME and web application)

The system has started its beta phase in January 2010 and is since then publically available 24 hours a day, 7 days a week [28]. Students as well as lectors, administrative staff and guests (a total number of about 16.000 people) are able to track themselves and their selected friends (Fig. 7 right presents a snapshot of the SIC application revealing our own position by a blue avatar and those of the friends by green ones) and perceive and post location-based information at the university campus due to their detected position within buildings and outside. For instance, students may find their way to their lecture halls displaying the current lecture type and times for these rooms; the event management announces upcoming activities or presentations to the users related to their location; teachers are able to ad-hoc exchange documents with students just because of their geographical attendance in a lecture room; etc. Generally, the SIC is supposed to enhance social, scientific and organizational networking within the campus, e.g., enabling the creation of communities and providing a practical research platform for location-based service issues.

As of this writing, the system is at the end of phase 2 of 3 rollout phases, providing indoor localization for 8 of 15 buildings, already covering about 72% of the whole indoor area. Due to their age these buildings have dissimilar construction characteristics that imply different radio propagation properties. To provide accurate localization in such an environment, the application of an architecture based on an *a priori* off-line training process has proven to be feasible. Consequently, a radio profile was taken at a larger number of spots in every building (cf. Table 2).

V. Results

The system setup is under constant assessment by 200 selected beta users at the university, with an average number of 33 concurrent position requests per second during a regular working day. In order to provide a satisfying LBS experience our tracking system is set up to assure the correct symbolic reference rather than the exact position. For most places on the campus this does not make a perceivable difference since the position has room level accuracy. If localized within one of the bigger lecture halls though, our approach might result in a bigger estimation error.

Fig. 7 shows a live snapshot of the J2ME client running on a Nokia E52 on an office floor. The position is indicated by the blue avatar in the center of the map. The symbolic name of the actual position appears at the top of the display, indicating the name of the building along with the floor. Since located in a corridor, no office name is displayed.

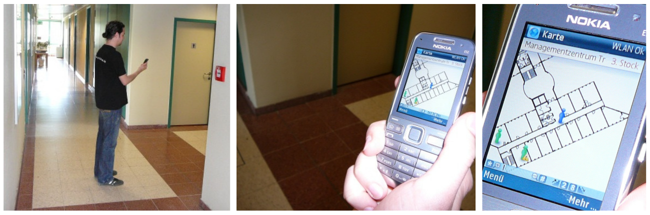


Figure 7. Tracking snapshot at office floor on J2ME client

The accuracy of the system primarily depends on the density of sniffer drones, their positions and the buildings' characteristics. Consequently, we encounter different localization precision at different sites. The following Figures 8, 9 and 10 depict exemplarily captured tracks of a Nokia E52 client walking along a predetermined path (indicated by the red line) in 3 different buildings on the campus of *Johannes Kepler University*. All position measurements (indicated by the red crosses in the figures) were taken weekdays around midday. As the data was collected during the academic semester, the offices and hall ways were averagely crowded with students and university staff. The effects of the time of day on an indoor localization system have already been studied by Tao et al. in [31]. Their studies showed that the signal strength histograms of measurements vary noticeably as a function of the time of day, with significantly more noise when more people are in the building. Consequently, our sample tracks were captured at the same time of day. However, the bias provoked by an arbitrary amount of people interfering with signal measurements is not compensated in our framework yet. Thus, we have to keep in mind that our test cases are compared under slightly divergent environmental conditions. A more detailed investigation on effects derived from building characteristics and people-depending signal strength variations has been presented in [32] along with an approach of compensating for consequential positioning errors.

Fig. 8 shows the first test track recorded on the ground floor of the *Kepler Building* (KE). It comprises a base area of 11782m^2 on 3 stories and is one of the oldest buildings on the campus, consisting of several lecture halls, libraries and public meeting places. Since its wall structure is thick and radio absorbing (concrete and brick walls) it is equipped with 50 Sniffer Drones to cover the whole building (cf. Table 2). The actual track through the building started and ended at the lower left quadrant of the figure and took 4:11 minutes for a distance of 328m, using a client transmission interval of 1 second. It comprises 80 position updates, all of which correctly located at the ground floor.

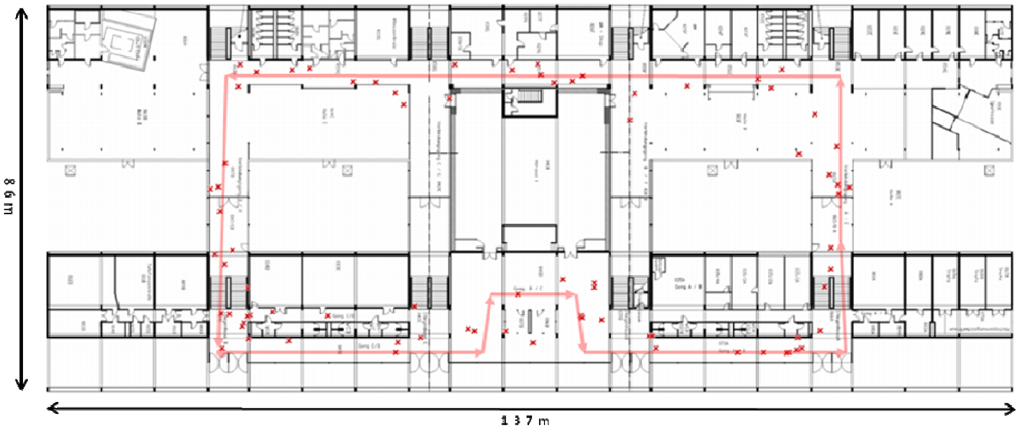


Figure 8. Track through auditorium building (KE)

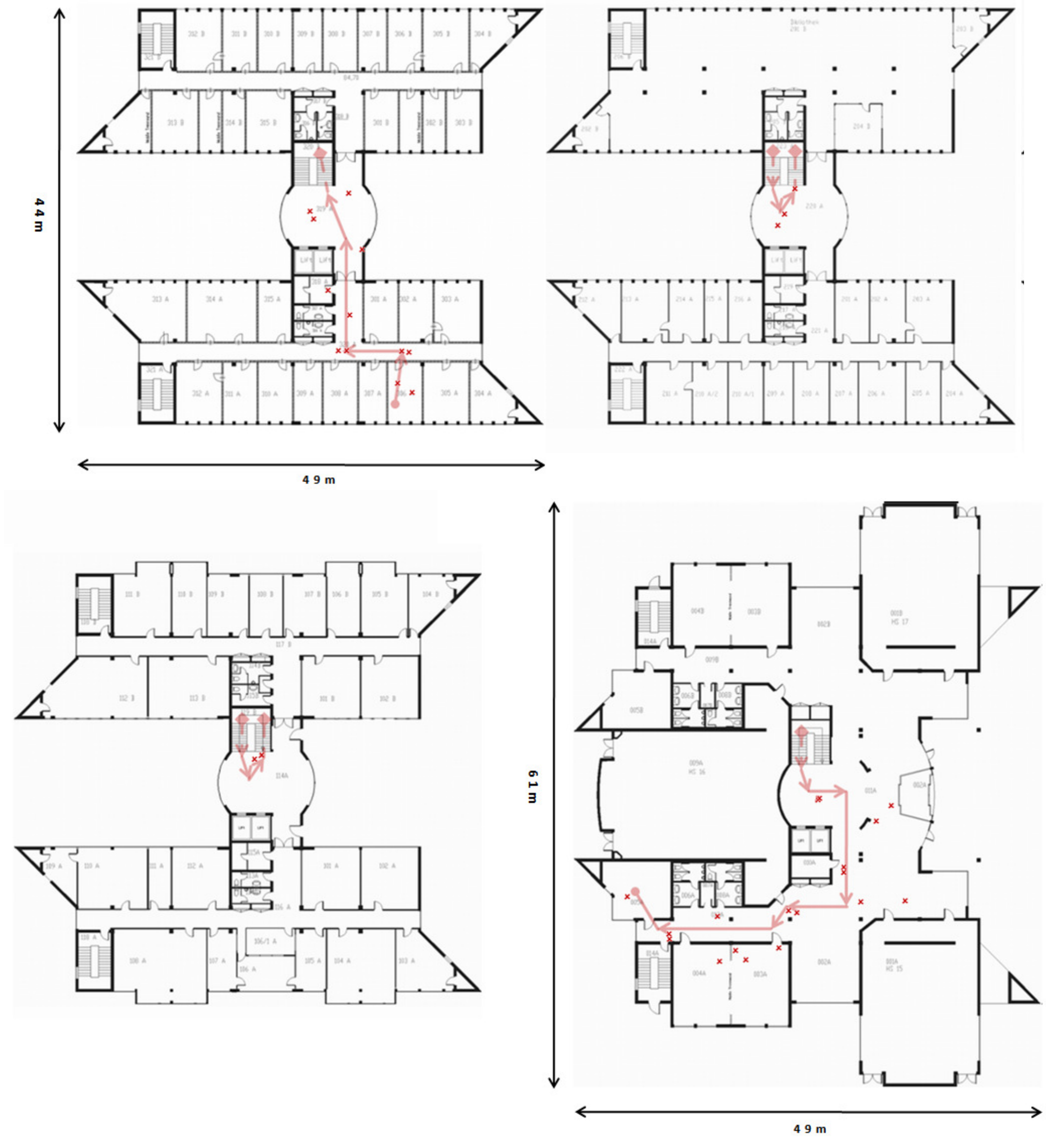


Figure 9. Track through office building (MZ)

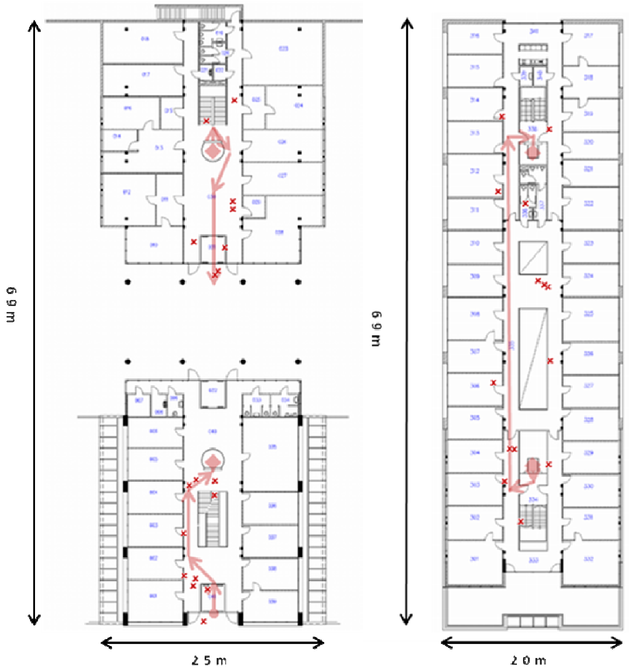


Figure 10. Track through laboratory building (HF)

In Fig. 9, the track is captured within the *Management Zentrum* (MZ), a typical office building comprising meeting rooms and smaller lecture rooms as well. It has been built in 1990 using light-frame construction. The small circle in the upper left picture marks the starting point in an office at the third floor. The client then descends to the ground floor using the stairway and passing the second and the first floor (cf. upper right and lower left picture). Each floor transition has been estimated correctly. The track finally ends in a small lecture room indicated by another red circle in the lower left quadrant of the lower right picture. It took 3:28 minutes for a distance of 126m, featuring 37 position updates at the same transmission interval as above but at a lower walking pace.

The third track (cf. Fig. 10) was measured inside the *Hochschulfond Building* (HF), which has been built in the year 2003. Because it is mainly dedicated to practical research, it houses technical laboratories, offices and storage spaces. The constructive form of the building can be characterized as modern architecture, emphasizing glass as building substance and a large galleria section spanning from the first to the third floor. The track started at the south entrance (at the bottom of the left picture) on the ground floor. The diamond marks the entering of an elevator that ascended to the third floor (right picture). The track led alongside the galleria towards the north elevator. After descending again, the track ended on the ground floor (in the middle of the picture on the left). Since the elevator is surrounded by a glass construction, the tracking engine was able to estimate each floor and position correctly even within the elevator. The track was finished after 3:56 minutes, even though the covered distance was only 96m including 33 position updates. The long duration was mainly caused by elevator waiting times. Transmission interval and walking pace correspond to track two.

TABLE III. ERROR DISTANCE STATISTICS

	KE	MZ	HF
Average	4.11 m	2.40 m	2.90 m
Std. deviation	2.30 m	1.38 m	1.69 m
25th percentile	2.33 m	1.30 m	1.75 m
Median	4.08 m	2.15 m	2.47 m
75th percentile	5.21 m	3.38 m	3.53 m
90th percentile	7.59 m	4.14 m	5.07 m
Maximum error	10.09 m	6.47 m	7.13 m

Detailed results of these three test tracks based on the error distance are summarized in Table 3. Fig. 11 depicts a comparison chart that clearly highlights the differences in accuracy referring to the respective building. Track one as well as Track three both showed two phases without a position update indicating that the new estimate would have been an outlier. For the first track, this can be explained by a sudden appearance of a larger group of people since these positions were near a lecture hall (Fig. 8, upper right) and a cafeteria (Fig. 8, upper left). In the third track, this phenomenon is caused by the open space from the galleria (Fig. 10, in the middle of the right picture). Due to the gap in the middle section ranging from the first up to the third floor, the signal strength fingerprints tend to be very similar on both sides of the galleria.

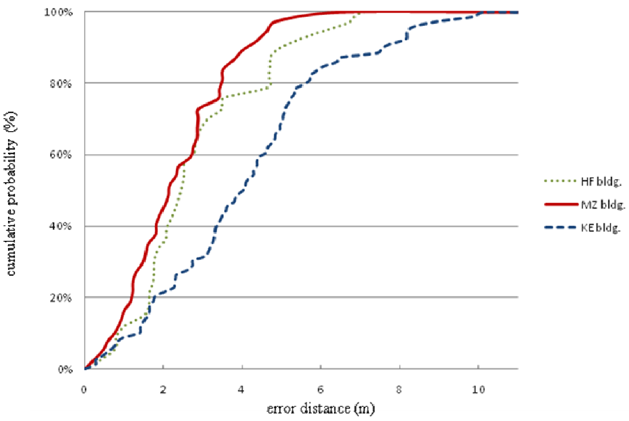


Figure 11. Comparison of the tracking accuracy in exemplary buildings

Concluding from these results, the tracking framework presented in this article shows a clear tendency towards better performance and accuracy within younger buildings. As reported in the context of client-based indoor localization setups [32], this can be explained with building characteristics such as radio absorbing wall substances or large spaces that provoke indistinguishable fingerprints. Out of all types of campus facilities, office floors turn out to provide the most promising environment for indoor tracking because of their room arrangement diversity that results in unambiguous fingerprints.

In terms of overall accuracy, the presented system still leaves room for improvement (compensating mechanisms for dynamic attenuation provoked by people for instance). Whereas newer contributions report mean estimation errors under 2m [18][22], our setup reveals values from 2.4m up to 4.11m (depending on the building properties). However, comparable tracking systems are commonly investigated under laboratory conditions (i.e., small areas of 500-1000m^2, tailored radio coverage, homogeneous building structures, single-story localization, etc.), not considering real-life scenarios in a continuous operation mode, as well as economic factors of comprehensive infrastructure deployment.

VI. Conclusion and Future Work

In this article, we presented a system capable of concurrently tracking numerous clients with no special client-side hardware prerequisites within a large-scale indoor setup. The proposed architecture has been deployed in a real-life setup at the University of Linz and is part of a project providing campus-wide LBS to an academic audience, outdoor as well as indoor. As the campus comprises diverse buildings of manifold types of architecture, the tracking system has to cope with a variety of different radio distribution characteristics. By comparing the performance of our system in three exemplary test cases that reflect the three most diverging radio environments on the campus site, we pointed out the feasibility of our architecture. The results clearly indicate that system accuracy benefits from the more modern style of building construction on the one hand, as well from an office floor layout. Overall, the system performance can compete with other related systems, even under real-life conditions and on a campus-wide scale.

We reported detailed deployment numbers and costs (in working hours) that were invested in our large-scale setup. In this regard, a clear disadvantage still exists in the form of the tedious training process (approximately 40 working hours for the full sensor coverage of an average building) that precedes the life system. Therefore, we are exploring alternatives to our two-phased fingerprinting architecture. We envision a benefit from making use of beam-forming technology that is one of the most promising features of the new 802.11n (still draft) standard for WLAN localization. Unfortunately, the driver used by the sniffer drones does not support obtaining lower level antenna reception information yet. Further experiments need to be conducted relating to the 802.11a standard in order to cope with the greater amount of alternative radio channels, that common 5GHz WLAN networks make use of (at JK University we're using 12) and that imply a greater hop cycle length in our current system. Since some WLAN NIC drivers capable of both 2,4GHz and 5GHz communication tend to favor the 5GHz band, the respective NIC drivers have to be configured explicitly to use 2,4GHz first. In the future we hope to avoid this user inconvenience.

The compensation of signal strength fluctuation effects provoked by people passing by is another important issue we are investigating at the time of this writing. Our focus lies on achieving a solution for our network consisting of 320 Sniffer Drones that solely runs decentralized, i.e., exclusively on the sensor platform to avoid a potential bottleneck at the backend. To this end, a mechanism to reliably detect spontaneously emerging crowds of people has to be realized that relies on pure radio environment fluctuations and does not depend on emitters carried by the people within the crowd [22]. This way, an even more accurate and temporarily stable tracking experience could be provided.

References

[1] H. Schmitzberger and W. Narzt, "Leveraging wlan infrastructure for large-scale indoor tracking," in *Wireless and Mobile Communications (ICWMC), 2010 6th International Conference on*, 2010, pp. 250 – 255.

[2] M. Hazas, J. Scott, and J. Krumm, "Location-aware computing comes of age," *Computer*, vol. 37, no. 2, pp. 95–97, 2004.

[3] "Skyhook wireless." [Online]. Available: http://www.skyhookwireless.com/
[last accessed on June, 28. 2011].

[4] "Ekahau rtls." [Online]. Available: http://www.ekahau.com/-products/real-time-location-system/overview.html
[last accessed on June, 28. 2011].

[5] "Aeroscout." [Online]. Available: http://www.aeroscout.com/
[last accessed on June, 28. 2011].

[6] T. Sohn, W. G. Griswold, J. Scott, A. LaMarca, Y. Chawathe, I. Smith, and M. Chen, "Experiences with place lab: an open source toolkit for location-aware computing," in *ICSE '06: Proceedings of the 28th international conference on Software engineering*. New York, NY, USA: ACM, 2006, pp. 462–471.

[7] J. Wierenga and P. Komisarczuk, "Simple: developing a lbs positioning solution," in *MUM '05: Proceedings of the 4th international conference on Mobile and ubiquitous multimedia*. New York, NY, USA: ACM, 2005, pp. 48–55.

[8] M. Youssef and A. Agrawala, "The horus wlan location determination system," in *MobiSys '05: Proceedings of the 3rd international conference on Mobile systems, applications, and services*. New York, NY, USA: ACM, 2005, pp. 205–218.

[9] P. Krishnan, A. S. Krishnakumar, W.-H. Ju, C. Mallows, and S. Ganu, "A system for lease: Location estimation assisted by stationery emitters for indoor rf wireless networks," in *Proceedings of IEEE Infocom 2004*, 2004.

[10] M. B. Kj, "A taxonomy for radio location fingerprinting," in *Proceedings of the Third International Symposium on Location and Context Awareness*, 2007.

[11] J. Krumm, Ed., *Ubiquitous Computing Fundamentals*. CRC Press, 2010.

[12] P. Bahl and V. N. Padmanabhan, "Radar: An in-building rf-based user location and tracking system," in *INFOCOM*, 2000, pp. 775–784.

[13] S. Ganu, A. S. Krishnakumar, and P. Krishnan, "Infrastructure-based location estimation in wlan networks," in *Proceedings of the IEEE WCNC 2004*, 2004.

[14] M. Youssef, A. Youssef, C. Rieger, U. Shankar, and A. Agrawala, "Pinpoint: An asynchronous time-based location determination system," in *MobiSys '06: Proceedings of the 4th international conference on Mobile systems, applications and services*. New York, NY, USA: ACM, 2006, pp. 165–176.

[15] J. H. H. Lim, L. Kung and H. Luo, "Zero-configuration, robust indoor localization: Theory and experimentation," in *INFOCOM*, 2006.

[16] L. F. M. de Moraes and B. A. A. Nunes, "Calibration-free wlan location system based on dynamic mapping of signal strength," in

MobiWac '06: Proceedings of the 4th ACM international workshop on Mobility management and wireless access. New York, NY, USA: ACM, 2006, pp. 92–99.

[17] "Cisco wireless location appliance." [Online]. Available: http://www.cisco.com/en/US/products/ps6386/
[last accessed on June, 28. 2011].

[18] J. Indulska and P. Sutton, "Location management in pervasive systems," in *ACSW Frontiers '03: Proceedings of the Australasian information security workshop conference on ACSW frontiers 2003*. Darlinghurst, Australia, Australia: Australian Computer Society, Inc., 2003, pp. 143–151.

[19] J. Yim, S. Jeong, K. Gwon, and J. Joo, "Improvement of kalman filters for wlan based indoor tracking," *Expert Systems with Applications*, vol. In Press, Corrected Proof, pp. –, 2009.

[20] n.-B. Galo Nu and J. M. Páz-Borrallo, "A new location estimation system for wireless networks based on linear discriminant functions and hidden markov models," *EURASIP J. Appl. Signal Process.*, vol. 2006, no. 1, pp. 159–159, 2006.

[21] N. Swangmuang and P. Krishnamurthy, "Location fingerprint analyses toward efficient indoor positioning," *Pervasive Computing and Communications, IEEE International Conference on*, vol. 0, pp. 100–109, 2008.

[22] M. Youssef, M. Mah, and A. Agrawala, "Challenges: device-free passive localization for wireless environments," in *MobiCom '07: Proceedings of the 13th annual ACM international conference on Mobile computing and networking*. New York, NY, USA: ACM, 2007, pp. 222–229.

[23] K. Vandikas, A. Katranidou, L. Kriara, H. Baltzakis, T. Papakonstantinou, and M. Papadopouli, "Empirical-based analysis of a cooperative location-sensing system," in *Autonomics '07: Proceedings of the 1st international conference on Autonomic computing and communication systems*. ICST, Brussels, Belgium, Belgium: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2007, pp. 1–11.

[24] "Dd-wrt." [Online]. Available: http://www.dd-wrt.com/
[last accessed on June, 28. 2011].

[25] "Libpcap." [Online]. Available: http://www.tcpdump.org/
[last accessed on June, 28. 2011].

[26] G. E. Violettas, T. L. Theodorou, and C. K. Georgiadis, "Netargus: An snmp monitor & wi-fi positioning, 3-tier application suite," in *ICWMC '09: Proceedings of the 2009 Fifth International Conference on Wireless and Mobile Communications*. Washington, DC, USA: IEEE Computer Society, 2009, pp. 346–351.

[27] "Ieee standard for a precision clock synchronization protocol for networked measurement and control systems," *IEC 61588:2009(E)*, pp. C1 –274, 2 2009.

[28] W. Narzt, G. Pomberger, A. Ferscha, D. Kolb, R. Müller, H. Hörtner, and R. Haring, "Addressing concepts for mobile location-based information services," in *UAHCI'07: Proceedings of the 4th international conference on Universal access in human-computer interaction*. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 507–516.

[29] "The smart information campus," 2011. [Online]. Available: http://sic.jku.at/
[last accessed on June, 28. 2011].

[30] "The kuix user interface framework," 2011. [Online]. Available: http://www.kalmeo.org/projects/kuix
[last accessed on June, 28. 2011].

[31] P. Tao, A. Rudys, A. M. Ladd, and D. S. Wallach, "Wireless lan location-sensing for security applications," in *Proceedings of the 2nd ACM workshop on Wireless security*, ser. WiSe '03. New York, NY, USA: ACM, 2003, pp. 11–20.

[32] K. Kaemarungsi, "Design of indoor positioning systems based on location fingerprinting technique," Ph.D. dissertation, University of Pittsburgh, 2005.

Sensor Data Fusion Middleware for Cooperative Traffic Applications

Teemu Leppänen [1], Mikko Perttunen [1], Pekka Kaipio [2], Jukka Riekki [1]

[1] Department of Electrical and Information Engineering
University of Oulu
Oulu, Finland
{teemu.leppanen, mikko.perttunen,
jukka.riekki}@ee.oulu.fi

[2] Department of Information Processing Science
University of Oulu
Oulu, Finland
pekka.kaipio@oulu.fi

Abstract – **Recent advancements in information and communication technology accelerate the development and deployment of intelligent transportation systems. Cooperative traffic systems utilize information collected by road users and traffic operators about their immediate environment sharing it system-wide, enabling services for all traffic entities. In this work, we contribute a data processing and data fusion middleware relying on wireless mobile sensor nodes and public infrastructure services for traffic data collection for future cooperative traffic applications and services. The system is demonstrated with an example application, utilizing mobile phones with integrated sensors as sensor nodes and producing the visualization of travelled routes annotated with travel mode and detected anomalies of road surface. Middleware features include modular component-based architecture, reconfigurable and reusable data processing components, dynamic addition of components and interfaces into the system in runtime and data fusion of heterogeneous sensor data through chaining of components into application specific data fusion levels.**

Keywords - Cooperative systems; sensor networks; wireless sensing; information fusion.

I. Introduction

Increasing bandwidth and 3G technologies available for wireless communication enable the advancement of intelligent transportation systems towards cooperative traffic systems, where traffic entities collect and share data and become more and more context-aware [1]. In [2], cooperative traffic is given the following description: "Road operators, infrastructure, vehicles, their drivers and other road users will co-operate to deliver the most efficient, safe, secure and comfortable journeys. The vehicle-to-vehicle and vehicle-to-infrastructure cooperative systems will contribute to these objectives beyond the improvements achievable with stand-alone systems." These actors in traffic can be provided with information, for example, about traffic disorders, road conditions, current weather conditions and usage statistics as well as routes based on dynamic traffic information [3].

The increased need for accurate traffic information has led to the integration of offline traffic information, such as fixed road infrastructure sensor data, with real-time information from supplementary data sources, such as cameras, GPS and cell phone tracking [4]. In addition of providing complementary data, these data sources assist in decreasing uncertainly of the data from individual sources and enhance the decision maker's performance [4][5]. This will result in many traffic engineering problems to become data fusion problems, producing estimates or an improved model of the system being observed [4]. Different techniques for data fusion exist: statistical approaches, probabilistic approaches and artificial intelligence [4]. Sensor data fusion techniques can be characterized in several domains [5]. In the application domain, only application relevant concepts are considered. In the fusion objective domain, the objectives of the fusion are considered, for example object recognition. The selected sensor types dictate compatibility and complementary nature of different sensors. The sensor configuration, as in concurrent or temporally separated measurements, defines the sensor usage. Finally, the fusion process is usually described as a three-level hierarchy: data fusion, feature fusion and decision fusion. In any of the three levels spatial or temporal fusion may occur [5]. This model with the hierarchical levels with a varying level of detail in information has been widely accepted [4][5].

Participatory sensing is a method for collecting data from many unknown and independent contributors in collaboration [6]. Cooperation between contributors can take several forms [7]. Informative cooperation occurs when users extend the system range by transmitting their sensory data in one-way communication. In descriptive cooperation, sensory data is augmented with intentions such as the intended direction of movement of the user. Coordinative communication allows objects to reason about and modify their behavior depending on other's intentions, which requires two-way communication. The key question here is: What information should be shared and in which level of processing and fusion? [7] Performing data processing and fusion refines the data and reduces the amount of data transferred in the system. However, it also reduces the sensor network coverage, reliability and accuracy of the data as fewer samples are available. Communication technology issues, such as wireless channel capacity, also limit coverage and reliability.

Wireless (mobile) sensor networks offer several advantages over fixed sensor networks [8]. First, the coverage of a network can be extended easily with smaller costs. Second, nodes can store data locally and provide data only when requested, which contributes towards scalability and energy saving. Third, if a fixed sensor node fails or experiences network failures, mobile data collectors can be used to compensate the data loss. Fourth, localized algorithms in nodes can be utilized in data collection and processing tasks. Khanafer et al. [9] define two main categories for wireless sensor networks for intelligent transportation systems: planar and multi-tiered. Planar architectures utilize mobile sensors and can have infrastructure-less communication paradigm supporting vehicle-to-vehicle networks, forming mobile ad hoc wireless sensor networks. Another paradigm for planar architectures is infrastructure-based supporting vehicle-to-infrastructure and infrastructure-to-vehicle communication by utilizing fixed roadside units. Roadside units relay data to services and can also form clusters of members around them. Some challenges for planar architectures exist, related to network topology and scalability. Furthermore, there are known constrains such as processing capabilities of the nodes, which affects to quality of service among other things. Networks in the second category, multi-tiered architectures, can have better performance by supporting heterogeneous communication, such as wireless local area networks providing more available bandwidth. This means having better quality of service when computations can be relayed to more powerful units in upper layers utilizing the increased bandwidth available. Of course, enabling more communication technologies adds complexity and costs to the system.

As mobile phones are today widely used, equipped with sensors and interfaces to external sensors and they have processor, memory, battery and communication units, they can be also seen as sensor nodes [10]. Mobile data collectors, however, have communication issues when transferring data to the sinks as bandwidth is limited [8]. Also, battery life, processing power and memory capacity are limited. Normal phone usage will limit the data collection capability [10]. Data routing is influenced by the network topology, the quality of the service parameters and contents of the data [8].

The Finnish Cooperative traffic program [11] envisions sustainable traffic using extensive information sharing based on novel technologies and services. In the Sensor Data Fusion and Applications project as a part of this program, our focus is on utilizing new sensor data sources and data fusion methods for generating new potential applications [11]. Our vision in [1] includes the usage of mobile phones with integrated sensors, giving information on the behavior of the actors in traffic. Instead of focusing on tailored applications on the field, *our goal is to develop a flexible system that can be used to demonstrate and analyze a variety of cooperative traffic scenarios and applications working cooperatively with little development overhead.* To enable this we derive a wide set of requirements from the literature and our example scenarios and take a bottom up approach, contributing a practical data collection system with the ability to integrate heterogeneous sensor data and infrastructure services as data sources, data fusion capabilities and pluggable data processing components. The demonstration system includes a visualization client application that produces annotated maps and enables end-to-end demonstrations. In this paper, we report the system prototype and an example application featuring real-time travel mode and road surface anomaly detection, with mobile phones with integrated sensor as wireless sensor nodes.

The rest of this paper is organized as follows: Section II reviews the related work in the area. In Section III, we define requirements and scenarios for a cooperative traffic sensor network and describe our system. In Section IV, we compare the developed system against the requirements and represent field test results. Finally, in Section V, thoughts for future work are given.

II. Related work

We describe here a number of existing applications utilizing mobile traffic-related sensor data and existing solutions for data processing and fusion for intelligent transportation systems.

A. *Traffic sensing applications*

Several participatory sensing systems have been reported. In the Mobile Millennium project [12], GPS enabled mobile phones are used for the collection of traffic data, which is then fused with data from sensors in the road infrastructure. The system is employed for monitoring and estimating the traffic flow in real-time. The produced estimates are then transferred back to the phones. Google Maps for Mobile [13] offers live traffic conditions monitoring by retrieving the speed of vehicles from the GPS data from the mobile phones. TJam [14] uses GPS receivers to predict traffic jams by measuring the velocity of the vehicles. Users' coordinates are transmitted to the service and the probabilities of congestion in a given region are sent back. In Nericell [10], mobile phones with accelerometer sensors are used in vehicles to detect road bumps, among other features. The detection software is installed in the mobile phone itself and they introduce the concept of triggered sensing where less energy consuming sensors are used to trigger the usage of other sensors.

B. *Data fusion platforms*

In the following we describe several platforms developed for processing traffic-related information from sensor networks. The hierarchical Content Delivery Network architecture described by Elshenawy and others [15] has network hierarchies to provide scaling. Vehicles contain on-board units acting as clients in the network. Road-side units work as a surrogate servers having storage for the content and self-healing mechanism for communication failures in the network. In the next hierarchy level, geographical areas are grouped into geographical domains, which are controlled by domain managers. It is the manager’s responsibility to dynamically route content to the vehicles under their domain, which should also decrease delivery time. Domain managers can be grouped together recursively. Services sent add and delete messages, which propagate through the network nodes, based on the geographical area the service covers. On the other hand, on-board units can send discovery messages to locate available services in a geographical area.

In the Cooperative Vehicle Infrastructure System [16][17] project, OSGi open source platform has been selected as the application platform. Communication is done using the CALM standard, providing both wireless local area and cellular network access, which can be selected dynamically by having better quality of service. The system consists of three layers, in the upper layer is the central management working on the system-wide level. The middle layer represents the roadside infrastructure at the regional level and the lowest level represents the individual vehicles.

Tacconi et al. [18] describe an information retrieval scenario, where mobile sinks, for example vehicles, query data from edge nodes in the wireless sensor network. Sensor nodes and sink nodes are aware of their geographical position or spatial distribution. Data routing from the nodes to mobile sinks is done by predicting the current position of the mobile sink based on its mobility information within the network from the original location of the injected query.

The Telematic Management System [19] component suite has many similarities with our approach. It has three main components: the kernel, communication subsystem and the data module. Framework users provide a set of decision modules accessing the communication system and data module. The decision modules and their dependencies on other modules are defined as graphs in XML documents. With the decision modules, users define a protocol component, which propagates incoming messages to interested modules through the graph. The kernel provides interfaces for the decision modules and initiates them. Data modules define a mechanism to access the data concurrently. The communication subsystem uses TCP/IP asynchronously. For each vehicle there is a local queue in the subsystem, which is used to store messages directed to it. Messages from the vehicles are stored in the general system queue.

Lv et al. [21] define ubiquitous intelligent transportation system architecture as vehicle-to-vehicle or vehicle-to-infrastructure communication with access to large-scale information systems. Their system utilizes multi-source real-time location-based mobile sensor data and fixed sensor data via context-awareness technology. The key idea is the interaction between any devices in any location or time. In their system there are five layers. The resource management layer collects data from the mobile sensors and sensors in the roadside, for example weather stations. Preprocessing of the data and classification is also done in this layer. The information awareness layer fuses the data requested by the services running in the system. Next, the intelligent service layer provides access to data and services from other service providers. These services may be accessed via websites or mobile phones for example. The terminal application layer provides the dynamic system level services such as real-time traffic information or road guidance to personal computers and other terminals. Finally, the ubiquitous network layer provides hybrid network communication models to support usage of multiple communication technologies.

III. System and middleware

We identify various cooperative traffic scenarios to sketch the requirements for the data processing system. In the scenarios, the target users include a car driver, a pedestrian and a traffic operator.

One scenario sketches services supporting efficient responses to traffic accidents. If an intelligent vehicle notices changes in road conditions, or for example detects a traffic accident from abrupt braking, the vehicle can send this information to be added to the real-time situation model of the road segment. The new information can then be disseminated to other traffic entities in the system. The traffic accident location can be inserted to a database. Moreover, a real-time warning can be issued to the drivers planning to drive in that road by updating the route prediction data in their vehicle navigators. A weather service integrated into the system can be used to determine the current local weather and for example lighting conditions when the accident happened. This is useful information at least during the Nordic winter. This information aggregated with the road condition data and on-board diagnostics data from the vehicle can be stored as accident information to the database, where it is available for further analysis by road authorities. In the future, this allows drivers to be warned to avoid accidents when similar weather and road conditions appear. In addition, mobile phone data or roadside sensors, such as cameras, can be used to detect approaching dangerously speeding vehicles. Information about speeding vehicles, aggregated with the real-time road situation information, can produce an alert issued to other entities in that road segment and to the authorities.

Another scenario is warning the driver of a vehicle when approaching a school area where children are walking or bicycling nearby, which can be detected from children's mobile phone location or roadside sensor data and even aggregated with current time. Additionally, static data for a road segment for improving situation awareness can be requested from a service in the Internet. In Finland, this kind of static road data is available from a public Internet service called DigiRoad [20]. The data includes: road name, entry and exit points, direction, speed limits, number of lanes, traffic signs, location of pedestrian crossings, bus and taxi stops, elements such as illumination, service elements such as car parks and gas stations, and finally even information about scenic locations for tourists.

All this information combined together provides local real-time situation of the traffic in that road segment. Mobile phones with integrated GPS receivers in vehicles without any sensors can provide real-time information to the system, such as the location, speed and direction of the vehicle. This helps detecting the real-time congestion in road segments, which can be used in the system to guide drivers through alternative routes. Eventually the authorities are provided with a real-time situation of traffic in the whole system level. This can be used for real-time traffic monitoring, planning road infrastructure development and in a smaller scale, for planning future heavy transport scenarios.

Although we identified several scenarios, we do not target the data processing middleware for specific sensors or applications. Thus, services offered by the middleware cannot be tailored and optimized as suggested in [22]. Furthermore, isolated data sources and heterogeneous communication networks set challenges for the systems [22].

A. General Requirements

Ducourthial [23] describes an architecture for vehicular networks and lists unique issues not currently experienced within other networks. These networks are highly dynamic, unreliable, asynchronous, penalized with low bandwidth availability and short communication duration. In spite of this, vehicular networks require robustness, high quality of service and real time operation. Khanafer et al. [9] suggest a number of general requirements for intelligent traffic systems. Fault tolerance and real-time communications are essential to maintain the quality of service. As node deployment is not fixed and network topology can change, a scalable communication network is required. Cost and power consumption of a single sensor node should be minimized. System security should be guaranteed as the data is important real-time data, such as real-time traffic situation.

Furthermore, we adapt several general requirements for sensor networks from [1][8][10][24] to the scope of cooperative traffic data processing middleware: 1) the architecture should be data-centric as we will handle large amounts of heterogeneous sensor data, 2) asynchronous communication should be used because of communication outages, 3) component-based architecture and modular components follow from the diversity of the applications, 4) data processing middleware should have means for self-configuration to achieve runtime scalability with additional modules and to react to dynamic changes in network, 5) means for self-maintenance are required in the case of sensor node failures, 6) support for unknown types of future sensors is required, 7) data processing components should be easily deployable to the system and 8) the system should be able to reconfigure functionality and launch components based on data requests from the client applications. Furthermore, from the project goals we derive the following requirements: 9) component chaining is required to implement a modular system with data fusion capabilities, 10) output interfaces should be well defined and extendable to enable the usage of the resulting information in a wide variety of client applications utilizing common ontology and 11) real-time performance is a major issue when considering the usage of information in traffic.

The middleware running in the system should facilitate rapid application development and deployment as the system is developed constantly [25]. The service access points are provided by the middleware. It should also have means to manage system policies and security and privacy functions. To have a consistent quality of service in applications, the same data should be available to all applications. Thus standardized access to data and storage services need to be provided in the middleware.

B. System Description

The purpose of this prototype system is to demonstrate the chosen applications and test the feasibility of the system. Secondarily, the system functions also as a sensor network test bench for data processing algorithms under development. Currently, we have three main components in the prototype system: 1) mobile phones as mobile sensor nodes with the data collection software and integrated sensors, 2) the middleware for data processing in a remote platform in a public network and 3) a client application showing a map in a web browser on end-user workstation. The system prototype features two-way communications as users can utilize sensory data from other users to assist in decision-making, thus, we have both informative and coordinative cooperation [7].

The developed middleware architecture is data-centric and using a centralized database, as we want all the data to be rapidly propagated for the needs of any client application or data processing component. Our aim is to collect as much as possible raw data in our system. Specifically, we do not want to limit the future use of the collected data sets by applying application specific preprocessing as the level of detail is highest in the lowest level and preserving raw data enables the largest set of applications. We do not consider the level of detail of information and data shared in the system in this work. The data processing functionality can indirectly reduce the reliability and accuracy of the data as the potential loss of details occurs in all processing stages. These are important considerations for the data fusion as well [5]. We do not limit the data acquisition methods available, thus streaming, polling and event-based acquisition all can be used (req. 6). Sensor data producers are decoupled from consumers, because the commonly available sensor nodes in the system, such as mobile phones, should not be concerned about the data processing needs of the client applications. The remote data processing platform provides global endpoints for sensor data producers and infrastructure services to enter data into the system and applications to retrieve it. Global endpoints also allow heterogeneous communication methods, regardless of the physical location of the sensor nodes or the applications (req. 2). Based on these features, when considering architecture categories described by [5], our middleware architecture falls to the multi-tiered category of wireless sensor networks. Multi-tiered architectures avoid the problems with sensor network routing, maintain the quality of service better and also support data fusion in higher levels of processing, and when processing requires more powerful computation platforms [6].

We selected Global Sensor Network (GSN) [26], a sensor network middleware, as a middleware platform for our system [1]. As we are looking for complete open source implementation, for example partially released CarTel [27] cannot be considered. Lightweight Java implementation, possibility to use only a subset of modules and minimal configuration needs make GSN deployable to a large set of system configurations. Simple API and the number of already implemented features reduce the amount of required

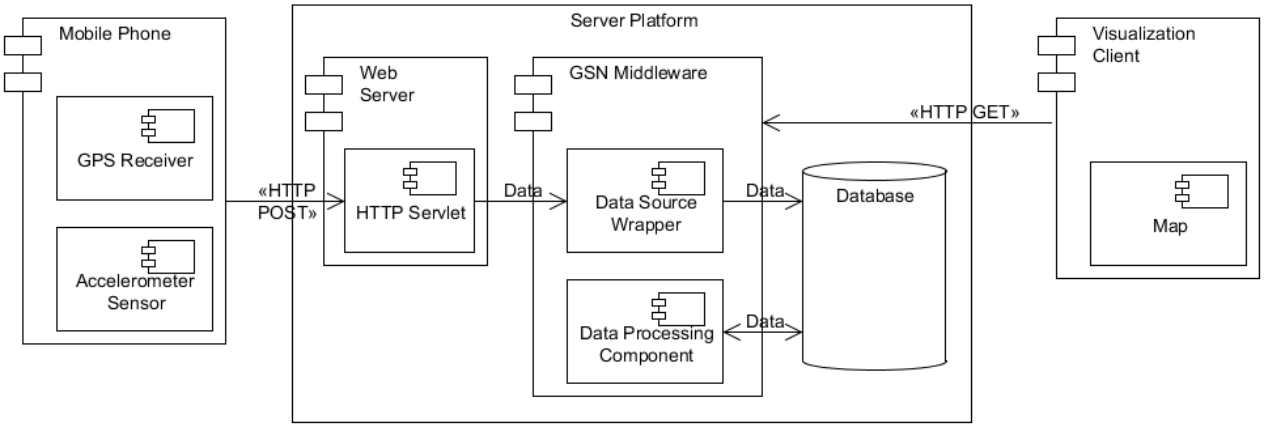


Figure 1. System architecture.

implementation work. The dynamic deployment of sensor nodes within the GSN is handled just by adding component configuration files to the system [26] (req. 7). The basic configuration defines the input/output data streams for a component. Component chaining and data fusion are enabled through defining the data input and output streams for the components (req. 9). The runtime reconfiguration of system functionality can be done by modifying the configuration file, and then GSN will dynamically start the required Java objects in the system (reqs. 4 and 8). The objects are alive only as long as required, and there is a built-in fault tolerance system for components (req. 5). Also, the dynamic use of components will not interfere with on-going data processing as the data streams are shared [26]. Addition of new data types in the system introduces the data schema evolution problem in the current components. This can be solved in GSN, as it offers means for filtering out unwanted data items for the input data stream of a component (req. 6). In case several components request the same data items, overlapping data queries are internally handled in the GSN middleware. The output data in all phases of processing can be saved to a database (req. 1).

C. Middleware components

System architecture is shown in Figure 1. Mobile phones with integrated sensors, serving as sensor nodes, collect data and disseminate it further for data processing to the server platform. In the server platform, we utilize HTTP servlet in a web server and as a sink node receiving data from the sensor nodes as HTTP POST requests. This servlet is responsible for delivering data elements to corresponding components (data source wrappers) in the middleware. For each session, the corresponding data stream is recognized from the session ID and by session we mean all the data produced by individual sensor node. The raw unprocessed data is also stored to a database for future use. In GSN, the data are streamed through MySQL database views from component to component at all steps of processing, allowing even intermediate processing results to be utilized immediately in simultaneously running algorithms and rapidly propagated to client applications (reqs. 1 and 3).

For the required application-specific data processing and data fusion, we have developed template components and component configurations for the use of data processing algorithms. The thread hosting these components will be alive as long as the session is alive and receive all the data in the session. In addition to data processing, these components can also act as interfaces to external, infrastructure or other, services in the network, for example to a public real-time traffic flow service or real-time local weather data. Template components subscribe to their required data source streams and publish single data stream as a result. These components can be deployed to the system any time by introducing new configuration files, thus automatically launching new data processing components. Another way of dynamically starting components in the system is by using the GSN's built-in web interface methods for requesting data from components. This fulfills the requirements 3, 4, 6, 7 and 8. This web interface can also be used to retrieve historical data from the database. The data processing algorithms and their parameters are defined in the template configuration files and can currently be: Java objects, binary executables or external services in the network. The subscribed data source streams are also defined in the configuration file.

For the data fusion of heterogeneous sensor data, we implemented a system to stream each type of sensor data in its own stream. All data stream elements include session ID and timestamp. This also solves the data schema evolution problem, at least partially, since all data and sources are separated from each other and required item types can be filtered from the stream. Applications can subscribe freely multiple types of sensor data for any fusion algorithms without additional overhead. Components can mix data from the different levels of data fusion and a give feedback to the system from higher levels, providing even learning capability to the system [5]. Additionally, data from external services in the network, location as GPS coordinates and user defined filters provided by GSN's built-in component interface can be utilized in data fusion. This way we can limit the data

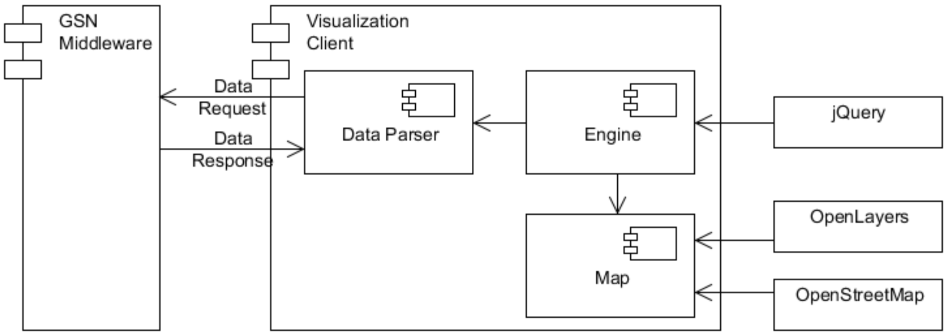


Figure 2. Visualization application architecture.

processing, data fusion or data queries, for example to certain geographical area, to specific time or to certain types and values of collected sensor data. Thus our system introduces freely configurable sensor data fusion for data processing components at every fusion level. This applies to raw and refined as well as to data from external services.

To visualize the data processing results from the middleware, we developed a web application displaying data on a map. In addition to GPS navigation and track logging systems, we have the ability to query and display properties of multiple clients dynamically with travel path history. The map shows the current location and the path history of the sensor nodes and their travel mode. Also detected anomalies on the road are displayed with a warning sign. The application polls the middleware for requested data, using GSN's built-in web interface over HTTP. Data is returned in XML based documents in the simplified format of GPS Exchange Format (GPX). TABLE 1 presents an example of this format. The first waypoint, described by element <wpt>, gives the current location and the element <rte> describes the path history as a series of waypoints. The visualization application architecture is presented in Figure 2 and Figure 3 shows example outputs of the visualization application on a map.

The application runs completely on the client-side, although relaying on up to date libraries from the Internet. Implementation of communication and operations are done with JavaScript and Ajax (jQuery) technologies in engine component. The data parser component parses the GPX data for the annotated map. Maps are based on OpenStreetMap and the dynamic map content is displayed with OpenLayers library. These libraries are loaded from their home sites at the start.

D. *Example application*

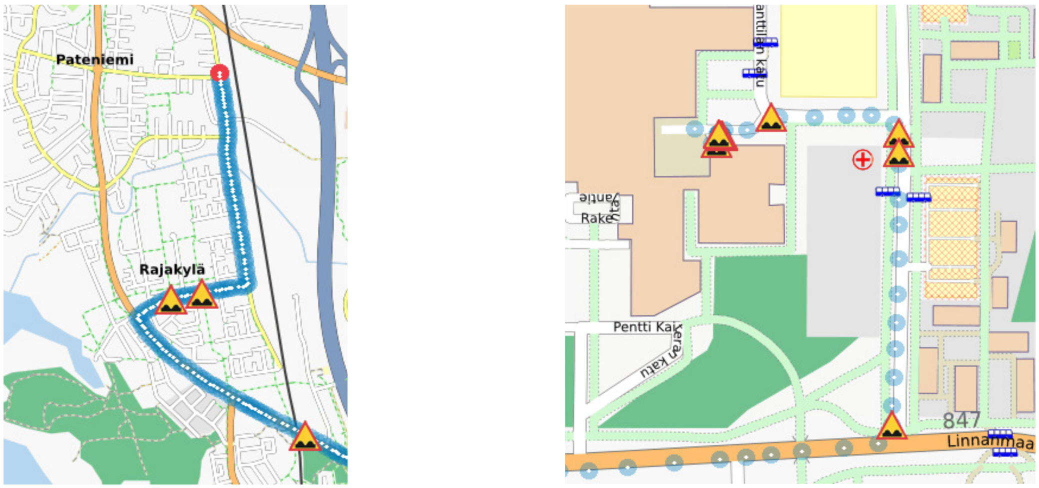
To demonstrate the system capabilities, integration of external data processing algorithms and heterogeneous data fusion in the components, we implemented an example application for recognition of the user's travelling mode and detection of anomalies of the road surface.

First, in the mobile phone sensor nodes, data are collected in the frequency of 38 hertz from the built-in accelerometer sensor and in the frequency of 1 hertz from the built-in GPS receiver. Data are stored temporarily on the phone memory and sent at given intervals as HTTP POST requests over the available communication network (for example GPRS or WLAN) to the remote platform end-point handler component, the data source wrapper. The current implementation of the collection software is written in Python for the Nokia N95 and in C for the N900 mobile phones.

In the middleware, we implemented components for map-matching for GPS location data, travel mode detection and road surface anomaly detection for the accelerometer data. Detailed descriptions of the data processing algorithms used in the applications can be found in [28][29][30]. First, the lowest level of data fusion occurs when accelerometer data and GPS location data are fused based on timestamps. The travel mode is detected from this data with timestamps and known locations and labeled accordingly, which constitutes an example of feature level fusion. Currently we can recognize several travel modes: stationary, walking,

TABLE 1. GPX DOCUMENT EXAMPLE

```
<gpx>
    <version>1.0</version>
    <creator>SDFA</creator>
    <wpt>
        <lat>65.059446</lat>
        <lon>25.472444</lon>
        <ele>9.5</ele>
        <type>Driving</type>
    </wpt>
    <rte>
        <name>Oulu</name>
        <desc>Testing</desc>
        <rtept>
          <wpt>
              <lat>65.059338</lat>
              <lon>25.473037</lon>
              <ele>10.1</ele>
              <type>Driving</type>
          </wpt>
          <wpt>
              <lat>65.059338</lat>
              <lon>25.473037</lon>
              <ele>9.8</ele>
              <type>Driving</type>
          </wpt>
        </rtept>
    </rte>
</gpx>
```



a)

b)

Figure 3. Visualization client example outputs: (a) a red dot showing the current location, blue dots showing path history and warning signs showing detected anomalies, (b) additional information such as bus stops and a health center (from OpenStreetMap) shown in a close-up.

jogging, bicycling and driving a vehicle. Next, the map-matching component receives the fused and labeled data from the previous step and matches the location to the nearest road segment. For the road segment data, we use static local OpenStreetMap database downloaded from the Internet and this can be considered being an infrastructure service in the system. As a result, location coordinates for the travel mode are updated accordingly and are published for further use in the system. This can be considered a second example of feature level data fusion, refining sensor data with infrastructure service. Next, the accelerometer data and travel modes labeled as "driving" with the corrected GPS location are received by the road surface anomaly detection component. It performs anomaly detection and publishes the data. When refined data are shown in the map in a web browser, it constitutes an example of decision level data fusion supporting the user's decision making in the example application. The data flow between components is shown in Figure 4 from the perspective of data fusion.

IV. Results and Discussion

Considering the requirements for cooperative traffic and wireless sensor network system implementation, we have demonstrated data-centric multi-tiered system architecture able to accomplish data fusion in various levels of data processing of sensor data to support decision making for the end-user. The example application demonstrates data fusion

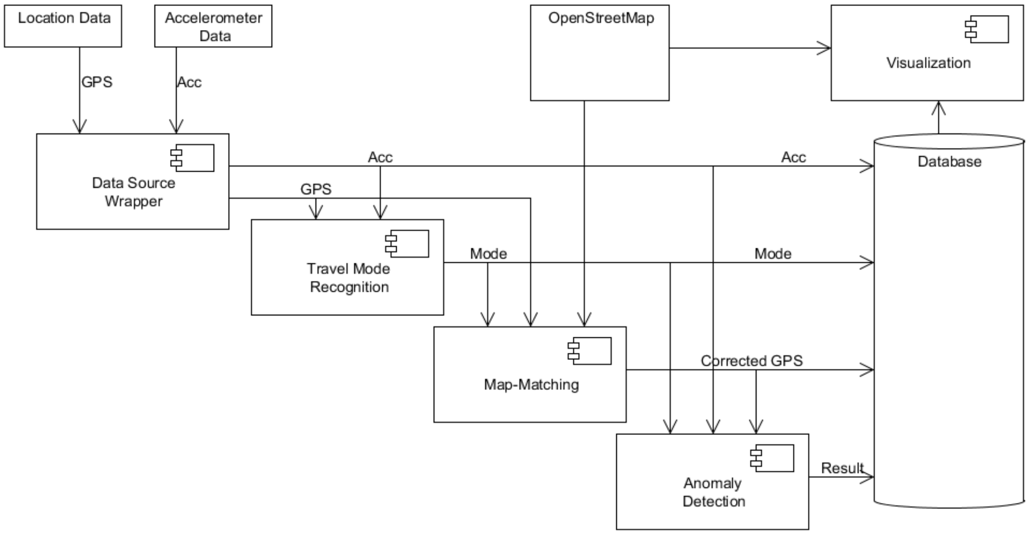


Figure 4. Data flow in the example application.

in the application domain, but the system allows also objective domain and the three-level hierarchies of data fusion. Informative and coordinative cooperation are demonstrated in the system. Collected raw and refined sensor data can be freely utilized by the system components, thus all information is available for information sharing. Applications can configure in components the required level of detail of the requested data. The system is also able to dynamically reconfigure functionality allowing runtime deployment of physical sensors and data processing components into the middleware. Furthermore, the system uses a mobile phone network and hence has support for asynchronous communication. Moreover, the system supports new types of future sensors and applications by providing public interfaces for data sources, for clients and into infrastructure services. The implemented example applications prove the system's ability to run multiple data processing components simultaneously, its ability for easy configuration and implementing component chaining for data processing. Scalability remains an open question until the applications are deployed for large-scale user tests in real-world situations, but this issue can be addressed with multiple server platforms all running the same middleware and perhaps even the same applications. These platforms can be easily interconnected using GSN's built-in features.

Compared to the OSGi platform [16][17], we consider both infrastructure services and individual vehicles being in the lowest layer. The upper layers constitute the data fusion capabilities and applications, which can be run in parallel. Concerning the Content Delivery Network [15] and the scenario given by [18], our system is different from the mobile phone scenario because of the routing is done in the mobile network and managed by the mobile network operator. Currently, we do not have a mechanism for clients to locate local services in the geographical area. The vehicle route prediction capability to reach the next node is not needed in the mobile phone network. In comparison to existing systems [10][12][13][14][27], we offer in addition runtime pluggable data processing components and heterogeneous data fusion capability, which are launched based on client application configuration and data requests. Google Maps for Mobile offer cooperative traffic applications, which are based on location data solely. CarTel, TJam and Nericell locally process data on the nodes. TJam also uses migratory services in nodes, which is a feature we might consider in the future. The Mobile Millennium uses roadside sensors and historical data, both of these features we would like to have in the future. In [19], a similar middleware is described with kernel, data module and decision module components. However, we do not utilize a general system queue and the middleware itself takes care of concurrent access to the data. The ubiquitous intelligent transportation system [21] also resembles our system. In their resource layer, they handle the preprocessing and classification of data, which is different from our approach as we have the possibility to access raw data in all layers in the system. Furthermore, we do not distribute functionality to separate layers for data processing, services or applications as data streams can be made available to all subscribers. For multiple communication technologies we utilize global endpoint handler components, which map to the ubiquitous network layer.

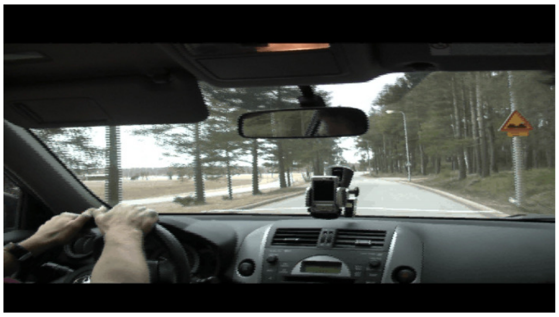


Figure 5. Test system set-up in vehicle. Mobile phone installed in car dock in the middle.

We conducted a small scale field testing for the system in real environment by walking and driving a vehicle in the city of Oulu, Finland, simultaneously collecting data by mobile phones with integrated accelerometer sensor and GPS receiver. Test system set-up in a vehicle can be seen in Figure 5. The test route consisted of 3 kilometers in a suburb area, 1 kilometer in a park and 1 kilometer in city center. Data was transmitted using HTTP over GPRS to the sink node in the intervals of 10, 30 and 60 seconds. See TABLE 2 for the results. Data transmission failed in 5 out of 99 testing runs (5.1%). There is no significant difference in failures concerning different data transmission intervals. However, an interesting detail was that all the transmissions failed around the same location in the city. For real-time applications, shorter intervals are feasible, but this depends on the requirements of the data processing algorithms. Another consideration for real-time applications is the speed of the vehicles, which contributes to the quality of collected data [15]. We also estimated the bandwidth required per node to be around three kilobytes per second for all the intervals, as the required message length for sending accelerometer data with GPS data is about 100 bytes and the accelerometer sensor runs in the mobile phone in 38Hz. In the future, we might have other sensors integrated into the mobile phone serving as a sensor node and we need to have estimation of the required bandwidth for the data amounts. When utilizing 3G technologies the available bandwidth can be significantly increased. As expected, during the testing, receiving of the GPS signal was sometimes disrupted in cities, 17.3% out of the total testing time, to be exact. However, this depends also on the GPS receiver hardware. The battery lifetime is always limited when using mobile

TABLE 2. FIELD TESTING RESULTS

Testing results	**Measurement intervals**		
	10s	***30s***	***60s***
Tests run	23	41	35
Failed transmissions	1	2	2

phones transmitting data continuously, but as our phones were often mounted in a vehicle, its power system was used. Cost of a single sensor node is not a considerable issue if the user's personal mobile phones can be utilized. What is more, the cost of air time and available bandwidth also varies locally.

Sometimes, the travel mode recognition did not work accordingly and we needed to abandon its results and publish all data to the anomaly detection component. The two mobile phone models had different accelerometer sensors, the Nokia N900 accelerometer being much more accurate than Nokia N95 accelerometer, so we needed to parameterize recognition components for each phone model. This means that online learning should be utilized to enable adapting the component for different physical sensors.

Also, we tested the feasibility of GSN middleware in a normal desktop PC (2.40 gigahertz processor with one gigabytes of RAM running Windows XP) with randomly generated data sets. We used 50 data sources simultaneously streaming 50 kilobytes of payload to the system at every 20 milliseconds. The delivery time from a data source wrapper to a data processing component was less than one millisecond per payload. This is very a promising result considering real-time capabilities of the system (req. 11). However, these tests have less real-world value as the used hardware components, available network bandwidth and programming skills of the developers largely contribute to the real-time performance in the prototype.

In this work, we have not considered security and privacy issues, which are unavoidable when collecting, fusing and visualizing data from multiple clients and with the integration of roadside infrastructure data and sensors. Koenders et al. [16] give a list of success factors for intelligent transportation systems. There is a need for communication partners in the system to work and penetration of similarly equipped vehicles is critical. This is where vehicle manufacturers need to co-operate by utilizing standards. Authorities can stimulate this, however clear benefits are needed as investments are required. For the individual users, personal privacy and reliability of the services needs to be guaranteed. Security and privacy concerns in the system can be addressed, for example by introducing user groups with restricted access to the data and results. In our prototype, the reliability is largely dependent on mobile phone network reliability and does not require additional investments from the users or traffic operators.

V. Conclusion and future work

We have started with a generic implementation of sensor network middleware for cooperative traffic applications, which meet the given requirements, drawn from the literature and from the project goals. Considering the goals, we have demonstrated modular component-based prototype sensor network implementation capable of sharing traffic related information collected by mobile sensor nodes and realizing the described usage scenarios. In the next phase, the developed prototype system will be deployed to be used with multiple sensor nodes and client applications. Increased scalability needs may require introducing multiple middleware platforms for distributing the system load. It would be possible to introduce remote interconnected application-specific platforms in the infrastructure providing data fusion featuring separate data sources in remote platforms.

An important goal of our future work is to implement a common ontology and a specific data fusion model in the middleware. This includes information on vehicle type, location, route, event, time, road condition and possible future application-specific parameters. Also, we will develop interfaces to sensors in instrumented vehicles and introduce usage of an on-board diagnostics module installed in test vehicles. These open many new possibilities as more detailed data of the vehicle behavior can be utilized. Usage of GPS data would allow us to feature also descriptive cooperation, such as the user's intentions or direction of movement, in the system [7]. Furthermore, interfaces to public Finnish government infrastructure services available in the Internet, such as DigiTraffic [31], a real-time traffic flow information service using roadside sensors, and DigiRoad, the static database containing road data and elements, allow us to widen the system perspective to the whole traffic system level. This information can be easily fused with additional local real-time data in the system and provide feedback to road users and operators.

Acknowledgment

The authors would like to thank all the personnel in the Sensor Data Fusion and Applications project. This work was funded by the National Technology Agency of Finland and was carried out in Sensor Data Fusion and Applications project as a part of the Cooperative Traffic research program of the Strategic Centre for Science, Technology and Innovation in the Field of ICT.

REFERENCES

[1] T. Leppanen, M. Perttunen, J. Riekki, and P. Kaipio, "Sensor Network Architecture for Cooperative Traffic Applications," in Proceedings of International Conference on Wireless and Mobile Communications, pp. 400-403, September 20-25, Valencia, Spain, 2010.

[2] F. Minarini, "eSafety for Road Transport: Investing in Preventive Safety and Co-operative Systems, the EU Approach," Advanced Microsystems for Automotive Applications, pp. 487-492, 2005.

[3] Cooperative Vehicle-Inftrastructure Systems. URL: http://www.cvisproject.org/ 12.07.2011.

[4] N. Faouzi, H. Leung, and A. Kurian, "Data fusion in intelligent transportation systems: Progress and challenges - A survey," Information Fusion, Vol. 12, 1, pp. 4, 2011.

[5] B. Dasarathy, "Sensor fusion potential exploitation-innovative architectures and illustrative applications," in Proceedings of the IEEE, Vol. 85, 1, pp. 24, 1997.

[6] C. Haythornthwaite, "Crowds and Communities: Light and Heavyweight Models of Peer Production," in Proceeedings of Hawaii International Conference on System Sciences, pp. 1-10, 2009.

[7] K. Lidstrom, T. Larsson, and L. Stranden, " Safety Considerations for Cooperating Vehicles using Wireless Communication," in

Proceedings of 5th IEEE International Conference on Industrial Informatics, Vol. 2, pp. 995, June 23-27, 2007.

[8] V. Dyo, "Middleware design for integration of sensor network and mobile devices," in Proceedings of the 2nd international doctoral symposium on Middleware, pp. 1-5, New York, NY, USA, 2007.

[9] M. Khanafer, M. Guennoun, and H. Mouftah, "WSN Architectures for Intelligent Transportation Systems," in Proceedings of 3rd International Conference on New Technologies, Mobility and Security, pp. 1, December 20-23, 2009.

[10] P. Mohan, V. Padmanabhan, and R. Ramjee, "Nericell: rich monitoring of road and traffic conditions using mobile smartphones," in Proceedings of the 6th ACM conference on Embedded network sensor systems, pp. 323-336, New York, NY, USA, 2008.

[11] The Finnish Cooperative Traffic Programme. URL: http://www.cooperativetraffic.fi/ 12.07.2011.

[12] The Mobile Millenium Project. URL: http://traffic.berkeley.edu/, 12.07.2011.

[13] Google Maps for Mobile. URL: http://www.google.com/mobile/products/maps.html, 12.07.2011.

[14] O. Riva, T. Nadeem, C. Borcea, and L. Iftode, "Context-Aware Migratory Services in Ad Hoc Networks," IEEE Transactions on Mobile Computing, Vol. 6, 12, pp. 1313-1328, 2007.

[15] M. Elshenawy, M. El-Darieby, and B. Abdulhai, "Scalable and location-aware its content management in vehicular environments," in proceedings of IEEE Intelligent Vehicles Symposium, pp. 627, June, 2010.

[16] E. Koenders and J. Vreeswijk, "Cooperative infrastructure," in Proceedings of IEEE Intelligent Vehicles Symposium, pp. 721, June, 2008.

[17] O. Brickley, S. Chong, M. Klepal, A. Tabatabaei, and D. Pesch, "A Data Dissemination Strategy for Cooperative Vehicular Systems," in Proceedings of 65th IEEE Vehicular Technology Conference, pp. 2501, April, 2007.

[18] D. Tacconi, D. Miorandi, J. Carreras, F. Chiti, and R. Fantacci, "Using wireless sensor networks to support intelligent transportation systems," Ad Hoc Networks, Vol. 8, 5, pp. 462, 2010.

[19] S. Bengochea, A. Talamona, and M. Parent, " A software framework for vehicle-infrastructure cooperative applications," in Proceeedings of IEEE Intelligent Transportation Systems, pp. 797, September, 2005.

[20] Digiroad – A National Road and Street Database. URL: http://www.digiroad.fi/en_GB/, 12.07.2011.

[21] W. Lv, B. Du, D. Ma, T. Zhu, and C. Wang, "Applied research of data sensing and service to ubiquitous intelligent transportation system," Frontiers of Computer Science in China, Vol. 4, 3, pp. 417-426, 2010.

[22] Y. Yu, B. Krishnamachari, and V. Prasanna, "Issues in Designing Middleware for Wireless Sensor Networks," IEEE Network, Vol. 18, pp. 15-21, 2003.

[23] B. Ducourthial, "About efficiency in wireless communication frameworks on vehicular networks," in Proceedings of The First International Workshop on Wireless Networking for Intelligent Transportation Systems, pp. 3:1-3:9, New York, NY, USA, 2007.

[24] K. Modukuri, S. Hariri, N. Chalfoun, and M. Yousif, "Autonomous Middleware Framework for Sensor Networks," Journal of Pervasive Computing and Communications, Vol. 1, pp. 337-345, 2008.

[25] T. Kosch, I. Kulp, M. Bechler, M. Strassberger, B. Weyl, and R. Lasowski," Communication architecture for cooperative systems in Europe", IEEE Communications Magazine, Vol. 47, 5, pp. 116, May, 2009.

[26] K. Aberer, M. Hauswirth, and A. Salehi, "A middleware for fast and flexible sensor network deployment," in Proceedings of the 32nd international conference on Very large data bases, Seoul, Korea, pp. 1199-1202, 2006.

[27] B. Hull, V. Bychkovsky, Y. Zhang, K. Chen, M. Goraczko, A. Miu, E. Shih, H Balakrishnan, and S. Madden, "CarTel: a distributed mobile sensor computing system," in Proceedings of the 4th international conference on Embedded networked sensor systems, pp. 125-138, New York, NY, USA, 2006.

[28] O. Mazhelis, "Using recursive Bayesian estimation for matching GPS measurements to imperfect road network data," in Proceedings of 13th International IEEE Conference on Intelligent Transportation Systems, pp. 1492, September, 2010.

[29] J. Kantola, M. Perttunen, T. Leppanen, J. Collin, and J. Riekki, "Context Awareness for GPS-Enabled Phones," in Proceedings of ION Technical Meeting, Manassas, VA, USA, 2010.

[30] M. Perttunen, O. Mazhelis, F. Cong, M. Kauppila, T. Leppänen, J. Kantola, J. Collin, S. Pirttikangas, J. Haverinen, T. Ristaniemi, and J. Riekki, "Distributed Road Surface Condition Monitoring Using Mobile Phones," in the 8th International Conference on Ubiquitous Intelligence and Computing, Banff, Canada, September 2-4, 2011.

[31] Digitraffic. URL: http://www.infotripla.fi/digitraffic/english/, 12.07.2011.

A New Wireless Architecture for In-Flight Entertainment Systems Inside Aircraft Cabin

Ahmed AKL, Thierry GAYRAUD, and Pascal BERTHOU
CNRS ; LAAS ; 7 avenue du Colonel Roche, F-31077
Universite de Toulouse ; UPS, INSA, INP, ISAE ; LAAS ; F-31077
Toulouse, France
{aakl, gayraud, berthou}@laas.fr

Abstract—**A primary difficulty when investigating communication requirements rises when a very specific field as an aircraft cabin is considered. The diverse needs of passengers are often incompatible to the strict constraints inside the cabin. Nowadays, *In-Flight Entertainment (IFE)* systems, for instance, are widely spread in modern flights. An IFE system usually consists of a Seat Electronic Box, the passengers terminal hardware, plus a Passengers Control Unit, the remote control to select the service, and a Visual Display Unit, the screen. Using the wireless technology in these systems can increase the satisfaction level of both the passengers and the avionics companies. However, the inside of the cabin is not a flexible environment; reliability and safety are two mandatory requirements, so different constrains are imposed. This means that off-shelf technologies (hardware including antennas, network topology, network protocols and services) are usually not suitable for such environment. Consequently, a new architecture has to be designed and implemented. This paper aims at integrating heterogeneous available communication technologies, showing their pros and cons, within this context, while considering the imposed communication restrictions inside the aircrafts cabin. From that, a new wireless heterogeneous architecture is proposed. In addition, to be able to use such architecture, we propose a new protocol, which utilizes the smart antennas technology to allow *Passenger Control Units* to be recognized and configured autonomously without any external intervention.**

Keywords-IFE system, Wireless networks, PLC, Smart antenna, Protocol engineering and evaluation

I. Introduction

In recent years, market surveys have revealed a surprising and growing trend in the importance of *In-Flight Entertainment* (IFE) with regard to choice of airline. With modern long range aircraft the need for ”stop-over” has been reduced, so the duration of flights has also been increased. Air flights, especially long distance, may expose passengers to discomfort and even stress. IFE can provide stress reduction entertainment services to the passenger. The IFE system is an approach that can utilize the wireless technology for the purpose of exchanging data -in both directions-between passengers and the entertainment system. Although wired communication gives better performance than wireless communication, but most of modern personal devices are based on the wireless technology, so providing wireless capabilities is essential for efficient utilization of these devices. Moreover, a wireless headphones is more satisfactory than a wired headset attached to the seat arm. We are not targeting to make the system fully wireless, but to emphasize the idea that different technologies can be used in the IFE system to improve the passenger’s service satisfaction level [1].

As stated by Niebla [2], users are becoming more and more familiar to personal equipments, such as mobile phones, laptops, and PDAs. This shows the importance of providing aircrafts with facilities that support these equipments.

Nowadays, different wireless technologies with different capabilities exist in the market such as WiFi, Bluetooth, infrared, and Wirelss Universal Serial Bus (WUSB). However, these technologies may have certain characteristics that make them unsuitable for usage inside the cabin. Although it was proven that WiFi and Bluetooth can be used inside the cabin with no fear of interfering with navigational equipments [3], it is still difficult to use them in large numbers since performance degrades due to allocating them in a small area inside a metallic tunnel (i.e., cabin), which is full of different obstacles (i.e., seats). In addition, the normal way of setting these technologies is usually done through predefined identifiers (i.e., IP address) or a user key (i.e., Bluetooth authentication key). Both techniques don’t match the constrains inside the cabin, where devices must be designed without any predefined identifiers and to be configured without any user intervention

However, usage of the wireless technology will help in decreasing the connecting wires; this is a valuable criterion in aircrafts designing. Using off-shelf technologies inside the cabin is usually not applicable when using them in the usual manner; the environment inside the cabin has very strict constrains since safety is a major requirement. Consequently, using just one technology can’t be the optimum solution. In fact, using a combination of different technologies can provide a better service while overcoming the existing constrains. The way passengers use their *Personal Electronic Devices (PED)* (i.e., mobile phones, laptop, etc.) was usually done through specialized devices [2], [3]. Nevertheless, there is no current research to use a combination of off-shelf technologies inside the cabin.

The IFE market attracted many companies to submit different IFE solutions. *Thales* [4], an avionic systems provider, introduced its own IFE system that can be tailored according to the needs of the airlines companies. Some other companies provide dedicated solutions for certain parts of the IFE systems; the *AeroMobile* [5] is a GSM service provider for the aviation industry that allows passengers to use their mobile phones and devices safely during the flight. Passengers can connect to an AeroMobile pico cell located inside the craft which relays the text messages and calls to a satellite link which sends them to the ground network. The AeroMobile system manages all the cellular devices onboard, so signal strength can be ketp at a minimum value to minimize interference. This system is adopted by Panasonic to be part of its in-flight cellular phone component. *FlyNet* [6] is an onboard communication service provided by Lufthansa to allow passengers to connect to the Internet during their flight.

In the next section, we will introduce an overview for the IFE system showing its components and requirements. Section II-C introduces the communication challenges that face the IFE systems. Section III shows the proposed available technologies that can be used to overcome the challenges in such environment. Section III-B introduces our approach for integrating them together. Section IV presents our proposed protocol that utilizes the smart antenna capabilities to connect the IFE devices. Section V shows how the protocol was verified and validated through a *Unified Modeling Language (UML)* model and NS2 simulation. Finally, we present the conclusion and future work.

II. Background and relevant work

The recognized economics of wireless networks and communications systems have made them an attractive target for environments where individual wires are cumbersome. An airplane cabin is such an environment. Dwayne [7] said that due to the need of rapidly reconfiguring the cabin's seating, it is further assumed that wireless networking, rather than cable or fiber optics, must be used to interconnect passenger's entertainment equipment with other elements of the system.

The use of wireless communication technologies on board of an aircraft provides an opportunity to remove wiring and save weight on the aircraft. The weight savings can be directly measured in terms of fuel savings and improved operating economics over the life time of an aircraft. However, there is a need to ensure that there is no interference with the aircraft's communication and navigation systems.

A. The need for IFE systems

Hao [8] mentioned that the enclosed environment of the aircraft can cause discomfort or even problems to passengers. IFE systems can greatly reduce these negative effects. This can be done by using e-books, video/audio broadcasting, games, internet, and On Demand services. The fact that passengers come from highly heterogeneous pools (such as age, gender, ethnicity, etc.) causes an impact on the adaptive interface systems.

As mentioned by Fariba [9], the IFE systems usually include screen-based, audio and communication systems. The screen-based products include video systems enabling passengers to watch movies, news and sports. These systems had progressed into video-on-demand, allowing passengers to have control when they watch movies. Air map display is another service that allows them to locate their flight's route. Exterior-view cameras also enable passengers to have the pilot's forward view on take-off and landing on their personal TV screens. Audio systems include different types of music channels and special programs recorded for the airlines. Communication systems include intra-communications with devices such as telephones, facsimile and in-seat power supplies, and inter-communications between the screen-based system and its subsystems (i.e., remote control).

B. The IFE system's components

In fact, the entertainment starts from the passenger's seat design where most of the IFE system components are embedded. Wiring cables connect together all of the electronic devices in the seat as well as connecting them to the whole system in the cabin. They run through the cabin's walls, floor, and seats. Unfortunately, conveying signals and power to seats with a connector for each seat would cause reliability and maintenance problems, and hinder timely cabin reconfiguration.

Nowadays, IFE systems are interactive systems, so a *Passenger Control Unit* (PCU) is usually needed to control the surrounding devices. The PCU should be compact and easily held. Moreover, the pocket holding the PCU has to be placed in a way that makes it reachable and not to affect the passenger's comfort. At the beginning, PCUs used to be fixed aside to the *Visual Display Unit (VDU)* at the back of the front seat. This orientation introduced a problem when the passenger setting beside the window wants to move to the corridor; where all his neighbors have to replace their PCUs to allow him to pass. To overcome this problem, PCUs are now connected to their VDUs through wires passing via their seat.

A *Visual Display Unit* (VDU) is usually fixed to the back of the front seat. Depending on the required features of the system, ordinary displays can be used to display the visual contents or touch screens can be installed to act as input devices. Another orientation is to be fixed in the ceiling as a shared display for a group of seats.

A *Seat Electronic Box* (SEB) can be used to connect the system's different components together. It is used to connect the passenger's devices and the IFE system instead of having a separate channel for each signal. For example, to transmit communication and video signals, two different networks should be available if the SEB is not used. When using the SEB, the communication and video devices are connected directly to it to convey signals to the rest of the IFE system through one single network. Then, it simplifies and facilitates maintenance procedure since malfunctioning devices can be easily replaced without affecting the IFE connections.

Halid [10] stated that *Power Line Communication (PLC)* can provide a way of communication through power lines networks. Power lines and communication networks have different physical characteristics, so a PLC modem must be used as an interface between the two networks. They must be designed to provide accepted network operation under typical power lines transmission conditions. However, power lines are not designed as a good transmission media. It suffers from attenuation, fading, and noise. Nevertheless, the great advances in digital signal processing, error detection and correction, modulation, media access control techniques encourage the use of PLC in communication field.

A part from the physical requirements of an IFE system, there are especial operational requirements to cope with its expected functionality.

- *Self configuration:* The system's units must be self configurable without any external intervention [11]. They must start, run, and cope with any changes in the system autonomously since the crew members are too busy to handle such operations; and even if they have the time and effort, they may lack the required technical background.
- *Minimized wiring and power consumption:* As mentioned before, these are valuable criteria in aircrafts' designing where wiring is considered as excess weight that can be expressed in terms of excess fuel consumption. Castagne [12] addressed the effect of weight over the *Direct Operating Costs (DOC)* by calculating the impact of weight increase on annual fuel consumption $I_{DOC} = \frac{f_c \times n_{seat}}{W_{op}}$, where f_c is the annual fuel consumption, n_{seat} is the number of seats, W_{op} is the operating weight. In fact, calculating the exact saved weight depends on various factors such as the cabin structure, cables routing, type of cables, etc. However, we can imagine the amount of saved weight through the contribution of Hurley [13]. He showed that cables flexibility is a design factor that affects cable routing; this means that cables may be extended through longer paths to reach its destination. Moreover, special jumpers and connectors are needed to attach cables, and groups of cables can be attached to a cable harness; this indicates that weight calculation will include the added weight of accompanied materials and equipments. It also indicates that wiring can hinder the maintenance process as well as imposing difficulties in changing the cabin's layout.
- *Easy to use:* Passengers of no knowledge about using modern technology must be able to use the system easily. For example, the PCU controls (i.e., Volume, Rewind, Forward, etc.) are known for almost everyone; especial purpose controls such as Settings, Mode, etc. can be carefully manipulated and, if used, to be provided by explanatory information when possible.
- *It has to be easily replaced:* In case of failure in one of the system's units, the unit has to be changed instantaneously and easily without the need of any technician, especially if the failure happened during the flight time.
- *Topology is not dynamic:* Once the network is setup, there will be no change in topology till the end of the flight unless a unit fails. In case of replacing a failed unit, it must be self configurable to join the network again. However, if passengers attached their PEDs to the system, some dynamism can be considered.
- *Scalability:* The system must be scalable to suit plans of different sizes and different seats layout.

C. Communication challenges

Although aircraft security may be seen as another burden due to its very strict requirements, but it is mandatory to be included during the design of communication and data services. A major concern for using wireless devices in aircraft cabin is their interference with the aircraft's communication and navigation system, especially unintended interference from passenger's *Personal Electronic Devices* (PED). Holzbock [14] said that the installed navigation and communication systems on the aircraft are designed to be sensitive to electromagnetic signals, so they can be protected against passenger's emitters by means of frequency separation. In addition, Jahn [3] mentioned that there are two types of PEDs' interference, intentional and spurious. The former is the emissions used to transmit data over the PED's allocated frequency band. The latter is the emissions due to the RF noise level.

Moreover, the existing systems suffer from bandwidth limitations; the trend toward bandwidth-consuming Internet services currently cannot be satisfied [3]. Passenger number and categories can be considered as a factor that affects network scalability. For example, the network bandwidth should be increased if the number of the first class passengers was increased to support the increasing need for video stream.

It is stated by Holzbock [14] that existing indoor channel models mainly investigate office or home environments, thus these models may not be appropriate for modeling an aircraft cabin channel. Attenuation of walls and multi path effects in a normal indoor environment are effects, which are not expected to be comparable to the effect of the higher obstacle density in a metallic tunnel. The elongated structure of a cabin causes smaller losses, than that expected in other type of room shapes. However, the power addition of local signal paths can lead to fading of the signal in particular points. In addition, small movements of the receiver can have a substantial effect on reception. The same opinion was emphasized by Diaz [15].

Different efforts were held to overcome this problem, Youssef [16] used the commercial software package *Wireless Insite* to model the electromagnetic propagation of different wireless access points inside different types of aircrafts. Moraitis [17] held a measurement campaign inside a Boeing 737-400 aircraft to obtain a propagation development model for three different frequencies, 1.8, 2.1, and 2.45GHz which represent the GSM, UMTS, and WLAN and Bluetooth technologies, respectively. *Path Loss Modeling* was presented by the formula:

$$\overline{PL}(d) = FSL(d_o) + 10n\, log_{10}(\tfrac{d}{d_o}) \;\; (dB)$$

Where $\overline{PL}(d))$ is the *average path loss* value in (dB) at a distance *d* in (m) from the transmitter to the receiver, $FSL(d_o)$ is the *free-space path loss* in (dB) at a reference distance d_o, and *n is the path loss exponent* (decay rate). The wave-guide effect was expected to be noticeable since the cabin is considered as a long metallic tunnel. Thus, the value of *n* should be lower than 2. However, his measurements showed that *n* was found to be slightly larger than 2; showing that the wave-guide effect was counterbalanced. This is due to the thick carpet covering of the floor, non reflective textile covering the seats, and the gaps between rows of seats trap the transmitted rays. According to his measurements, he represented the *average seat inserting loss* $\overline{L}_{seat}$ due to the backrests by the formula:

$$\overline{L}_{seat} = \tfrac{1}{N} \textstyle\sum_{i=1}^{N} (\overline{PL}_i^{meas} - FSL_i) \;\; (dB)$$

Where FSL_i is the *free-space loss* at the *i*-th seat, $\overline{PL}_i^{meas}$ is the average measured path loss at the *i*-th passenger Seat, and *N* is the total number of seats

The effect of human presence inside the cabin over *Ultra Wide Band (UWB)* propagation was addressed by Chiu [18]. He considered three scenarios for placing the transmitter and receivers; *Ceiling to Headrest*, *Ceiling to Armrest*, and *Headrest to Armrest*; for each scenario the cabin was empty, partially filled, and fully filled with passengers. For the first two scenarios, the transmitter was mounted at the ceiling and receiving antennas were mounted at the headrest and armrest levels. For the last scenario, the transmitter was located at the headrest level, and the receiver was located at the armrest level. The measurements showed that the path gain dropped by no more than a few dB for the *Ceiling to Headrest* scenario. and dropped by up to 10 dB for the *Ceiling to Armrest* and *Headrest to Armrest* scenarios. The measurements concluded that the presence of human body inside the cabin affects the wireless propagation and must be considered during wireless design.

Another challenge is that the cabin of an aircraft and the aeronautical environment in general define a very specific scenario that presents several constraints, which will affect the coverage and capacity planning. This is due to the fact that the space is very limited in an aircraft cabin, and its design allows installing equipment only in specific locations, where the configuration of the panels is easy to disassemble for maintenance [2]. Therefore, the replacement technique associated with the IFE system components, may affect the welling of the companies to use them. Replacing time consuming parts can lead to a long aircraft downtime or flight delays. Also, a device that fails during the flight, and is difficult to be replaced, will cause the passenger to be unsatisfied. Consequently, it is advisable to design components that are easily replaced with the minimum required technical skill.

III. From communication technologies to heterogeneous architecture

As mentioned by Holzbock [14], wireless Cabin aims at developing a communication infrastructure consisting of heterogeneous wireless access networks to provide aircraft passengers and crew members with access to IFE system. Passengers are able to access different services through state-of-the-art wireless access technologies such as W-LAN IEEE802.11, and Bluetooth.

A. Available technologies

Regardless of different technologies available in the market, we are concerned with the ones that can be utilized inside the cabin

1) Ethernet: Ethernet is currently the standard for wired communication in different fields. Haydn [19] showed that it is characterized by interesting features such as good communication performance, scalability, high availability, and resistance to external noise. However, Ethernet cabling is considered a burden for aircraft design

2) Wireless LAN: WLAN is a well known technology used in different commercial, industrial, and home devices, and can easily coexist with other technologies to form a heterogeneous network [2]. Jim [20] stated that WLAN and Bluetooth technologies are two complementary not a competing technologies. They can cooperate together to provide users with different connecting services.

3) Wireless USB: Universal Serial Bus (USB) technology allows different peripherals to be connected to the same PC more easily and efficiently than other technologies such as serial and parallel ports. However, cables are still needed to connect the devices. This raised the issue of Wireless USB (WUSB) where the devices can have the same connectivity through a wireless technology. Neal [21] stated that although it is difficult to achieve a wireless performance similar to wired USB, but the rapid improvements in radio communication can make WUSB a competent rival. It is based on the *Ultra Wide Band (UWB)* technology. In Europe, it supports a frequency range from 3.1 to 4.8 GHz. Moreover, Udar [22] mentioned that UWB communication is suitable for short range communications, which can be extended by the use of mesh networks. Although WUSB was designed to satisfy client needs, but it can also be used in a data centre environment. He discussed how WUSB characteristics can match such environment. This application can be of a great help in IFE systems, which strive to massive data communication to support multimedia services and minimizing the connection cables. Moreover, Jong [23] discussed the design issues related to WUSB. He stated that WUSB can support up to 480Mbps, but in real world it doesn't give the promised values; and he showed the effect of design parameters on the device's performance.

4) Power Line Communication: A PLC network can be used to convey data signals over cables dedicated to carry electrical power; where PLC modems are used to convert data from the digital signal level to the high power level; and vice versa. Using an existing wiring infrastructure can dramatically reduce costs and effort for setting up a communication network. Moreover, it can decrease the time needed for reconfiguring the cabin's layout since less cables are going to be relocated.

However, such technology suffers from different problems. A power line cable works as an antenna that can produce *Electromagnetic Emissions (EME)*. Thus, the PLC device must be *Electromagnetic Compatible (EMC)* to the surrounding environment. This means that it must not produce intolerable EME, and not to be susceptible to them. To overcome this problem, the transmission power shouldn't be high in order not to disturb other communicating devices [10]. However, working on a limited power signal makes the system sensitive for external noise. In spite of this, the PLC devices can work without concerns of external interference due to two reasons. Firstly, the PLC is divided into segments; this minimizes signal attenuation. Secondly, all the cabin's devices are designed according to strict rules that prevent EME high enough to interfere with the surrounding devices.

5) Smart Antennas: A smart antenna is a multi-element antenna where the emitted signal from each element can be controlled to direct the antenna's beam towards a certain direction as well as controlling the transmission power [24]. This feature is of great importance for ad-hoc networks domain where interference and power saving are two major issues.

Moreover, Okamoto [25] stated that smart antennas can provide the wireless environment with different advantages. First, it can significantly reduce the multi-path fading effect. Second, it minimizes the power consumption required for communication. Third, it can improve the system's *Signal-to-Interference Ratio (SIR)*.

B. The heterogeneous architecture

IFE system is a field starving for unusual ideas. Passengers can be satisfied by receiving services dedicated to a single user, but it will be more amusing if they can be offered services for multiple users, where passengers of similar interests can share their time. Using a single communication technology inside the cabin can't yield satisfactory results, but a combination of different technologies can have a great impact on the provided services.

The term heterogeneous in the networking domain usually implies the mix between wireless and wired networks. However, we mean by heterogeneity, the existence of different networking technologies cooperating together to achieve certain services. The network can be divided into *User Technology* and *System Technology*. A *User Technology* is the technology apparent and directly used by the user (i.e., Bluetooth, WiFi, etc.) to connect his devices to the system. A *System Technology* is the technology used by the system and is hidden from the user (i.e., PLC).

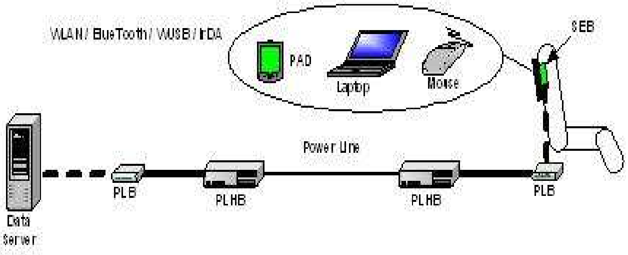


Figure 1. Heterogeneous network architecture

In our proposed heterogeneous architecture, we suggest the usage of a PLC network to convey data between a data server and the passengers' seats where he use his PEDs. In addition, wireless *Access Points (APs)* are connected to the PLC backbone as well; while WUSB is used to provide a way to connect some USB devices to the network (see Figure 1).

C. Architecture evaluation

In this section, we will introduce some experimentation results to show the applicability of the proposed technologies for a cabin's IFE system

1) PLC: The proposed PLC system is shown in Figure 1; it consists of a *Power Line Head Box (PLHB)* and a *Power Line Box (PLB)*. The PLHB connects the two terminals of the power line to connect the data server with the seats. Each PLHB service a group of seats, which are equipped with PLB per seat. The PLB is responsible for distributing the signal received by the PLHB to the seat's SEB. both PLHB and PLB devices can be configured through their internal web interface to define their IP address and other configuration parameters.

The MGEN (version 4.2) [26] traffic generator was used to emulate the traffic produced by the data server, and a laptop was used as a substitute to the SEBs. A traffic of 3480 bit/sec was used to represent each seat, so a total traffic of 3480×20 bit/sec was injected into the PLHB. The target of the test was to collect different statistics to study the behavior of the PLC system by injecting periodic traffic flows at constant intervals.

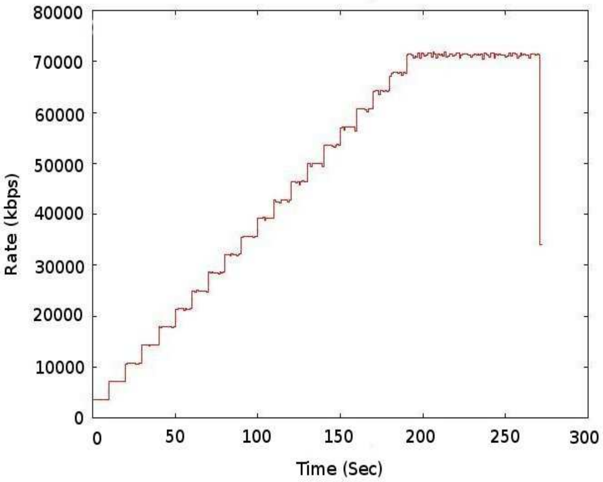


Figure 2. Flow rate of all flows

Figure 2 shows the sum of flows' rates. The stepping of the flow rate is constant indicating that the PLC connection is able to carry the injected traffic, and each PLHB can support up to 20 PLBs at a rate of 3480 bit/sec for each PLB. In addition, Figure 3 represents the packet count of the first flow. It is clear that the packet count stayed constant from the start to the end of the simulation without being affected by the injection of the subsequent flows. This emphasis the same results derived from Figure 2.

However, it is normal to have packet dropping during transmission. Figure 4 shows the obtained loss fraction; it is less than 0.05, which can be considered as a good value. Such configuration can provide the IFE with a way to provide video services by using the existing power cabling.

2) WLAN: We held different NS2 [27] simulations to propose a good distribution for the wireless *Access Points (APs)* inside the cabin. We used the same cabin configuration used by Alexandaros [28]. The cabin consists of 26 rows with 6 seats each (3 on each side of the aisle); this gives a total of 156 seats. The cabin is 21m long and 3.54m wide. The rows' separation distance is 81cm. By default, NS2 uses the standard 802.11 protocol which supports 2Mb rate. We used the more reliable 2Mb physical layer than the upper

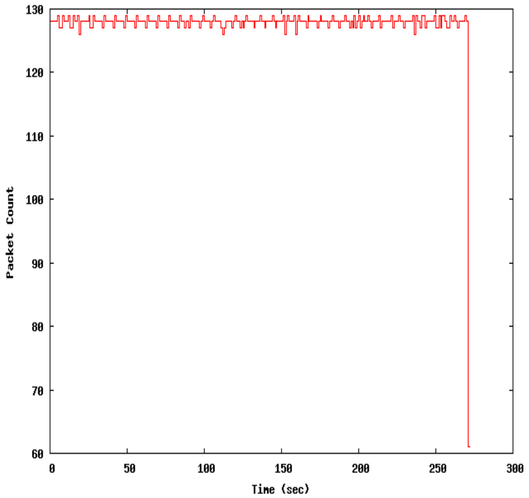


Figure 3. Packet count of the first flow

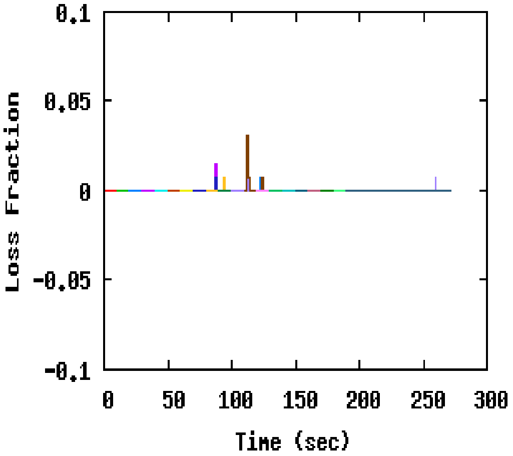


Figure 4. Loss Fraction

standard (i.e., 802.11b,g) since the transmission environment inside the cabin is not optimum.

A wireless node - representing a passenger's device - is located at the position of each seat, and APs are used to connect them with the data server. Using large number of wireless devices in a very narrow metallic tunnel like the cabin has a dramatic effect on the network's performance. For this reason, we are studying the effect of using frequency separation between APs. However, we need to determine the minimum number of APs required to cover the whole cabin, and their distribution inside the cabin, so we experimented with three scenarios. In scenario 'A', all nodes (each has a transmission range covering the whole cabin) are using the same communication channel. Scenario 'B' uses nodes with short transmission range, which allows connection only to the nearest Access Point (AP), while using the same channel. Scenario 'C' shows nodes with short transmission range and using channel separation. The channel separation in the third scenario is based on the fact that 802.11 only allows the usage of three non interfering channels (i.e., channels 1, 6, and 11). The impact of the three scenarios over average throughput, average delay, and number of transmitted packets is studied.

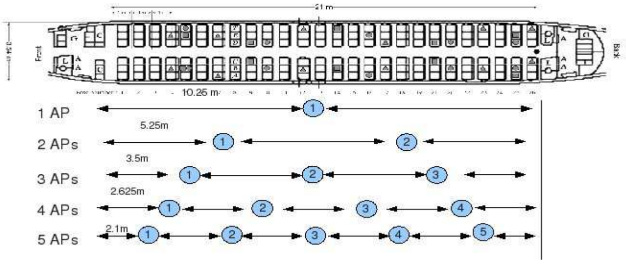


Figure 5. APs distribution

Each scenario was repeated 5 times while using different numbers of Aps located at the aisle. We started by using one AP and the number is incremented until we reached the maximum number of APs, which was determined according to the cabin's dimensions. The AP's transmission power was adjusted to minimize the transmission range, so the signal can travel a distance just enough to reach the seat beside the window in order to minimize the effect of its reflection. This allowed us to use a maximum number of 5 APs (Figure 5).

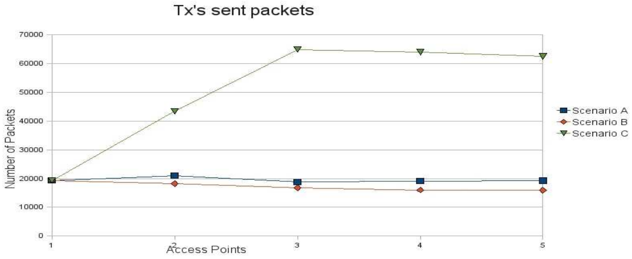


Figure 6. Packets sent by the transmitter

For all scenarios, the nodes (156 node + APs) were configured to have a large queue that can hold up to 1000 packets in order to prevent packet dropping. The transmission power was adjusted to 10mW as the minimum value defined in the 802.11 standard. In scenarios that use different channels, Channels 1, 6, and 11 were adjusted to their frequencies 2.412e9 Hz, 2.437e9 Hz, and 2.462e9 Hz respectively. The Rx threshold was determined according to the required transmission range. It was calculated by the tool "Threshold" provided as a separate program with the NS2 simulator. Table I shows the values used with each number of APs. For each simulation, the APs were distributed evenly throughout the aisle to provide a full coverage for the cabin.

Number of APs	Transmission range (meters)	Rx threshold
1	10.5	8.97474e-9
2	5.25	3.58989e-8
3	3.5	8.07726e-8
4	2.625	1.43596e-7
5	2.1	2.24368e-7

Table I
Rx threshold values

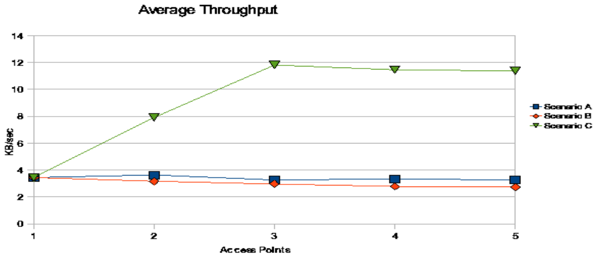


Figure 7. Average Throughput

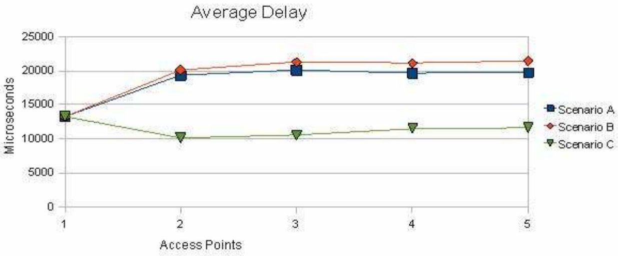


Figure 8. Average Delay

When comparing the three scenarios we can find that using just different number of APs doesn't have a great impact on the networks performance, but when accompanied with channel separation the networks performance is dramatically enhanced. Figure 6, Figure 7, and Figure 8 combine the results of scenarios A, B, and C. It is noticeable that there is no great difference between scenario A and B; this is due to the existence of large number of nodes in a small area. In addition, there are many nodes in the shared zone between every two APs. In this zone, nodes are able to detect two APs, but they select just one of them. In other words, on the physical level signals are interfering, while on the logical level only one AP is seen. However, as the number of APs increase, the difference between scenario A and B starts to increase slightly; this is because the number of nodes in the shared zone becomes less, so the interference decreases. On the contrary, when using channel separation (i.e., scenario C) performance was dramatically enhanced after using 3 APs

It is worthy to note that the number of nodes assigned to each AP affects its performance; the fewer nodes we use, the higher performance we get. When using 1, 2, 3, 4, and 5 APs, each AP will have 156, 78, 52, 39, and 32 nodes respectively. However, the difference in the number of assigned nodes with 3, 4, and 5 APs is small. This justifies the reason for saturation after using more than 3 APs; where APs almost handles the same amount of APs

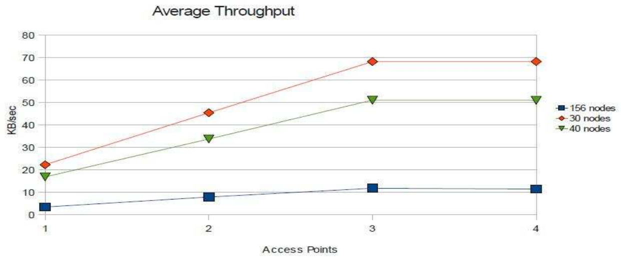


Figure 9. Average Throughput for different number of nodes

The results in Figure 7 showed that the best values introduced by scenario C is relatively small (i.e. 12 KB/sec); this is due to the high number of nodes within a small area causing interference between them. However, we have to highlight some issues. First, the APs are not used to convey video streaming; they are used to provide the passenger's PED with simple internet services, while video entertainment is achieved through the VDU dedicated for the IFE system. Second, we used the more reliable 802.11 version that supports 2 MB rate; this means that the passengers in the cabin can not be connected to the outside with a rate higher than that. Third, GPRS connections support a rate of 100 kb/sec, so that, if the performance degraded more than indicated in Figure 7 the PEDs can achieve internet services with the GPRS rate.

To study the effect of nodes' number we repeated scenario C with 30 and 40 nodes. Figure 9 shows that the network almost has the same behavior and the *Average Throughput* was greatly enhanced when the number of nodes decreased. This behavior shows the Airlines companies that they have two trade offs, either to provide a free internet service with poor performance, or to have a controlled access to keep acceptable performance. Consequently, a free internet access can be introduced in small flights since not all passengers are going to access the internet simultaneously, some of them will use other IFE services, or sleep, or talk. On the other hand, a prepaid internet access can be used in large flights where number of internet users will be considerably large.

3) WUSB: WUSB to connect passenger's devices seems to be an appealing solution since it doesn't require any additional adapters or connectors, and avoids interference with other wireless technologies (i.e., WLAN, Bluetooth, etc.) by using different bandwidth.

Elapsed time		Delay ratio
WUSB	915secs	((344/915)*100)-100=62.4%
Wired USB	344secs	

Table II
WUSB VS WIRED USB

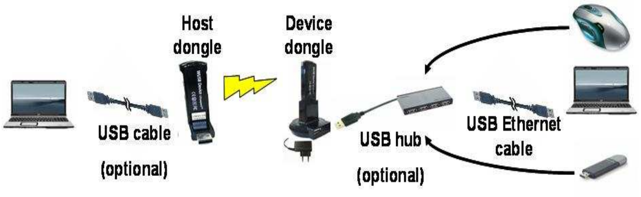


Figure 10. WUSB test-bed

Figure 10 shows our WUSB experimentations test-bed. WUSB *Host* and *Device* dongles were used to connect USB devices. The *Host* dongle is connected to the computer USB port, while the *Device* dongle connects the USB devices. The dongles driver allows changing of transmission as well as the transmission channels.

- *Connecting different USB devices:*Connecting multiple USB devices (i.e., mouse, and keyboard) was done in two different ways; firstly by using two Device dongles for each USB device, secondly by using a USB hub. The results of the first approach were not satisfactory because the two dongles were using the same channel causing interference between them. However, the Host dongle has the ability to choose between seven different channels. In other words, it is possible to use seven Hosts at the same transmission range without any interference between them. The second approach gave better performance. Moreover, a hub is much more economical than using a WUSB dongle dedicated for each device.
- *File transfer:*It is important to know if WUSB is capable of transferring large files, and to what extent it is comparable to wired USB, so 4064 files of size 892MB were transferred to a flash USB storage device using WUSB and wired USB.
  The results shown in Table II indicate that WUSB are slower by almost 60% than wired USB.
- *Transmission range with different power levels:*The test started by putting the Host dongle and the Device dongle on the same line of sight; then the device dongle is moved away until it is disconnected. The same procedure was repeated while using two Device dongles. The two dongles are placed at the same horizontal level with a separation of few centimeters, and are moved together. The whole experiment was repeated while changing the dongles transmission power level (i.e.,

low, normal and strong).

As shown in Table III, the existence of two dongles at the same area, and working at the same channel has a dramatic effect on transmission range, so when considering that the distances between seats inside the cabin is considerably short when compared with the minimum transmission range, then it is highly recommended to use different channels for neighboring dongles.

Transmission power	**Max achieved distance in meters between Tx and Rxs**	
	Single device	**Dual device**
Low	7	4.2
Normal	12	6.3
Strong	16	8.4

Table III
TRANSMISSION RANGE

4) Smart Antennas: Smart antennas can be used for node localization in WSN networks. Zhuhong [29] mentioned two methods for determining node's position, the range-based, and range-free methods. The first depends on the distance and angle information, while the later depends on estimating the location through the information of transmitted packets. According to this categorization, we will consider the range-based approach to provide our proposed protocol with the information necessary to allow each VDU to determine the position of its own PCU.

The smart antenna's location can be an issue for many arguments. One opinion is to fix the antenna in the seat's arm and to be directed towards the VDU, so the PCU will only act as a keyboard. Although this is an appealing solution, but it decreases the easiness of installation and reconfiguration of seats, and it may require physical changes to the seat arm's design. In addition, any changes in the position of the front seat's back, or the seat's arm itself (which can change its orientation in some types of seats) can affect the connection. For these reasons we propose to locate the antenna in the PCU itself. Our proposed protocol can provide a mechanism to determine the PCU's position; which can be determined by the proposed protocol as shown in the next sections.

IV. DESIGN OF THE PROPOSED PROTOCOL

For every VDU in the IFE system, there is a dedicated PCU to allow the passenger to choose his selections. Thus, each VDU is surrounded by different number of PCUs. Selecting the appropriate comrade is not an easy task especially if we considered that the PCUs are neither predefined nor pre-assigned for any VDU. Nevertheless, using non-configured PCUs makes the system more maintainable with respect to device failure where any failing device can be replaced instantaneously, and automatically recognized by the system. Accordingly, each VDU has to find its own PCU.

Device identification and device localization are usually treated in the literature as separate problems and are usually addressing outdoor situations. Radio frequency (RF) fingerprinting techniques [30]–[32] were used to identify wireless devices specially for security reasons. On the other hand, smart antennas are usually used for localization purposes [33], [34]. We did not find references of work done with the same assumptions, considering that our proposed protocol utilizes the localization capabilities of smart antennas to introduce a device identification technique.

The smart antenna technology can provide a significant help in such environment. First, it can overcome the drawbacks of some physical hindrances such as interference, and multipath fading. Second, it can provide the system with the location information between each transmitter and receiver in terms of distance and angle.

This information can be used in the coupling process between VDUs and PCUs; when a VDU is able to know the location information of the surrounding PCUs, it will be possible for it to select the required partner. However, such process needs a selection mechanism able to differentiate between the targeted and the unconcerned neighboring devices. Accordingly, the proposed protocol can use this information to allow the VDU to select its PCU without being confused by the large number of surrounding devices. The protocol is able to sense all the devices within range, identify the required device, and finally select it. Moreover, it is able to detect if the required device is out of service or not.

A. General requirements

Depending on the seats layout, each VDU is surrounded by one or more PCUs. When the system is started, these PCUs are not assigned to any VDU, so It is the task of each VDU to find its own PCU. The following problems may occur:

- A situation may exist where more than one PCU exist in the range of the same VDU. In this case, the protocol should be able to use the provided location information (i.e., angle and distance) to determine the suitable PCU.
- When the link between a VDU and its PCU is broken, the protocol must be able to detect the situation and determine if it is due to a PCU failure or because the user had moved it out of the VDU's range.
- When a failing unit is replaced (either a VDU or a PCU), it must be self configured to take its role in the network

Figure 11 shows a normal seat configuration where each VDU is fixed in its own seat and surrounded by different PCUs. The protocol has three phases, configuration phase, normal operation, and re-configuration phase.

- *Configuration Phase:* This phase occurs during the system's startup. It is responsible for determining the network's topology. Each VDU checks the availability of its PCU and responds with its status.
- *Normal Operation:* In this phase, the protocol must be aware of the availability of its assigned PCU.

- *Re-configuration phase:* It occurs when a VDU fails to connect to its PCU or vice versa. After the failing unit had been replaced or re-operated, it should be able to join the network automatically.

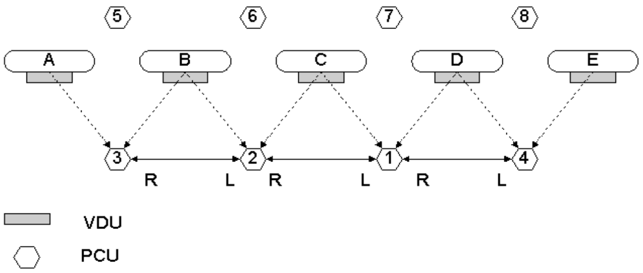


Figure 11. VDUs and PCUs distribution

B. Specifications

The protocol should be able to allow each VDU to find its own PCU and provide their connection status. In other words, it is not the protocol's responsibility to transfer data between nodes. Transferring data like audio or video streams can be accomplished by other protocols (i.e., TCP/IP).

The protocol should provide the running applications with information required to take certain actions (i.e., warnings due to a failing PCU). The following is a list of the proposed services:

- *Multiple PCUs awareness:* The protocol should be able to detect multiple PCUs that may exist in the VDU's range and select the appropriate one.
- *ID assignment:* The protocol should automatically assign a unique ID to both of the PCU and the VDU so they can communicate with each other.
- *Failure reports:* A failing VDU or PCU should be detected and reported.
- *Self adaptation:* After replacing a failing device, it must be able to join the network automatically.
- *PCU out of range:* when a user moves or directs the PCU away of the VDU, the protocol should be able to identify this situation and differentiate between being out of range and out of service.

C. Functionality and studied use cases

When the system is started, the *Configuration Phase* is initiated. In all scenarios, the VDU broadcasts a *QRY_search* request and waits for replies within a predetermined time interval to prevent indefinite wait states. The next step is to use the angle information to exclude the PCU(s) behind it (since it is only interested in the PCUs at its front side) and starts to handle the other PCU(s) of valid replies (i.e., seat 'D' in Figure 12). All functioning PCUs, which are not assigned to another VDU will respond to the request. The correct PCU should be located at the nearest distance on the right side of the VDU, so the selection procedure looks for the responding PCU, which has the least angle with the vertical "Y" axis, and the shortest distance. However, there may be a case where two PCUs are too close to each other to the extent that the VDU can't accurately determine the differences between their angle and distance. In this case the VDU asks them to start negotiation between each other. These situations are presented in Figure 12 and Figure 13 to present the following scenarios:

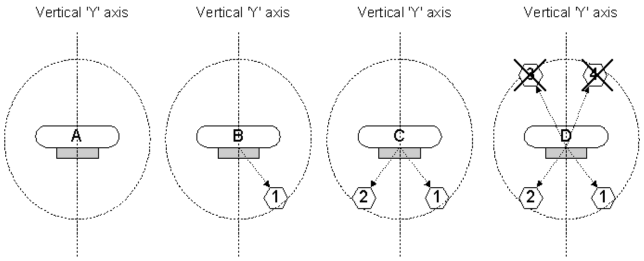


Figure 12. Different scenarios for less than three valid PCUs

1) *No PCU(s):* When The VDU doesn't receive a reply for its search request, it raises an error to indicate that no PCU(s) are within its range, and enters a search state until a PCU is found. (i.e., seat 'A').
2) *Best case:* only one valid PCU is located in its correct position within the VDU's range: The VDU sends a *QRY_join* request and the PCU replies with a *QRY_accept* to confirm the assignment (i.e., seat 'B').
3) *Two PCUs:* If the VDU received 2 valid replies within the time limit, then this indicates the presence of two PCUs within the range (i.e., seat 'C'). The PCU with the smallest angle with respect to the 'Y' axis is selected. If two PCUs are too close for the system to differentiate the difference in angle, then the PCU with the shortest distance is selected. If the difference in distance can't be determined, then the VDU sends a *QRY_negotiate* request to authorize the PCUs to elect one of them. The negotiation result is returned to the VDU to know its elected PCU.
4) *The worst case is the existence of more than two PCUs:* If the VDU received more than two valid replies, then it starts to sort them in ascending order firstly according to their angle to the 'Y' axis , secondly according to their distance. It is expected that the required PCU has the smallest angle and the shortest distance on the right of the 'Y' axis. There are different scenarios for this situation (see Figure 13). Table IV shows how each situation can be handled.
   - Seat 'E': PCU1 was selected because it has the smallest angle on the right side of the 'Y' axis.
   - Seat 'F': PCUs 1&4 are firstly selected since they are at the right side. However, they have equal angles, so their distance is checked.. Finally, PCU1 is selected because it has a shorter distance.

Seat	Situation	Selection criteria		
		Angle	Distance	Negotiation
E	Small angle	1	-	-
F	Same angle	1&4	1	-
G	Too close(same angle & distance)	1&2	1&2	1

Table IV
SELECTION CRITERIA

- Seat 'G': PCUs 1&2 were selected according to the angle and distance criteria. They are too close to each other to the extent that the VDU can't differentiate between their angles and distances, so the VDU initiates a negotiation session to elect one of them. Finally, PCU1 is selected.

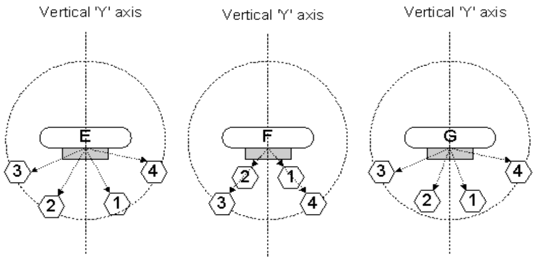


Figure 13. More than two PCUs within range

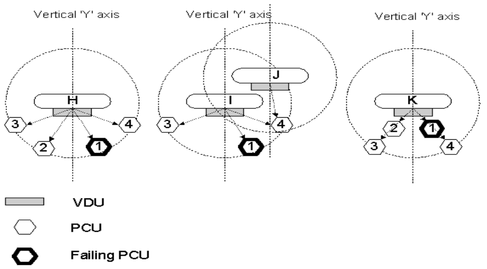


Figure 14. Failing PCUs scenarios

In fact, the real world is not that simple. If faults exist, then there will be exceptions in the above scenarios. For example, if the correct PCU was not functioning, then a wrong PCU can be chosen. This means that a PCU failure may affect its VDU as well as its neighboring VDU(s). To overcome this situation, the angle of the 1st PCU in the left quarter is always considered (i.e., PCU2). For instance, at seat 'H' (see Figure 14), if the angle of the recommended PCU for selection (i.e., PCU4) is greater than the angle of PCU2, this indicates that PCU1 is not working. This is due to the fact that the correct PCU must have the smallest angle and shortest distance to its VDU.

Unfortunately, this scheme doesn't solve the problem of seat 'I' where the angles and distances of PCU3 and PCU4 are equal, so they will enter a negotiation phase that ends up with electing PCU4 (which is not correct). Therefore, it is mandatory for PCUs to wait before starting negotiation to allow the wrong PCU (i.e., PCU4) to be chosen by its appropriate VDU (i.e., seat 'J'). In this case, seat 'I' can raise an error for not finding its PCU.

For seat 'K', PCU4 angle is equal to PCU2 angle, but with a greater distance, so PCU4 is not the correct PCU. In addition, each VDU has to inform all the PCUs in its range that it had found its PCU. On the other hand, a PCU, which knows that all the surrounding VDUs had found their own PCU will understand that its VDU is not functioning.

D. Selection mechanism

Each VDU creates a list of the surrounding PCUs containing their location information. At the start of the selection procedure, the VDU deletes from its PCUs list all the PCUs instances behind it, and then two lists are created, one for PCU(s) on the left hand side, and the other for PCU(s) on the right hand side. The two lists are sorted in an ascending order according to their angles. After sorting, the two lists can be categorized as shown in Table V. The table shows the actions that should be taken according to each state; remember that selection is taken according to angle, distance, and negotiation, respectively.

State	Number of PCUs		Action
	Left zone	Right zone	
1	≥ 0	0	Raise an error
2	0	1	Wait then select the PCU
3	0	>1	Select (according to angle, and distance, or negotiation)
4	≥ 1	≥ 1	Compare → Wait → Select

Table V
DOMAIN OF PCUS OCCURENCIES

- *Angle selection:* For state1, an error is initiated when there are no PCUs in the right zone. For state2, if only one PCU is present in the right zone then it will be selected after waiting for a time interval. The waiting time is important in case that the correct PCU is not functioning where another PCU may be selected. The waiting time gives the other PCU(s) the chance to be selected by its own VDU(s). This will lead to raising an error after the malfunctioning PCU is not detected. For state3, the selection between PCUs in the right zone is performed according to their angle and distance, or finally by negotiation. For state4 where there is at least 1 PCU in each zone, the selection is performed according to their angles where $\theta 1$ and $\theta 2$ represent the angles of PCUs in the right and left zones, respectively. The angle of the first PCU in both zones (with respect to the Y axis) is compared and actions are taken according to Table VI. Note that the absolute value of angles is used in the comparison.

Condition	Action
$\theta1 < \theta2$	$\theta1$ is selected if no other PCU in the right zone has the same angle, other wise a distance selection is performed
$\theta1 > \theta2$	Error is raised
$\theta1 = \theta2$	Selection according to distance is performed

Table VI
ANGLE SELECTION CRITERIA

Seat	Differences in		Action
	r	θ	
L	-ve	Zero	PCU 1 is selected
M	-ve	-ve	PCU 2 is selected
N	Zero	-ve	PCU 1 is selected
P	+ve	Zero	Error is Raised

Table VII
NEGOTIATION ACTIONS

- *Distance selection:* When two or more of the selected PCUs at the right side have the same angle, the PCU with the shortest distance "r" is selected. If two PCUs have the same shortest distance, then a negotiation is started to elect one of them and inform the VDU with the result.

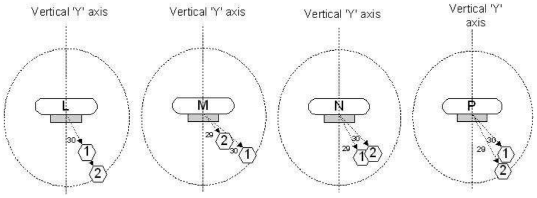


Figure 15. Negotiation cases

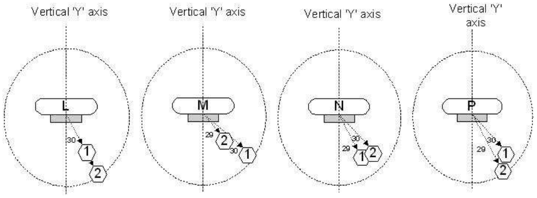


Figure 16. Negotiation cases

- *Negotiation selection*: The negotiation session is shared between the VDU, which initiates the request, and the PCUs that participate in the negotiation. Firstly, the VDU creates a *Participation List* for all of the concerned PCUs, it then sends a negotiation message that includes the list to each of the participants, and waits for their reply. Each PCU receives the message and tries to find its position with respect to the others. Each PCU is already aware of the VDU's position. Figure 16 shows different cases of negotiation and Table VII shows the related actions. For seat "L" PCUs 1&2 are able to communicate with each other and to decide that PCU1 is nearer to the VDU. The same thing happens to seats "M & N". For seat "P", they will notice that PCU2 is the nearest but with larger distance; this may be due to a failing PCU, so an error is raised.

V. PROTOCOL DESIGN AND EVALUATION

Fixing bugs in a protocol is an important and often the highest priority activity. Tracking down bugs, in non predefined protocol specifications, is a challenge to many designers. Checking protocol correctness is often done using verification techniques such as "*Reachability Analysis*" [35], which searches through all reachable states. It is almost impossible to do an exhaustive test, which often requires 100% of the reachable states. Another approach can be used, which is program proof. This requires an automated solution for analyzing and testing the design, so we used TAU version 3.1 [36] to build and verify our UML model. UML language is a formal language ensuring precision, consistency, and clarity in the design that is crucial for mission critical applications. It has a high degree of testability as a result of its formalization for parallelism, interfaces, communication, and time. After identifying the protocol's functionality, NS2 simulator was used to apply more scenarios and show the protocol's performance.

A. The UML model

The informal techniques used to design communication protocols (i.e., timing diagrams) yield a disturbing number of errors or unexpected and undesirable behavior in most protocols, so we are interested in formal techniques, which are being developed to facilitate design of correct protocols. It is accepted that the key to successfully develop a system is to produce a good system specification and design. This task requires a suitable specification language, satisfying the following needs:

- A well designed set of concepts.
- Unambiguous, clear, and precise specifications.
- A thorough and accurate basis for analyzing the specifications.
- A basis for determining whether or not an implementation conforms to the specifications.
- Computer support for generating applications without the need for the traditional coding phase.

UML language has been defined to meet these demands.

Three different layers were modeled, *Upper*, *Protocol*, and *Lower* layers. The *Upper* layer initiates the session by a request to start the search phase and waits for the results; while the *Lower* layer provides the protocol layer with the distance "r" and the angle "θ". The *Protocol* layer provides the necessary functionality that our protocol needs to work correctly. In addition, a model was used to represent the environment and determines the number of PCUs and their locations with respect to the VDU.

1) The model's structure: The protocol's model consists of three main classes; *VDU* and *PCU* classes - to represent the behavior of the VDU, the PCU - and the *Network* class, which determines the scenario parameters. Each scenario consists of a VDU, and a set of PCUs of different locations. The *Network* class is responsible for informing the working instances of the VDU and PCU(s) with their locations.

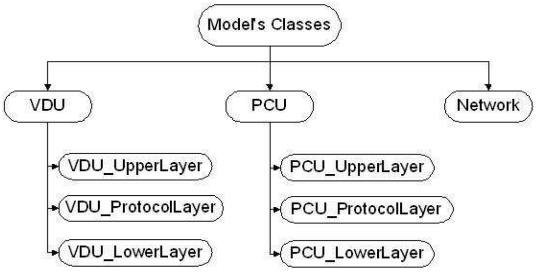


Figure 17. Model structure

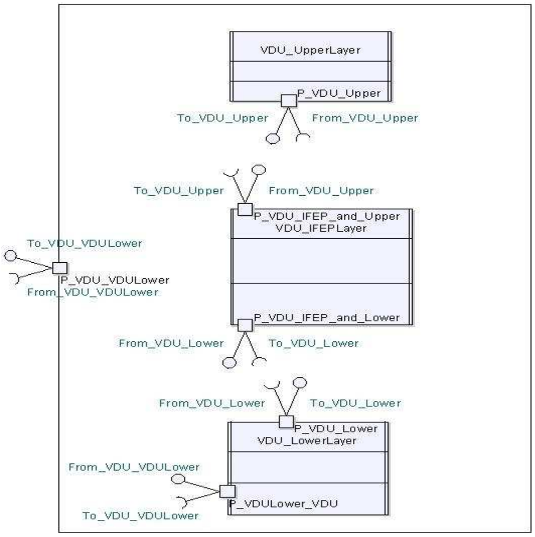


Figure 18. VDU Class

Both of the *VDU* and *PCU* classes consist of three internal classes, the *Upper Layer* class, the *Protocol Layer* class, and the *Lower Layer* class (see Figure 17). The *Protocol Layer* class represents the core of the protocol, while the other two layers are just assistances to provide the needed services. The connection between these layers and the surrounding environment takes place through the main class (i.e., *VDU* class, *PCU* class). Figure 18 represents the VDU class as an example of the UML structures. Each internal class has input and output interfaces to communicate to each other. The lower layer class has interfaces to the containing VDU class to allow it to communicate with external entities.

For example, to start a search request, the request is sent from the *Upper Layer* to the *Protocol Layer* where the correct decision is taken and the required action is determined. Now, the action should be sent to a corresponding instance (i.e., PCU). A signal is sent to the *Lower Layer* then to the containing class, which in turn sends the signal to the corresponding instance. When the corresponding instance receives the signal, the signal reaches the *Protocol Layer* of the instance through the same reversal internal path.

On the other hand, the Network class has a different structure since it is not concerned with the protocol's behavior. It determines the VDU and PCU instances, and provides the working instances with their location information in order to simulate the services provided by the smart antennas

2) The model's behavior: An example for the model behavior is shown in Figure 19. As an initial preparation, the *Network* class sends the location information to the VDU and PCU(s) instances so that each instance knows its own location (signal 1). After the VDU had received its initialization data, its *Upper Layer* sends a search request to its protocol layer (signal 2). The *Protocol Layer* broadcasts this request to the neighboring PCU(s). When the *Protocol Layer* of a PCU instance receives the request, it replies with a signal that shows its presence (signal 3).

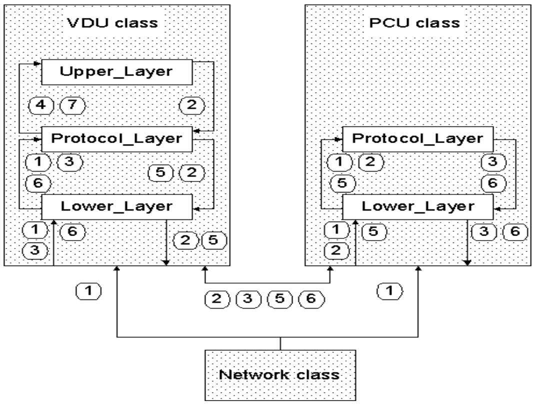


Figure 19. Model's signals

The VDU waits until it receives the replies to count the number of available PCUs. If no PCU had replied, then an error message is sent to the upper layer (signal 4). If one or more PCU had replied, then a selection procedure starts. The result of this selection is used to send a "Join" signal to the selected PCU (signal 5) and waits for its "Reply" signal to confirm its joining (signal 6). The confirmation is sent to the upper layer to inform it with the PCU that belongs to the PCU (signal 7).

B. performance evaluation

Obviously, TAU can provide us with a way to verify the correctness of the protocol through limited scenarios. It is difficult to use it to experiment with complicated scenarios, and determine performance issues. NS2 simulator [27] was used as the next step. It is a part of VINT (Virtual INternet Testbed) project [37]. It is an open source simulator that can be used to evaluate different issues for both wired and wireless networks. In the simulation part we are trying to verify the written code for the NS2 as well as to find out the protocol's points of weakness.

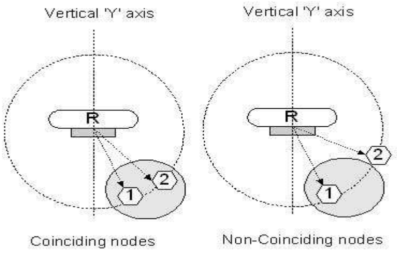


Figure 20. Threshold area

A problem that faced us was the unavailability of a smart antenna module embedded in NS2 because the protocol's behavior is highly dependent on their presence. However, this was not a great issue because NS2 keeps track of the location of each node in the simulation through the class *MobileNode*. This means that the results of the simulation represents the actual performance of the protocol's behavior.

The NS2 simulation is defined by TCL scripts, and C++ codes where the protocol's module was implemented in C++ and linked to the TCL script for further configuration. For example, if we used the provided coordinates we will never be able to start a negotiation session, because the VDU will always see that the PCUs are of different angles and distances. In other words, to implement negotiation scenarios, the VDU must consider the PCUs as if they are coinciding. This was solved by using a *Threshold* variable (changed through the TCL script) through which two PCUs are coinciding if the distance between them is less than the *Threshold* value. The Threshold area is represented by dark circle in Figure 20, which represents two coinciding nodes, when they are located within a circle of radius equal to the *Threshold* value, and are considered non-coinciding if the distance between them is greater than the *Threshold*.

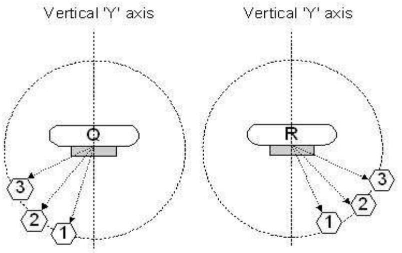


Figure 21. NS2 extra scenarios

In addition to the scenarios mentioned before (i.e., seats "A" to "P"), we implemented two extra scenarios (see Figure 21) Seat "Q" represents an error situation (because there isn't any PCUs in the right area). Seat "R" represents a normal operation. They are almost like the situations of seat "A" and "B" respectively, but we used them just to prove that the existence of multiple PCUs within the same region doesn't affect the correctness of the selection

Source	Message	Meaning
VDU	Search_Request	Starts the search phase
	Search_Join	Accepts its own PCU
	Negotiate	Starts a negotiation session
PCU	*Search_Reply*	A respond to *Search_Request*
	Search_Accept	A respond to *Search_Join*
	Negotiate_Request	Starts negotiation between PCUs
	Negotiate_Accept	Confirms acceptance of *Negotiate_Request*
	Negotiate_Reply	A respond to Negotiate

Table VIII
MESSAGES LIST

Table VIII summarizes the types of messages exchanged between VDUs and PCUs instances. They are categorized according to the initiating device. The message sequence depends on the type of situation if it is a normal operation (Figure 22) or an error situation (Figure 23) or a negotiation operation (Figure 24).

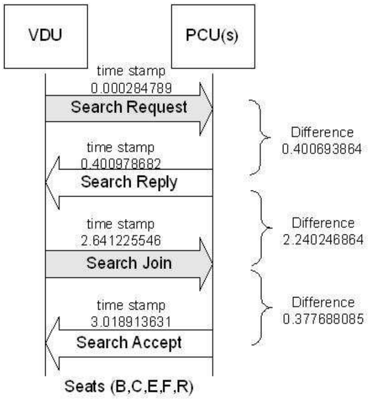


Figure 22. Normal operation

Figure 22, Figure 23, and Figure 24 show timing diagrams for three categories of scenarios, normal operation, error operation, and negotiation operation respectively. Each message is labeled by its transmission time stamp. When it happens that the same type of message is sent from different transmitters we choose the time stamp of the latest one (maximum value). For example, when the VDU broadcasts a *Search_Request* message, it receives a *Search_Reply* message from all the neighboring PCUs. In this case we choose the time stamp of the last received *Search_Reply*. At the right side of the figures, we calculated the time delay between each two successive messages. At the bottom of the figures we indicated the scenarios (i.e., seats), which match each operation.

Figure 22 shows the results of normal operation scenarios where the VDU broadcasts the request and the PCU(s) send(s) their replies. The VDU decides, which PCU is the required one and sends a join request for the chosen one, which in turn replies with its acceptance. It is obvious that the maximum delay in this operation is the wait period, which the VDU uses to wait for all available PCUs to respond. The delay was set to approximately 2secs. The value was chosen to be relatively large to show its impact on the protocol's performance; considering that the processing time of the requests is trivial when compared to the wait time.

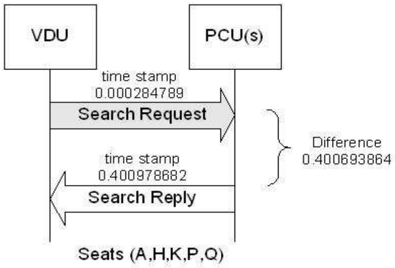


Figure 23. Error operation

Figure 23 shows the fastest operation, which took place when the required PCU is not detected. After waiting for the delay period (i.e., 2secs) through which it receives all the *Search_Reply* messages (if any). The VDU raises an internal error to show the failure of finding the PCU.

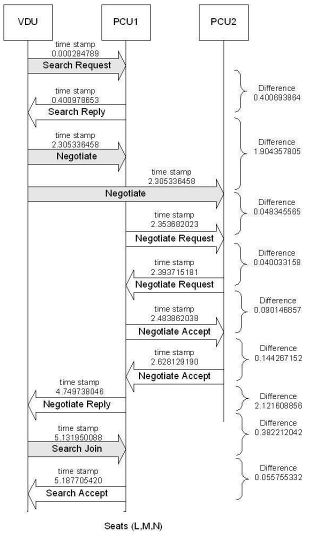


Figure 24. Negotiation operation

Figure 24 shows the most time consuming operation, which takes place during negotiation between PCUs to elect one of them. The first part is the same as the start of a normal operation, but when the VDU fails to distinguish the location difference between two PCUs where one of them is probably the required one, it asks them to start negotiation and elect one of them. The most time consuming parts are the waiting periods (mentioned above), and the negotiation process between the PCUs. Each of them is about 2 s.

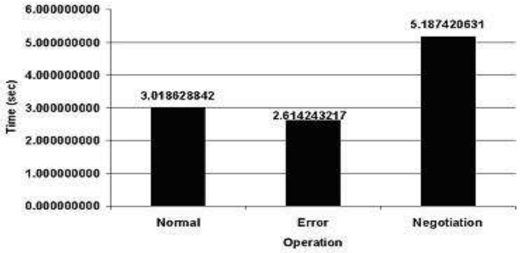


Figure 25. Convergence time

Figure 25 shows a comparison for the convergence time of each operation. It indicates that the negotiation operation is the slowest one, while the difference between a normal operation and an exception (error) is not large. However, the delay of the slowest case is still acceptable during the system's startup. On the other hand, no significant comparison can be made to previous work since the wireless cabin environment is still under investigation.

VI. Conclusion and future work

Heterogeneous network architecture is a promising solution for such application. Using PLC networks can be a competitive solution since it decreases the amount of cabling inside the cabin, and can be used to connect the APs (to support mobility) directly to the network system. Moreover, it overcomes the interference constrain, and can provide enough bandwidth to support heavy traffic required for multimedia services. When combined with WUSB, it becomes easier for passengers to connect their PEDs.

Through experimentation results and simulations, this work proves that it is possible to build a heterogeneous network, which contains all the mentioned technology; each to solve a certain part of the problem. Multimedia content distribution supported by PLC and Ethernet architecture added to personal communication supported by WUSB and WiFi can provide the IFE system with a satisfactory solution needed for such systems. This can be done without interfering with each other.

Smart antennas can solve or minimize interference problems. However, new wireless technologies like smart antennas require special mechanisms to fully utilize their capabilities. The proposed protocol is designed to use these capabilities to provide the IFE remote control with self-configurable wireless characteristics. Although the protocol's procedures seems complicated, but in fact they are not because it depends on comparing existing information without using excessive messaging. This behavior enhances the convergence time and the protocol's performance.

The UML model and the NS2 simulation proved that the proposed protocol is able to utilize the location information provided by the smart antennas to allow each VDU to detect its own PCU. Moreover, the protocol considered the probable failure situations, and was able to detect and handle them. However, the protocol point of weakness is its internal timer. The simulation results showed that the value of the timer has a great impact on convergence time.

In addition, the usage of an UML model before creating a NS2 simulation had proved to be of great importance to the protocol's design life time. Although designing the UML model seemed to be a time consuming part, but it saved the effort of bug tracking and semantic errors during implementing the NS2 module.

In this phase of the work, we aimed to have a proof of the concept to show the feasibility of our proposed protocol. The next step is to enhance the written code by using better data structures to minimize the processing delay and improve the convergence time. In addition, we are aiming at trying simulations that represent a real cabin configuration, and inject scenarios with randomly failing devices. Furthermore, we are looking forward to implement a real test-bed to experiment the performance of our protocol in a real environment.

REFERENCES

[1] Akl A., Gayraud T., and Berthou P., *"Investigating Several Wireless Technologies to Build a Heteregeneous Network for the In-Flight Entertainment System Inside an Aircraft Cabin"*, The Sixth International Conference on Wireless and Mobile Communications (ICWMC), pp 532-537, 2010.

[2] Niebla C.P., *"Coverage and capacity planning for aircraft in-cabin wireless heterogeneous networks"*, IEEE Vehicular Technology Conference, pp 1658-1662, 2003.

[3] Jahn A. and Holzbock M., *"Evolution of aeronautical communications for personal and multimedia services"*, IEEE Communications Magazine, vol 41, pp 36-43, 2003.

[4] Thales IFE site: http://www.thales-ifs.com/, last visited: 13 July 2011.

[5] AeroMobile site: http://www.aeromobile.net/, last visited: 13 July 2011.

[6] FlyNet site: http://konzern.lufthansa.com/en/themen/net.html, last visited: 13 July 2011.

[7] Folden D., Jackson T., Panique M., Tiensvold R., Wolff R.S., Howard T., Julian E., Junkert L., Lopez D., and Oudshoorn M.J., *"An Aircraft Cabin Wireless System for Games and Video Entertainment"*, ACM Computer in Entertainment, pp 1-17, 2007.

[8] Liu H., *"In-Flight Entertainment System: State of the Art and Research Directions"*, Second International Workshop on Semantic Media Adaptation and Personalization (SMAP 2007), pp 241-244, 2007.

[9] Alamdari F., *"Airline in-flight entertainment: the passengers' perspective"*, Journal of Air Transport Management, vol 5, pp 203-209, 1999

[10] Hrasnica H., Haidine A., and Lehnert R., *"Broadband Powerline Communications Networks"*, John Wiley & Sons Ltd., 2004.

[11] Mcauley A. J. and Manousakis K., *"SELF-CONFIGURING NETWORKS"*, 21st Century Military Communications Conference Proceedings, vol 1, pp 315-319, 2000.

[12] Castagne S., Curran R., and Collopy P., *"Implementation of value-driven optimisation for the design of aircraft fuselage panels"*, Proceedings of International Journal of Production Economics, pp 381-388, 2009.

[13] Hurley W.C. and Cooke T.L., *"Bend-insensitive Multimode Fibers Enable Advances Cable Performance"*, Proceedings of the 58th International Wire and Cable Symposium (IWCS), pp 458-467, 2009.

[14] Holzbock M., Hu Y., Jahn A., and Werner M., *"Advances of aeronautical communications in the EU framework"*, International Journal of Satellite Communications and Networking, vol 22, pp 113-137, 2004.

[15] Diaz N.R. and Esquitino J.E.J., *"Wideband Channel Characterization for Wireless Communications inside a short haul aircraft"*, Vehicular Technology Conference, pp 223-238, 2004.

[16] Youssef M., Vahala L., and Beggs J.H., *"Wireless network simulation in aircraft cabins"*, IEEE Antennas and Propagation Society Symposium, vol 3, pp 2223-2226, 2004.

[17] Moraitis N., Constantinou P., Fernando Perez F., and Valtr P., *"Propagation Measurements and Comparison with EM Techniques for In-Cabin Wireless Networks"*, Journal EURASIP Journal on Wireless Communications and Networking - Special issue on advances in propagation modelling for wireless systems, pp 1-13, 2009.

[18] Chiu S. and Michelson D.G., *"Effect of Human Presence on UWB Radiowave Propagation Within the Passenger Cabin of a Midsize Airliner"*, IEEE Transactions on Antennas and Propagation, vol 58, pp 917-926, 2010.

[19] Thompson H., *"Wireless and Internet communications technologies for monitoring and control"*, Control Engineering Practice, vol 12, pp 781-791, 2004.

[20] Lansford J., Stephens A., and Nevo R., *"Wi-Fi (802.11b) and Bluetooth: Enabling Coexistence"*, IEEE Communications Magazine, pp 20-27, 2001.

[21] Leavitt N., *"For Wireless USB, the Future Starts Now"*, IEEE Computer Society, vol 40, pp 14-16, 2007.

[22] Udar N., Kant K., Viswanathan R., and Cheung D., *"Characterization of Ultra Wide Band Communications in Data Center Environments"*, Proceedings of ICUWB, pp 322-328, 2007.

[23] Sohn J.M., Baek S.H., and Huh J.D., *"Design issues towards a high performance wireless USB device"*, IEEE International Conference on Ultra-Wideband, vol 3, pp 109-112, 2008.

[24] Winters Jack H., *“Smart Antenna Techniques and Their Application to Wireless Ad-hoc Networks“*, IEEE Wireless Communications Journal, pp 77-83, 2006.

[25] Okamoto Garret T., *”Smart antenna systems and wireless LANs“*, Kluwer Academic Publishers, pp 225, 2002.

[26] MGEN site: http://pf.itd.nrl.navy.mil/mgen/mgen.html, last visited: 13 July 2011.

[27] NS2 site: http://www.isi.edu/nsnam/ns/, last visited: 13 July 2011.

[28] Kaouris A., Zaras M., Revithi M., Moraitis N., and Constantinou P., *”Propagation Measurements inside a B737 Aircraft for In-Cabin Wireless Networks“*, IEEE Vehicular Technology Conference (2008), pp 2932-2936, 2008.

[29] You Z., Meng Max Q.H., Liang H., Li S., Li Y., Chen W., Zhou Y., Miao S., Jiang K., and Guo Q., *”A Localization Algorithm in Wireless Sensor Networkcs Using a Mobile Beacon Node“*, Proceedings of the International Conference on Information Acquisition, pp 420-426, 2007.

[30] Scanlon P., Kennedy I., and Liu Y., *”Feature Extraction Approaches to RF Fingerprinting for Device Identification in Femtocells“*, Bell Labs Technical Journal, vol 15, pp 141-151, 2010.

[31] Aldasouqi I. and Salameh W., *”Detecting and Localizing Wireless Network Attacks Techniques“*, International Journal of Computer Science and Security (IJCSS), vol 4, pp 82-97, 2010.

[32] Danev B. and Capkun S., *”Transient-based Identification of Wireless Sensor Nodes”*, Proceedings of the International Conference on Information Processing in Sensor Networks, pp 25-36, 2009.

[33] Zhang B. and Yu F., *“LSWD : Localization Scheme for Wireless Sensor Networks using Directional Antenna”*, IEEE Transactions on Consumer Electronics, pp 2208-2216, 2010.

[34] Seshan Srirangarajan S.F.A.S. and Tewfik A.H., *“Implementation of a Directional Beacon-Based Position Location Algorithm in a Signal Processing Framework”*, IEEE Transaction On Wireless Communication, vol 9, pp 1044-1053, 2010.

[35] Lin F.J., Chu P.M., and Liu M.T., *”Protocol Verification using reachability analysis“*, ACM Computer Communication Review, pp 126-135, 1988.

[36] IBM Rational Tau site : http://www-01.ibm.com/software/awdtools/tau/, last visited: 13 July 2011.

[37] VINT project site: http://www.isi.edu/nsnam/vint/, last visited: 13 July 2011.

Exploiting Multimedia Frame Semantics and MAC-layer Enhancements for QoS Provisioning in IEEE 802.11e Congested Networks

Anastasios Politis
Dept. of Applied Informatics
University of Macedonia
Thessaloniki, Greece
anpol@uom.gr

Ioannis Mavridis
Dept. of Applied Informatics
University of Macedonia
Thessaloniki, Greece
mavridis@uom.gr

Athanasios Manitsaris
Dept. of Applied Informatics
University of Macedonia
Thessaloniki, Greece
manits@uom.gr

Abstract—**Wireless Local Area Networks (WLANs) supporting modern streaming multimedia applications constitute a very challenging and rapidly changing field of research. Towards implementing effective multimedia wireless networks, the IEEE has published the "state of the art" IEEE 802.11e standard, which introduced a QoS-aware MAC-layer along with a series of efficiency enhancements. However, it has been proven inadequate in handling multimedia traffic optimally in periods of congestion. For the efficient support of multimedia applications in high load situations, numerous mechanisms have emerged, most of them focusing on altering the static nature of resource allocation specified in IEEE 802.11e. Nevertheless, traffic characteristics must be taken into consideration in order to achieve the highest gains. In this paper, an application-aware MAC-layer mechanism is developed that exploits multimedia frame semantics and existing MAC-layer enhancements to adequately cope with high congestion situations in IEEE 802.11e infrastructure networks. The proposed algorithm makes use of existing acknowledgment policies and adaptive resource allocation techniques depending on multimedia frame significance. The effectiveness of the algorithm is proven by means of simulations, where its functionality is evaluated and compared with other existing schemes.**

Keywords- WLANs, Multimedia, IEEE 802.11e, QoS, MAC-layer

I. INTRODUCTION

Wireless Local Area Networks (WLANs) have been established as one of the preferred network technologies by the majority of electronic equipment users. At the same time, networked multimedia applications have penetrated the market with a tremendous success. Hence, the combination of multimedia applications and WLANs has been an extremely interesting research topic for the networking scientific community. The ultimate goal is to design WLANs in a way to support efficiently the incorporated multimedia traffic.

In an attempt to address this challenge, the Institute of Electrical and Electronics Engineers (IEEE) has released a series of amendments, improving the functionality of the initial IEEE 802.11 WLAN standard [2]. The majority of these amendments focused on signal modulation techniques, in an attempt to provide high data rates at the physical (PHY) layer (IEEE 802.11a/b/g) [3], [4], [5]. However, it was soon discovered that MAC layer enhancements were also needed in order to efficiently utilize the available bandwidth. Furthermore, since multimedia applications demand certain and strict Quality of Service (QoS) levels, it is required that the IEEE 802.11 MAC layer is capable of traffic differentiation.

To this direction, several new mechanisms demonstrated an increased efficiency in multimedia applications support in WLANs, by providing prioritized access to different traffic flows and/or reducing MAC layer overhead [6], [7], [8]. Yet, the final act to these research efforts for providing multimedia support in WLANs, was the standardization of the IEEE 802.11e amendment by the IEEE Standards Committee [9]. Most of the enhancements provided by this standard are also included in the recently released IEEE 802.11n standard [10].

IEEE 802.11e specified a QoS-aware MAC layer protocol capable of service differentiation together with a series of MAC layer enhancements. According to the specification, a new coordinating function is introduced, namely the Hybrid Coordination Function (HCF). Two access methods are defined under HCF: the Enhanced Distributed Channel Access (EDCA) and the HCF Controlled Channel Access (HCCA). EDCA provides service differentiation and thus prioritized access to the wireless medium while HCCA is an enhanced version of legacy PCF (Point Coordination Function) with improved QoS features. Unfortunately, both HCCA and PCF mechanisms are rarely implemented in wireless networking products [11]. Therefore, our main concern focuses on the EDCA distributed channel access method.

Older and recent research studies revealed that EDCA functionality lacks adequate multimedia support in high load conditions in wireless infrastructure networks [1], [12]. This outcome is produced by a very common and critical issue present in these topologies, namely the downlink/uplink asymmetry problem. This phenomenon refers to the fact that, in general, downlink traffic (traffic destined to wireless stations) is, typically, considerably larger than the traffic destined to the wired network. In turn, the Access Point (called QAP in IEEE 802.11e terminology) becomes overcrowded and highly congested suffering from large queuing delays, buffer overflows and low throughput [1]. This has an immediate effect on QoS levels of the downlink multimedia flows. This phenomenon is mainly due to the static nature of resource allocation defined by the IEEE 802.11e standard.

In order to alleviate this problem, numerous solutions exist in the scientific bibliography, focusing on altering the static assignment of network resources to multimedia flows at the IEEE 802.11e MAC layer. However, as also noted in [13], such a layered approach to the QoS issue in multimedia networks leads to a simple and independent implementation, often achieving a suboptimal multimedia performance. The solution is the use of cross-layer techniques in order to achieve the highest possible efficiency. Roughly speaking, cross-layer design refers to protocol design by actively exploiting the dependence between protocol layers to obtain performance gains [14].

In this paper, following the work presented in [1], we confront the QoS degradation issue in congested IEEE 802.11e infrastructure networks by designing an application-aware MAC-layer mechanism, which exploits application level information in order to select the appropriate handling of the multimedia traffic at the MAC layer. The proposed mechanism is centralized and placed at the most congested node in the network (QAP) and its effectiveness is proven by means of simulation.

The rest of the paper is organized as follows: in Section II, a thorough overview of the EDCA and the acknowledgment policies defined in IEEE 802.11e is provided. In Section III, a performance comparison between the new acknowledgment schemes and the standard positive acknowledgment mechanism is given. This comparative study will aid the analysis and explanation of the proposed mechanism. Section IV identifies the primary reason for congestion in infrastructure WLANs. An overview of multimedia traffic characteristics and the quality metrics for voice and video applications is provided in Section V. Related work is outlined in Section VI, and our proposed semantic-aware MAC-layer mechanism is described in Section VII. Simulations results are discussed in Section VIII. The paper is concluded in Section IX with the final remarks.

II. Overview of the EDCA Access Method

The EDCA mechanism defined by IEEE 802.11e is a modified DCF scheme designed to provide differentiated and distributed channel access. The service differentiation is distinguished between 8 different User Priorities (UPs), from 0 to 7, with 7 having the highest priority. Each frame from the higher layer arrives at MAC layer with a specific UP which is marked, afterwards, to its MAC header. An 802.11e STA (called QSTA), shall implement four Access Categories (ACs), from 0 to 3, with 3 having the highest priority. Hence a QSTA has four MAC queues, where each queue corresponds to an AC. Each AC is an enhanced variant of DCF and each frame is mapped to an AC according to its UP value as shown in Table I. The relative prioritization is described in the IEEE 802.1D specification [15].

The key feature of EDCA is that for each AC a different set of MAC parameters are assigned in order to achieve service differentiation. An AC uses *AIFS[AC]*, *CW_{min}[AC]* and *CW_{max}[AC]* instead of *DIFS, CW_{min}* and *CW_{max}* defined by legacy DCF. *AIFS* is the new Arbitration Inter-Frame Space introduced by IEEE 802.11e and is given by:

TABLE I. UP TO AC MAPPING

UP	802.1D Traffic Type (Acronym)	AC (AC Number)	IEEE 802.11e Designation
1	Background (BK)	AC_BK (0)	Background (BK)
2	Spare (-)	AC_BK (0)	Background (BK)
0	Best Effort (BE)	AC_BE (1)	Best Effort (BE)
3	Excellent Effort (EE)	AC_BE (1)	Best Effort (BE)
4	Controlled Load (CL)	AC_VI (2)	Video (VI)
5	Video (VI)	AC_VI (2)	Video (VI)
6	Voice (VO)	AC_VO (3)	Voice (VO)
7	Network Control (NC)	AC_VO (3)	Voice (VO)

$$AIFS[AC] = AIFSN[AC] \times T_{slot} + T_{SIFS} \qquad (1)$$

AIFSN[AC] is a positive integer. The lower the value of *AIFS[AC]* the greater the priority for the contenting AC. Moreover, the value of the backoff timer for each AC is chosen randomly with a uniform distribution in the range [0,*CW[AC]*]. This gives the flexibility to ACs with higher priority to select a smaller contention window. In the case of simultaneous expiration of the backoff timers of two or more ACs belonging to the same QSTA, a virtual collision handler is responsible to grand access to the AC with the highest priority.

Another feature introduced by 802.11e is the concept of Transmission Opportunity (TXOP). This is defined as the interval of time in which a QSTA, after winning contention, has the right to initiate multiple frame transmissions, as long as the total transmission time does not exceed a limit called TXOP limit. This procedure is called Contention Free Burst (CFB) and is optional for a QSTA to utilize it. There is a set of default values specified by the IEEE 802.11e standard for the TXOP limit under the EDCA access mechanism. These values depend on the AC type and on the underlying physical layer and are depicted in Table II. The table reveals the prioritized access given on multimedia applications (video and voice) which use smaller values of *AIFSN*, *CWmin* and *CWmax*. Furthermore, the TXOP limit values provide multimedia applications with the CFB feature whereas background and best effort traffic are allowed to transmit single data frames before they re-enter the contention phase.

TABLE II. EDCA RELATED DEFAULT PARAMETER VALUES

AC	CW_{min}	CW_{max}	AIFSN	TXOP limit (µs)	
				802.11 *802.11b*	*802.11a* *802.11g*
BK	CW_{min}	CW_{max}	7	0	0
BE	CW_{min}	CW_{max}	3	0	0
VI	$(CW_{min}+1)/2–1$	CW_{min}	2	6016	3008
VO	$(CW_{min}+1)/4–1$	$(CW_{min}+1)/2–1$	2	3264	1504

By allowing multiple frame transmissions after winning a contention, CFB reduces the number of backoff periods and the number of RTS/CTS frames exchanged, thus resulting in lower overhead. A CFB transmission chronicle is depicted in Fig. 1.

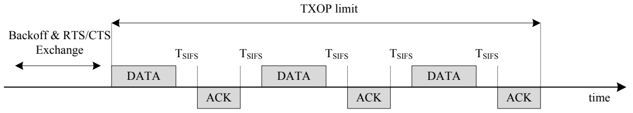


Figure 1. CFB timing structure.

It is a straightforward conclusion that assigning large values to TXOP limit will allow higher throughput and lower delays to a specific AC. As Table II depicts, the default TXOP limit values are statically assigned. Nonetheless, the IEEE 802.11e standard permits dynamic allocation of TXOP limit values.

Towards reducing the MAC layer overhead even further, IEEE 802.11e introduced two new acknowledgment policies besides the default DATA-ACK handshake. The standard acknowledgment policy has the disadvantage that a QSTA has to wait a significant amount of time before continuing with the transmission of the rest of its buffered frames. As depicted in Fig.1 each frame is required to be individually acknowledged before the QSTA may proceed with the next frame in its sequence. Hence, the actual subsequent data frame transmission commences not before the passing of $2\cdot T_{SIFS}+T_{ACK}$ period of time. Accumulating all these waiting periods, the final amount of time dedicated to the exchange of control frames (ACK) may occupy a significant portion of the available TXOP limit assigned to the QSTA, depending on the ACK and data frame sizes as well as the SIFS period (which is PHY dependent). Suppose that a station has n data frames buffered and gained control of the channel. If their transmission times do not exceed the TXOP limit, then the total CFB transmission time can be expressed as follows:

$$T_{CFB}^{std} = \sum_{i=1}^{n} T_{DATA_i} + (2n-1)T_{SIFS} + nT_{ACK} \qquad (2)$$

In order for a QSTA to fully utilize the available TXOP limit, the Block Acknowledgment (BA) and the No Acknowledgment (NoACK) policies are defined under the IEEE 802.11e standard. These acknowledgment policies combined with the CFB feature may drastically improve channel utilization and MAC efficiency.

A. Block Acknowledgment Policy

The Block Acknowledgment scheme improves the MAC layer efficiency by aggregating multiple acknowledgments into a single frame. In this way, the control overhead imposed by the standard acknowledgment policy is reduced. The use of the BA mechanism is optional and is a subject of negotiation between the sender and receiver.

After gaining control of the channel, the sender may request the usage of the BA policy by transmitting an ADDBA (Add Block Acknowledgment) request frame to the receiver, who must acknowledge its reception. The receiver may accept or reject the proposal by issuing an ADDBA response frame. After acknowledging the response, the sender may proceed with a different acknowledgment policy if a rejection was indicated. In the case of a successful agreement between the sender and receiver regarding the usage of the BA policy, the sender will proceed with the transmission of its buffered frames in a CFB manner, without violating the assigned TXOP limit. Upon reception, the receiver shall not produce acknowledgment frames, until the reception of a Block ACK request (BAR) frame indicating the ending of the frame burst and the request of an aggregated acknowledgment frame by the sender. Afterwards, the receiver initiates the transmission of a Block ACK frame (BA) destined to the sender, indicating which frames were received correctly. At this moment, the receiver has two options: initiate an immediate Block ACK or a delayed Block ACK. The former option is suitable for low latency applications, while the latter is used by applications that tolerate moderate latency.

If the Block ACK frame indicates unacknowledged data frames, the sender shall retransmit the lost frames in this or a later TXOP. Otherwise, the BA mechanism is terminated from the sender with a DELBA (Delete Block Acknowledgment) frame which must be acknowledged by the receiver. Moreover, after a timeout of inactivity, the BA agreement may be torn down automatically.

According to IEEE 802.11e, the use of the BA policy is permitted if the following conditions are satisfied:

- A protective mechanism is used (such as HCCA or RTS/CTS) in order to reduce the possibility of other stations transmitting during the TXOP. If no protective mechanism is used, then the first frame of the burst should be acknowledged individually to help the other stations to update their Network Allocation Vectors accordingly.
- The sender may not transmit more frames than the receiver has indicated to be able to buffer.
- All frame transmissions are limited by the TXOP. However, the sender may split frames using this mechanism across several TXOPs.

Assuming that, the receiver has successfully accepted the BA scheme, the Block ACK is immediate, the RTS/CTS protection mechanism is used and all frame exchanges are within the TXOP limit, then the timing of the BA procedure can be modeled as in Fig. 2.

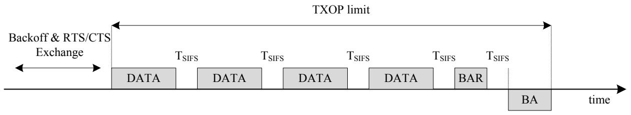


Figure 2. CFB timing structure with BA policy.

If a station transmits n data frames using the BA mechanism, the total CFB time is given by:

$$T_{CFB}^{BA} = T_{BAR} + T_{BA} + \sum_{i=1}^{n} T_{DATA_i} + (n+1)T_{SIFS} \qquad (3)$$

The overhead of individually acknowledging each frame in the sequence is replaced by the BAR and BA frames exchange after concluding the data frames transmission. Furthermore, for a large number of data frames the SIFS periods are almost halved compared to the standard acknowledgment policy.

B. No Acknowledgment Policy

The concept of NoACK policy is fairly simple: for every data frame received, the receiver does not produce an acknowledgment packet, thus the overhead imposed by the acknowledgment frames is completely eliminated. The benefit of exploiting the NoACK policy on packet delay is straightforward. However, in this way the MAC-level recovery mechanism is suppressed and the reliability of the traffic is reduced due to the probability of lost frames from interference, collisions or time-varying channel conditions. To cope with that, the IEEE 802.11e standard proposes that a protective mechanism is used (such as HCCA or RTS/CTS) to reduce the probability of another QSTA transmitting during the TXOP. Similarly to the paradigm given in Fig. 2, the total CFB time consumed to transmit n data frames with the NoACK policy may be written as:

$$T_{CFB}^{NoACK} = \sum_{i=1}^{n} T_{DATA_i} + (n-1)T_{SIFS} \tag{4}$$

Eq. 4 shows the great overhead cost reduction but at the expense of reduced reliability. Hence, the NoACK policy resembles a UDP-like behavior at the MAC layer.

III. Comparative Study of IEEE 802.11e Acknowledgment Policies

In order to facilitate the description and analysis of the proposed mechanism in this paper, this section provides a simple comparative study of the Standard (StdACK), BA and NoACK acknowledgment policies. The efficiency improvement of the new acknowledgment schemes defined under IEEE 802.11e is calculated in terms of total CFB transmission time and compared to the CFB transmission time obtained when using the StdACK policy.

More specifically, a comparison of total CFB times is provided containing different number of equally-sized data frames for various payloads, using Equations 2, 3 and 4. In order to accomplish that, we need to specify the values of T_{DATA}, T_{ACK}, T_{SIFS}, T_{BAR} and T_{BA}. T_{SIFS} is a constant time period and its value depends on the underlying physical technology. Every data and control (ACK, BAR and BA) frame is charged with a physical and MAC overhead. The physical overhead, T_{PHY}, is constant and comprised by a PLCP preamble and a PLCP header. The MAC overhead is frame type dependant and consists of the MAC header and the FCS field. Depending on the physical layer used, the physical overhead has different sizes. For example, in 802.11b the overhead is 192 μs (when using the long preamble) while in 802.11g the overhead is reduced to 20 μs. The MAC overhead for frames carrying data depends on whether the transmission is directed in the same Basic Service Set or in the Extended Service Set, to or from Access Points etc.

The physical layer divides data from the MAC layer into a series of symbols for transmission. Each symbol encodes a certain number of bits, L_{SYM}, depending on the transmission rate selected and then it is transmitted at a prescribed symbol rate, $1/T_{SYM}$. Hence generalizing, T_{DATA}, T_{ACK}, T_{BAR} and T_{BA} may be derived from the following equation:

$$T = T_{PHY} + \left\lceil \frac{L_{MPDU} \cdot 8}{L_{SYM}} \right\rceil \cdot T_{SYM} + T_{SIGNAL} \tag{5}$$

L_{MPDU} is the size of the MAC Protocol Data Unit (MPDU) measured in bytes and is composed by the payload, p, (or MAC Service Data Unit – MSDU) and the MAC overhead (L_{MAC}) and can be expressed as L_{MPDU}=L_{MAC}+p. T_{SIGNAL} is an additional time extension for encoding purposes, applicable only to IEEE 802.11g PHYs.

A. Assumptions

In this comparative study the IEEE 802.11g specification was assumed as the underlying physical layer. IEEE 802.11g specifies actually four physical layers defined as Extended Rate Physicals (ERP's) [5]. These layers make use of the DSSS, OFDM or both modulation methods in order to provide IEEE 802.11a data rates in the 2.4 GHz band and backward compatibility with legacy IEEE 802.11b systems. For this study, the so called ERP-OFDM physical layer was assumed which is used when all stations in a BSS are IEEE 802.11g compliant. Table III summarizes the physical characteristics for the ERP-OFDM PHY, operating at 54Mbps.

TABLE III. IEEE 802.11g Physical Characteristics

T_{PHY}	20 μs
T_{SYM}	4 μs
L_{SYM}	216 bits
T_{SIFS}	10 μs
T_{SIGNAL}	6 μs

TABLE IV. MPDU Sizes for Data and Control Frames

Type of Frame	L_{MPDU}=L_{MAC}+p	No. of OFDM Symbols
Data	28+128 Bytes	6
	28+512 Bytes	20
	28+1024 Bytes	40
	28+1500 Bytes	57
ACK	14 Bytes	1
BAR	24 Bytes	1
BA	152 Bytes	6

Regarding the size of the L_{MPDU}, Table IV summarizes the different sizes for data and control frames. Furthermore, the table reveals the number of OFDM symbols for each frame to be transmitted at the rate of 54 Mbps. L_{MAC} for the data frame is always 28 Bytes, as long as the frame is directed to a station belonging at the same BSS [2]. Four values are assumed as payload size in data frames, namely 128, 512, 1024 and 1500 Bytes. All L_{MPDU} sizes for control frames are taken from [2] and [9].

Furthermore, the following series of assumptions are made in order to simplify the comparative study:

- The channel is error-free, meaning that all frames are received correctly.
- There are no collisions present.
- Regarding the BA scheme, the immediate BA mechanism is used.
- No protective mechanism such as RTS/CTS, CTS-to-self or HCCA is used.
- Packets are not fragmented.
- A CFB is comprised of equal-sized data frames.

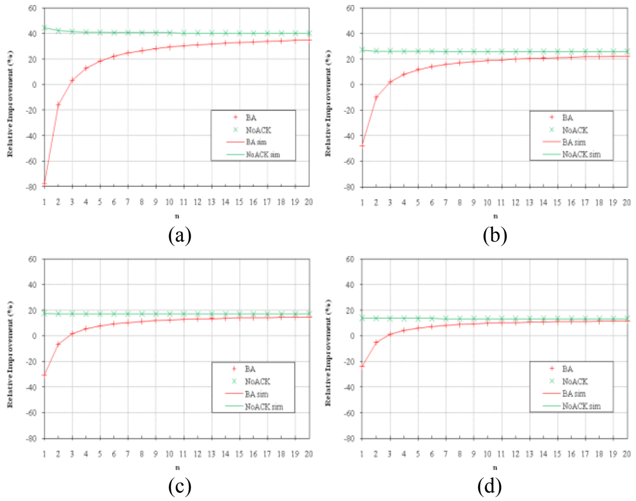
B. Results and Analysis

The relative improvement of the total CFB transmission time was used as a measure for comparing the different acknowledgment policies and is defined as the improvement of CFB transmission time achieved by the BA and NoACK mechanisms relative to the CFB transmission time experienced by the usage of the StdACK policy. This metric was obtained from Equations 2, 3 and 4 which were verified via simulations with the OPNET simulation tool [16]. Fig. 3 displays the relative improvement obtained by increasing the number of equally-sized frames, *n*, in CFB for different payload sizes, *p*. The solid lines are the values obtained by simulations. It must be noted that the default TXOP limit values during simulations were adjusted accordingly, in order to include the different number of frames in the CFB.

A first comment on the displayed outcome may be the observed improvement reduction of both BA and NoACK schemes as payload size increases. This is an expectable finding since larger data frames exhibit large transmission delays and thus the control frames exchange in the StdACK policy occupies a smaller percentage of the total CFB transmission time, thus reducing its margin between the CFB times of BA and NoACK policies.

Another observation from the depicted graphs is the negative improvement achieved by the BA policy on all four cases for $n \leq 2$. This is also an expected result since the BA mechanism uses a large-sized control frame (Block ACK frame) for data acknowledgment. For a small number of data frames this BA frame increases the total CFB transmission time of the BA policy.

As an overall results conclusion, it can be stated that using the BA and NoACK policies for large frames does not provide significant performance improvement, while exploiting these mechanisms for applications with small and constant-sized data frames (such as VoIP applications) leads to a significant lessening of CFB transmission times. This reduction of CFB transmission times of an AC enables the queue to transmit more frames in the remaining portion of the TXOP limit or release the channel sooner for another competing AC to capture it, thus achieving greater intra and inter-AC efficiency.



(a) (b) (c) (d)

Figure 3. Relative Improvement of BA and NoACK policies for payload sizes (a) 128 Bytes, (b) 512 Bytes, (c) 1024 Bytes and (d) 1500 Bytes.

IV. Downlink/Uplink Asymmetry

One very common and critical issue in infrastructure WLANs is the downlink/uplink asymmetry. This refers to the fact that the downlink traffic (traffic transmitted from the Access Point to the QSTAs) is in most cases considerably larger than the uplink traffic (traffic transmitted from QSTAs to the Access Point). An AC in an Access Point (QAP) which serves all downlink traffic receives the same access priority with the AC in a QSTA which serves the uplink traffic. This leads to unfairness problem in which the QAP ACs suffer from large queuing delays, buffer overflows and low throughput [1], [12].

Since QAP accumulates all downlink traffic, there must be a centralized mechanism to allocate dynamically the needed channel resources to ACs in the QAP. A way to allocate these resources is to adapt the TXOP limit of the ACs in the QAP depending on their queue size. This method has been proven to be extremely beneficial in terms of channel efficiency and application performance. However, also noted by [17], little work has been done in the literature regarding VBR video traffic (such as streaming video) which exhibits time varying characteristics.

V. Multimedia Traffic Characteristics

This section summarizes the multimedia traffic characteristics. We focus on VoIP and MPEG streaming video, since they exhibit an increased popularity on both real applications and network related studies. The most frequently used quality metrics of these applications are also described.

A. VoIP

The traditional voice encoder is the G.711, which uses Pulse Code Modulation (PCM) to generate 8 bits samples per 125 μs, and leads to a minimum bandwidth requirement of 64 Kbps for each traffic flow. New voice encoding schemes have been implemented in order to drastically reduce bandwidth reduction, but at the cost of additional coding delay. Popular techniques include the G.729A and G.723.1 codecs. G.729A [18] is one of the most commonly used codecs in VoIP applications, due to its lower bandwidth requirements (8 Kbps) and acceptable complexity. Unless silence compression techniques are used, VoIP codecs typically produce constant bit rate streams with low frame sizes (≤160 Bytes).

The voice applications requirements are stringent. Their demand for assured quality real-time communication restricts the maximum tolerable one-way delay to 100 – 150 ms. Furthermore, the jitter imposed by the network must remain at low values (maximum 50 ms). These strict delay requirements lead to the need of QoS provision by the underlying network.

The most popular performance metric of VoIP applications used in multimedia networking studies is the Mean Opinion Score (MOS) [19]. According to this method, the perception quality of a VoIP call is determined by a single numerical value from 1 to 5, with 1 representing the lowest and 5 the highest quality. Table V presents typical MOS values for implementations of G.711, G.729A, and G.723.1 codecs.

TABLE V. MOS VALUES OF G.711, G.729A AND G.723.1 CODECS

Codec	Data Rate	Frame Size	MOS
G.711	64 Kbps	160 Bytes	4.3
G.729A	8 Kbps	20 Bytes	3.7
G.723.1	5.3 Kbps	20 Bytes	3.62

From Table V, the existence of a trade-off between lowering the required data rate and the perceived quality becomes obvious. Data rate reduction requires higher complexity algorithms which, in turn, produce a lower quality outcome.

B. *Streaming Video*

The main principle of MPEG encoding is inter- and intraframe coding. It distinguishes between three frame types, namely I, P and B-frames. I-frames are completely intra-coded, P-frames are predicted from previous I or P-frames, and B-frames depend on both previous I or P-frames and forward I or P-frames. Frames are arranged in so-called Group of Pictures (GoP). The sequence of frames from a given I-frame up to and including the frame preceding the next I-frame forms one GoP. A GoP pattern is determined by the total number of frames, *N*, comprising it and the number of B-frames, *M*, enclosed by successive P-frames. Thus the notation G_NB_M is used to symbolize the GoP pattern of a video sequence. Typical GoP patterns include: G_6B_2, G_9B_2, $G_{12}B_2$ and $G_{15}B_2$, depending on the required video quality [20]. Fig. 4 depicts a G_9B_2 GoP pattern and the forward and backward references that exist between the frame types.

I-frames contain by far the most information, thus they exhibit the lowest compression ratios. By exploiting temporal redundancies, P-frames achieve higher compression rates than I-frames. Since B-frames are predicted from both previous and following frames they are appointed as the frames with the highest compression ratios.

In contrast with VoIP applications, MPEG video frames are distinguished by their semantics. I-frames are identified as the most significant frame type, since their absence will render a GoP completely undecodable. On the other hand, B-frames are not needed for the decoding of any other frame, thus they are appointed with the lowest significance. P-frames have a variable significance. There is a distinction between the semantics of P-frames, rooting from their relative position in a GoP sequence. Considering the GoP presented in Fig. 4, a possible loss on the first P-frame in the sequence will have a negative chain effect on 89% of the GoP. Similarly, losing the second P-frame will influence the decoding process of 55% of the GoP. The P-frames significance becomes even higher when scalable video is considered, where both I and P-frames are needed to provide a basic video quality [21].

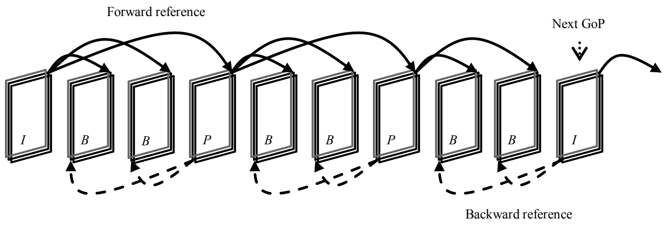


Figure 4. MPEG GoP coding structure.

The standard method for assessing the perceived video quality is to calculate the Peak Signal to Noise Ratio (PSNR) between the original (transmitted) and the received (possibly distorted) image. It is a differential metric which is determined image-wise and yields a quality indicator for each received image of the video sequence. Symbolizing with *f* the original image, with *f′* the distorted image and assuming an $m \times n$ image size, the PSNR is determined as:

$$PSNR = 20\,log_{10}\left(\frac{MAX}{\sqrt{MSE}}\right) \tag{6}$$

$$MSE = \frac{1}{m \times n}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\left[f(i,j) - f'(i,j)\right]^2 \tag{7}$$

MAX is the maximum possible value of a pixel (255 for an 8 bit pixel). *MSE* is the Mean Square Error and calculates the difference between each pixel of the original and distorted picture. Typical values for video compression lies between 30 to 50 dB with higher values preferred over lower ones.

VI. RELATED WORK

There is a growing research literature on semantic-aware QoS provisioning in WLANs supporting multimedia traffic. The most related to our work are briefly reviewed in this section.

In [22], Ksentini et al. propose a QoS cross-layer architecture based on both application and MAC layer features for improving H.264 video transmission over IEEE 802.11e networks. The mechanism relies on a data partitioning technique at the application layer and an appropriate QoS mapping at the IEEE 802.11e MAC layer. More specifically, the authors map the application layer video generated slices to appropriate ACs at the MAC layer according to their significance. AC_VO, AC_VI and AC_BE are used for this purpose while AC_BK is left for serving all other traffic. Furthermore, the retry count parameter at the MAC layer is exploited to unequally protect the high priority information against lower significance frames.

The semantic-aware mechanism presented in [23], follows a similar approach for MPEG-4 video transmission in IEEE 802.11e networks. This scheme introduced a single-video multilevel queue by assigning I-frames to AC_VO, P-frames to AC_VI, B-frames to AC_BE and non-video frames to AC_BK.

In [24], the authors follow a different approach on cross-layer design for H.264 video traffic transmission. At first, they determine the significance of a video frame by using a method called first order estimation. According to this method, the PSNR of all packets in the video sequence is determined by intentionally dropping selected frames. Thus, a frame is more important when its PSNR value is lower. Afterwards, the packets are placed to ACs in the MAC layer according to the access waiting time of an AC and its priority. Hence, the AC with the lowest waiting time is selected for serving a particular video packet.

In [25], Goel and Sarkar propose a mechanism that resides in the interface between LLC and MAC layers to provide QoS for streaming video traffic. The essence of this scheme is to mark I-frames of a video sequence as the Most Valuable Video Packet (MVVP) and en-queue these frames to a higher priority queue called Video Friendly Queue (VFQ) in the interface between LLC and MAC layers. Other frames are en-queued in the so-called Interface Queue (IFQ) and receive FIFO treatment. Whenever frames need to be send to the MAC layer the VFQ receives priority against IFQ. In this way, preferential treatment is provided to MVVP frames ensuring that they get highest priority which minimizes delay.

In [26], a dynamic mapping algorithm of MPEG-4 video frames is proposed. According to this algorithm, the video frames are allocated to ACs according to their significance and network load. When the size of AC_VI reaches a certain threshold, the newly arrived frame is mapped to a lower priority AC (AC_BK or AC_BE). The choice of the AC is determined by the frame significance.

It is clear that the entire semantic-aware mechanisms presented, exploit a mapping technique to allocate video frames to ACs either statically [22], [23], [25] or dynamically [24], [26]. However, with the exception of [26], they disregard the QoS issues of VoIP traffic by allocating voice frames to ACs with lower priority or mixing them with video traffic.

Furthermore, MAC-layer mechanisms, such as dynamic TXOP limit tuning and acknowledgment policies, are left completely unexploited. The usage of MAC-layer strategies may improve the system efficiency, and thus multimedia application performance, dramatically.

VII. The Proposed Scheme

Extending the work presented in [1], we propose a semantic-aware MAC-layer mechanism that falls into the cross-layer mechanisms category. The proposed scheme exploits multimedia frame semantics to decide an appropriate MAC-layer strategy for handling these frames. The mechanism is centralized and intends to improve EDCA performance at the QAP in times of congestion. Furthermore, only the functionality of multimedia ACs is affected, leaving the rest of the ACs (AC_BK and AC_BE) uninfluenced.

The essence of the proposed algorithm is to map multimedia frames into AC_VI or AC_VO according to their significance. To this direction, two categories of multimedia frames are introduced: High Priority Multimedia Frames (HPMF) and Low Priority Multimedia Frames (LPMF). I and P video frames are indicated as high significance frames and tagged as HPMFs, while B-frames and voice packets as LPMFs. Every category is linked to a specific AC: HPMF to AC_VO and LPMF to AC_VI. Such a distinction among the multimedia frames can easily be accomplished by manipulating the UP of the multimedia frame:

```
If UP of packet i $\in$ (4, 5, 6)
    {
        If UP of packet i = 4||5 && packet_type $\in$ (I, P)
            UP of packet i = 6
        ElseIf UP of packet i = 6
            UP of packet i = 4
    }
```

At this point, the appropriate MAC-layer strategy must be selected for both AC_VI and AC_VO. HPMFs belonging to AC_VO, are treated with the maximum protection by using the standard acknowledgment policy. However, regarding the high sizes of these frames, we apply a TXOP limit adaptation algorithm, ensuring the periodical relaxation of this queue.

The TXOP limit adaptation algorithm for the AC_VO traffic class is calculated every Service Interval (SI) which is defined as the time between the start of two subsequent TXOPs. At the beginning of the SI the actual queue length is calculated and the average frame size of all packets contained is determined. Then the TXOP limit is computed as the time needed to successfully transmit the en-queued frames:

$$TXOP\_limit = N \times T_{frame} \tag{8}$$

where N is the number of frames contained in the AC_VO queue at the start of the SI and T_{frame} the successful frame transmission time with the average payload size L. The T_{frame} is computed as:

$$T_{frame} = T_L(r) + T_{ACK}(r) + T_{SIFS} \tag{9}$$

where $T_L(r)$ and $T_{ACK}(r)$ are the transmission times of the data and acknowledgment frames respectively for a specific

PHY data rate *r* and accounting PHY and MAC overhead. T_{SIFS} is the Short Inter-frame Space. Eq. 9 does not contain any contention waiting periods (*AIFS* and backoff time) since at the start of the SI the contention is already won by the AC_VO.

Regarding the AC_VI queue, which holds all the LPMF frames, we propose that no TXOP limit adaptation takes place, in order to keep complexity as low as possible. However, taking into consideration the low significance of B-frames, the loss tolerance of voice frames and the low sizes of the LPMFs (compared to HPMFs), we propose the usage of the NoACK scheme as the acknowledgment policy of this queue. By doing so, we aim at the reduction of the transmission times of LPMFs (as described in Section III) at the cost of an increased loss probability. Nevertheless, in congested networks the usage of this policy is beneficial in retaining medium quality voice calls, as noted in [27]. Furthermore, the loss of B-frames is acceptable, to a certain degree, since their absence will not significantly reduce the video quality.

VIII. Simulations and Results

In order to evaluate the proposed algorithm the OPNET network simulator was used [16]. In this section we provide a description of the simulation scenarios after which the results that were obtained are analyzed and explained.

A. Setup

We considered an infrastructure IEEE 802.11e network with a QAP and four QSTAs in the QBSS. Eight G.729A (20-Bytes frames transmitted every 20ms) encoded VoIP streams (UP=6) were traversing the network: four in the downlink direction and four in the uplink direction. Four MPEG-4 streaming video flows (UP=4) were destined to the QSTAs from the wireline network. Finally, two HTTP connections (UP=0) were representing best effort traffic. The real video trace “Highway”, available from [28], was used as the transmitted video sequence. The trace was encoded in MPEG-4 CIF (Common Intermediate Format) with a GoP pattern G_9B_2 (IBBPBBPBB), 2000 frames, frame rate of 30 frames per second and 67 seconds duration. The video sequence exhibits a mean bit rate of 0.41 Mbps and a peak bit rate of 1.89 Mbps.

The wireless channel was assumed to be error-free, hence no packets were lost due to fading effects. The PHY data rate was set to 11 Mbps. There were three simulation scenarios: the first scenario applies standard EDCA default values as depicted in Table II, the second applies an implementation of a cross-layer mechanism similar to [25] (named Cross), and the third scenario implements the modified version of EDCA according to the proposed algorithm. The Cross implementation allocates I-frames to AC_VO, while all other multimedia frames (P, B video frames and voice frames) are served by the lower priority AC_VI.

All scenarios had a total simulation time of 120 sec. The starting times of each application with respect to the starting time of simulation run are depicted in Table VI.

TABLE VI. Application Timing Characteristics

Application	Start Time (sec)	Stop Time (sec)	Duration (sec)
VoIP	5	105	100
Video Streaming	15	80	65
HTTP	7	End of Simulation	113

As depicted in Table VI, for 65 seconds all applications coexist in the network, creating a highly congested period at the QAP.

To compare the EDCA performance obtained by all three scenarios, four performance metrics were considered: overall network application-level throughput (goodput), overall network application-level end-to-end delay, average PSNR for video streaming and MOS for VoIP applications.

B. Simulation Results and Analysis

Fig. 5 shows the overall network application-level throughput (goodput) in packets/sec for both VoIP and video streaming applications. The video streaming application enjoys a large improvement in throughput performance as show casted in Fig. 5(a). The overall goodput is almost leveled at 120 packets/sec (for all four video streams) while both the standard EDCA and the implemented Cross-layer scheme exhibit a significant performance degradation due to congestion at the QAP. As far as VoIP goodput is concerned (Fig. 5(b)), looking carefully at the graph one can observe that during the presence of the four video flows, VoIP performance for standard EDCA and Cross exhibits large oscillations while the proposed algorithm produces a smoother graph.

Analyzing the overall application level end-to-end delay, Fig. 6(a) and (b) reveal a significant improvement on the packet delay that both multimedia applications receive. Specifically, the VoIP application, that has strict delay requirements, receives better QoS by applying the proposed algorithm. Video packet delay oscillates well below the delay produced by the other schemes as depicted in Fig. 6(a).

Finally, all the benefits of applying the proposed algorithm are revealed from Fig. 7(a) and (b), where the average PSNR of streaming video and the MOS values for VoIP applications are plotted. The majority of PSNR values are well above 35 dB, indicating an excellent quality of the received video streams. Regarding the VoIP applications, both the EDCA and the Cross schemes are outperformed by the proposed algorithm with acceptable MOS values for downstream calls.

As an overall simulation results conclusion, it can be stated that the proposed algorithm clearly produces a significant EDCA performance improvement. The exploitation of the multimedia packet semantics combined with the appropriate MAC-layer enhancements is capable of dealing with high congestion periods in infrastructure IEEE 802.11e WLANs.

IX. CONCLUSIONS

This paper addresses the challenge of transporting multimedia traffic over IEEE 802.11e congested WLANs. The multimedia frame semantics are exploited to select an appropriate MAC-layer strategy. A TXOP limit adaptation scheme is used together with acknowledgment policies in order to relax the ACs in the congested QAP, and at the same time protect high significance multimedia frames. The proposed semantic-aware algorithm is proven extremely beneficial in terms of application level throughput, end-to-end delay and QoS metrics for video and voice applications.

REFERENCES

[1] A. Politis, I. Mavridis and A. Manitsaris, "Enhancing Multimedia Traffic Performance in IEEE 802.11e Networks", *in Proc. of the Sixth International Conference in Wireless and Mobile Communications (ICWMC 2010)*, IEEE Press, Valencia, Spain, pp. 125-130, September 2010.

[2] ISO/IEC and IEEE Std., "Part11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications", 1999.

[3] IEEE Std. 802.11a, "High-Speed Physical Layer in the 5 GHz Band", 1999.

[4] IEEE Std. 802.11b, "Higher-Speed Physical Layer (PHY) Extension in the 2.4 GHz Band", 1999.

[5] IEEE Std. 802.11g, "Further Higher-Speed Physical Layer Extension in the 2.4 GHz Band", 2003.

[6] F. Cali, M. Conti, and E. Gregori, "IEEE 802.11 Protocol: Design and Performance Evaluation of an Adaptive Backoff Mechanism", *IEEE Journal on Selected Areas in Communications,* Vol. 18, No. 3, pp. 1776–1784, 2000.

[7] N. Vaidya, P. Bahl, and S. Gupta, "Distributed Fair Scheduling in a Wireless LAN", in *Proc. Sixth Annual International Conference on Mobile Computing and Networking (MOBICOM)*, Boston, MA, pp. 167-178, 2000.

[8] J. Sobrinho and A. Krishnakumar, "Quality-of-Service in Ad hoc Carrier Sense Multiple Access Networks", *IEEE Journal on Selected Areas in Communications*, Vol. 17, No. 8, pp. 1353–1368, 1999.

[9] IEEE Std. 802.11e, "Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements", 2005.

[10] IEEE Std. 802.11n, "Enhancements for Higher Throughput", 2009.

[11] M. M. Rashid, E. Hossain, V. K. Bhargava, "Controlled Channel Access Scheduling for Guaranteed QoS in 802.11e-Based WLANs", *IEEE Transactions on Wireless Communications*, Vol. 7, No.4, pp. 1287-1297, 2008.

[12] A. Andreadis and R. Zambon, "QoS Enhancement for Multimedia Traffics with Dynamic TXOPlimit in IEEE 802.11e", *in Proc. of the 18th IEEE International Symposium in Personal, Indoor and Mobile Radio Communications (PIMRC 07)*, Athens, Greece, pp. 16-22, 2007.

[13] M. van der Schaar and D. Sai Shankar, "Cross-layer wireless multimedia transmission: Challenges, principles and new paradigms", *IEEE wireless Communications*, Vol. 12, No. 4, pp. 50- 58, 2005.

[14] C. Bouras, A. Gkamas and G. Kioumourtzis, "Challenges in Cross Layer Adaptation for Multimedia Transmission", *in Proc. of the IADIS International Conference Wireless Applications and Computing 2007*, Lisbon, Portugal, pp. 129 – 133, 2007.

[15] IEEE Std 802.1D "IEEE Standard for Local and Metropolitan Area Networks: Media Access Control (MAC) Bridges", Revised version, 2004.

[16] OPNET Technologies Inc., http://www.opnet.com, accessed 13/1/2011.

[17] H. Liu and Y. Zhao, "Adaptive EDCA Algorithm Using Video Prediction for Multimedia IEEE 802.11e WLAN", *in Proc. of the 2nd International Conference on Wireless and Mobile Communications* (ICWMC), Bucharest, Romania, 2006.

[18] ITU-T Annex A to Recommendation G.729, "Coding of Speech at 8 Kbit/s using Conjugate Structure Algebraic-Code-Excited Linear Prediction (CS-CELP)", November 1996.

[19] ITU-T Recommendation P.800, "Methods for subjective determination of transmission quality", 1996.

[20] A. Lazaris, P. Koutsakis and M. Paterakis, "A new model for video traffic originating from multiplexed MPEG-4 videoconference streams", *Performance Evaluation*, Vol. 65, No. 1, pp. 51-70, 2008.

[21] P. Seeling, M. Reisslein and B. Kulapala, "Network Performance Evaluation Using Frame Size and Quality Traces of Single-Layer and Two-Layer Video: A Tutorial", *IEEE Communications Surveys and Tutorials*, Vol.6, No.2, pp. 58-78, Third Quarter 2004.

[22] A. Ksentini, M. Naimi and A. Gueroui, "Toward an improvement of H.264 video transmission over IEEE 802.11e through a cross-layer architecture", *IEEE Commun. Mag.*, Vol. 44, No. 1, pp.107-114, January 2006.

[23] A. M. Jama, S. Issa and O. O. Khalifa, "Performance evaluation of MPEG-4 video transmission over IEEE 802.11e", *IJCNS*, Vol. 2, No. 5, pp. 11-15, May 2010.

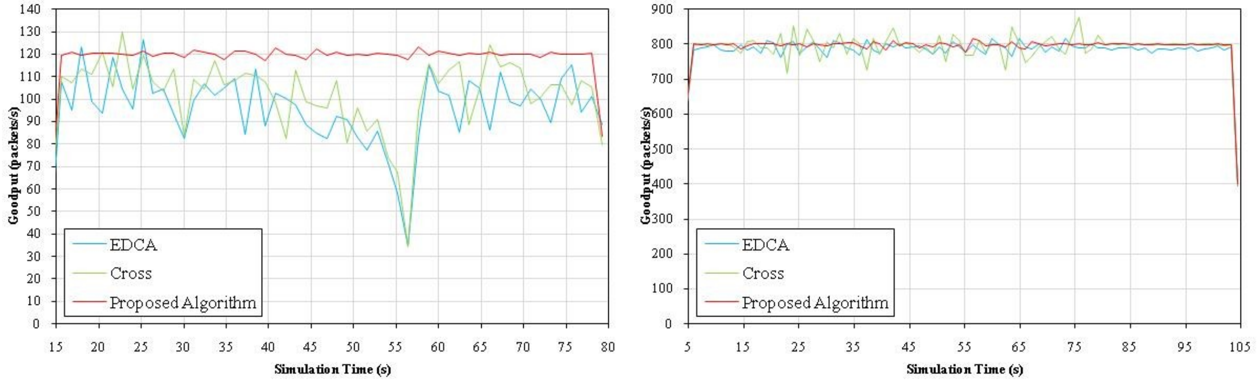
[24] C.-H. Mai, Y.-C. Huang and H.-Y. Wei, "Cross-Layer Adaptive H.264/AVC Streaming over IEEE 802.11e Experimental Testbed", *in Proc. of the 71st IEEE Conference on Vehicular Technology (VTC 2010)*, Taipei, Taiwan, May 2010.

[25] R. Goel and M. Sarkar, "Enhancing QoS of streaming videos over WLANs", *in Proc. of IAENG WCECS '08*, Berkeley, California, 2008.

[26] C.-H. Ke, C.-H. Lin, C.-K. Shieh, N. Chilamkurti and S. Zeadally, "A Novel Cross-Layer Architecture for MPEG-4 Video Stream over IEEE 802.11e Wireless Network", *in Special Issue of International Journal of Telecommunications Systems (SPRINGER)*, Vol. 42, No. 3-4, December 2009.

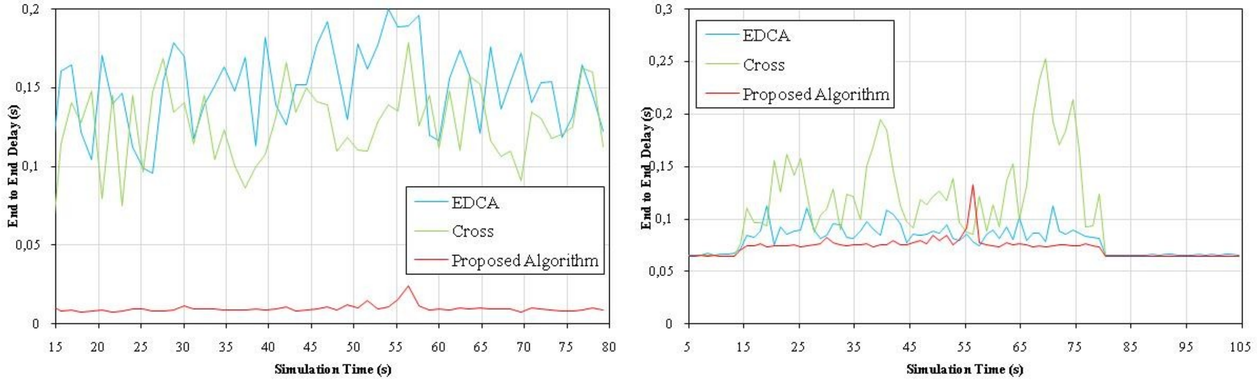
[27] J. Barcelo, B. Bellalta, A. Sfairopoulou, C. Cano, M. Oliver, "No Ack in IEEE 802.11e single-hop ad-hoc VoIP networks", *in Proc. of MED-HOC-NET '08*, Mallorca, Spain, 2008.

[28] Video Traces Research Group, http://trace.eas.asu.edu/, accessed 13/1/2011.



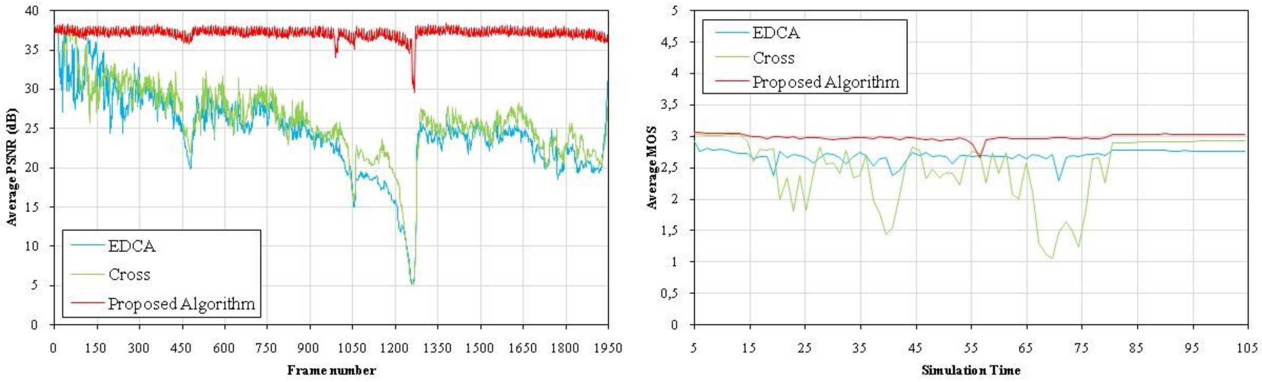
(a) (b)

Figure 5. Overal network application-level throughput (goodput): (a) Video, (b) VoIP.



(a) (b)

Figure 6. Overall network application-level end-to-end delay: (a) Video, (b) VoIP .



(a) (b)

Figure 7. QoS metrics: (a) Average PSNR, (b) Average MOS.

AAODV/AAOMDV Routing Protocols: Single and Multipath Routing in WMNs

Horia Ştefănescu
POLITEHNICA University of Bucharest
Orange Romania
Bucharest, Romania
horia.stefanescu@orange-ftgroup.com

Mariusz Skrocki, Sławomir Kukliński
Orange Labs, Telekomunikacja Polska
Warsaw University of Technology
Warsaw, Poland
{mariusz.skrocki, slawomir.kuklinski}@telekomunikacja.pl

***Abstract*—This paper consists of two parts. In the first part, we propose a new routing protocol, named Adaptive Ad hoc On-Demand Distance Vector (AAODV). It is able to establish routes using any per link calculated routing metric, due to its ability to separate the monitoring of the quality of the paths from the routing mechanisms. By using AAODV, the following widely used metrics: hop count, delay, jitter and Expected Transmission Count (ETX) are compared using ns-3 simulations performed in eight randomly generated topologies with different traffic patterns. The results have shown that in the case of random topologies none of the routing metrics used provides significantly better results than the other one. In the second part of our work, the AAODV functionality is enhanced by adding multipath routing and end-to-end Real-Time Monitoring (RTM) of the paths. The new improved protocol is named Adaptive Ad hoc On-Demand Multipath Distance Vector (AAOMDV). AAOMDV provides multiple paths between source and destination nodes, therefore it is mandatory to implement an algorithm that selects the preferred path and switches the traffic on it when this is expected. Our simulations performed in ns-3 provide an inside look into AAOMDV functionality and prove that AAOMDV is able to enhance network performance when the network load increases.**

Keywords-AODV; AAODV; AAOMDV; multipath; routing metrics; Wireless Mesh Networks

I. INTRODUCTION

The scope of this paper is to analyze and to propose new solutions to overcome some of the routing challenges that appear in Wireless Mesh Networks (WMNs). Typically, the WMNs are based on single path, single metric and single radio. In this paper we will focus on the influence of the new concepts, such as: real-time WMN monitoring, multi-parametric metrics and adaptive path selection in multipath routing.

Our work is composed of two parts. In the first part, we propose a new routing protocol, called AAODV [1], which is based on the well-known Ad hoc On-Demand Distance Vector (AODV) protocol [2]. The main innovation of AAODV is that it can be implemented with any kind of per link calculated routing metric. The routing metrics are used during the routing table building phase in order to select the best path in the network. The impact of AAODV routing metric type on the network performance has been verified by extensive simulations. In order to obtain generic results, eight randomly generated topologies with random background traffic were used in the simulations. In each simulated case, we monitored Packet Loss Ratio (PLR), delay and jitter of the source traffic, and the results were averaged accordingly.

In the second part of this paper, the Adaptive Ad hoc On-Demand Multipath Distance Vector (AAOMDV) is presented. AAOMDV is the AAODV protocol extended by the multipath routing and real-time path monitoring that is used for the selection of data forwarding path. The end-to-end path monitoring functions are realized by the specially designed component of the routing architecture, named Real-Time Monitoring (RTM). The RTM simultaneously monitors PLR, delay and jitter of the path. To monitor the active path, the RTM uses traffic packets. For all other paths, called inactive paths, probe messages, of size similar to the traffic packets, are sent to evaluate their quality. This way, RTM provides the real values of PLR, delay and jitter of the active paths and just an estimate of delay and jitter values for the inactive paths. Note that for the inactive paths, RTM does not evaluate the PLR value.

The behavior of AAOMDV has been verified by simulations. The results have shown that the AAOMDV protocol combined with the algorithms for path selection and switching increases the network performance in terms of PLR, delay and jitter; thus providing a better user experience. In the simulation scenarios, we used random topologies and considered different traffic patterns. The simulations were performed using ns-3 [3]. The AODV protocol was used as a benchmark.

The paper is structured as follows. This section describes the research motivation and introduces the proposed concept. Section II presents the related work. In Section III, the AAODV protocol is described, whereas Section IV shows the results of the AAODV simulation. Section V describes AAOMDV, the algorithm for discovering multiple link disjoint paths and the RTM implementation. Section VI presents in detail the algorithm applied for the best path selection in the multipath case. In Section VII, an approach used for active path switching is described. In Section VIII AAOMDV simulation results are presented. We conclude this paper in Section IX.

II. Related Work

It has been shown that the practical performance of a WMN differs from the simulated one [4]. This is the reason for which some modifications of the original protocols have been proposed in the WMN implementations. In [5], a test-WMN (ReMESH) based on Optimized Link State Routing (OLSR) protocol combined with a modified ETX metric (in fact the Minimum Packet Loss Ratio parameter) is proposed. The authors have shown that in comparison to the original OLSR, the performance of the mesh network was improved, leading to more stable routes, lower packet loss rates, smaller delays, and in many cases a small increase in the network throughput. In [6], another test-WMN, called RoofNET, is described. RoofNET is based on a routing protocol named SrcRR that tries to maximize the throughput of the paths. The results presented in both previously cited papers were obtained in a real environment.

In WMNs, the routing metrics greatly impact the network performance. As it has been proved, they should also consider physical layer phenomena, like SINR, interflow interferences or the so-called flow self-interferences, introduced by the hidden and exposed terminal problems [7]. Examples of the most popular WMN routing metrics include: hop count, delay, jitter and ETX. There are approaches, which take into account physical layer processes, e.g., traffic aware metrics like PPTT [8]. However, these metrics are mainly probabilistic ones and they impose quite complex cross-layer operations. It is worth mentioning that the hop count metric does not establish the path according to its actual quality. The other metrics select the routes taking into account parameters of the component links. Therefore the routing protocol adapts to the network state. Routing protocols that use more than one metric can be found in literature. An example of such a protocol is Sharp Hybrid Adaptive Routing Protocol (SHARP) [9]. In the existing approaches, there is no decoupling between the monitoring and routing. This is the reason why it is very difficult to find a comparison of the same routing protocol with different routing metrics used. Nevertheless, some comparisons exist, e.g., in [10], in which the authors compared ETX metric with hop count metric using a grid topology only. Unfortunately, the grid topology is a particular case and it cannot yield relevant results for a wide variety of WMNs. However, a lot of theoretical comparisons exist and the most complete are [11] and [12].

The common approach for routing in both wired and wireless communication systems is the single path approach. However, it has been observed that the reliability and the performance of the network may be improved when more than one path between source and destination nodes is used. There are few scenarios, in which the multipath feature is useful. The simplest one is to discover the additional paths and to use them as a backup when the main route fails (AODV-BR [13], AOMDV [14] and AODVM [15]). This way, it is not necessary to perform the route discovery procedure every time the path breaks, because another path is already available in the routing table. This is the major advantage, especially in the networks with mobile nodes. When the multiple paths are used simultaneously [16], the traffic may be split among them on per packet or on per flow basis, enabling the load balancing. Another possibility is to replicate the data on each of the discovered paths, thus ensuring enhanced reliability. It has been shown that the improvement of multipath may be achieved only when a limited number of additional paths are kept in the routing table. According to [14] they should be limited to two or three paths, in order to avoid the existence of stale paths in the routing table. In [14] also the disjointness of the paths in the network is considered. Two paths may be either link or node disjoint. In the first case, it is acceptable for two paths to share common nodes. However, if the mutual node fails, both paths will become useless. The second option is much stricter, but at the same time it improves the reliability of the communication. The problem appears in small or sparse networks, because it may be impossible to establish node disjoint routes there.

III. AAODV

There are two main challenges for a routing protocol, i.e., finding the best path and loops avoiding. The “best path” can be defined as the path from the source to the destination that minimizes the end-to-end PLR, delay and jitter. One of the most popular protocols designed for ad-hoc networks, AODV, accomplishes only the second property by using sequence numbers in order to find loop free paths. AODV is relatively simple (see [2]). Every time a node wants to send a packet and does not know a route to the destination, it broadcasts a Route Request (RREQ) message. When any node, including the destination, receives this RREQ, it checks whether it has received a duplicate RREQ within a fixed interval of time. If such RREQ has been received, the node silently discards the newly received RREQ. During the RREQ broadcasting period, the reverse path (from the destination to the source) is established. When the destination receives a new RREQ it responds with a unicast message to the source – Route Reply (RREP). During the transmission of the RREP, the path from source to destination (the forward path) is established. Also, note that in the meantime when RREQ and RREP are sent, the intermediate nodes set their paths to the source and respectively to the destination. AODV will not always find the best path in the network, because for path selection it uses the sequence number as the first criterion and the hop count as the second one. More than one RREQ message can be sent to find a path to a given destination, what has an impact on the AODV control traffic overhead.

In order to make a metric based route selection in AODV, it is necessary that the source node receives more RREPs with the same destination sequence number as the destination sequence number already stored in its routing table. Such an approach is possible if the destination node does not change its sequence number and sends more than one RREP. This simply implies that the source should send more than one RREQ to discover the route to every destination or that more copies of the same RREQ should reach the destination via different paths. In AODV, as long as the source does not have a route to the desired destination, it broadcasts a RREQ with a new identifier, even if the packets should be routed to the same destination. Of course, the maximum number of RREQs that can be sent per second is limited. The main drawback of AODV is that it calculates the paths considering only the hop count metric, which is more appropriate for wired networks than for wireless ones, in which many factors should be taken into account when finding a path. They include path self-interferences, interferences between the paths (flows), the quality of links, etc. [11]. Considering this main drawback of AODV, we propose a new routing algorithm – Adaptive AODV (AAODV) that is an improvement of the AODV protocol.

AAODV is able to calculate and simultaneously use multiple routing metrics. This process is supported by a Monitoring Layer (ML) that is independent on the routing protocol. The ML is responsible for the measurements and the calculations of metrics in a per link manner. The Monitoring Layer consists of a Metric Container (MC). The ML components are implemented in every node. Also, at each node, the metrics for every neighbor, i.e., delay, jitter and ETX are stored in its MC. The hop count metric is implicitly implemented. A new kind of HELLO messages (see Figure 1) is used for ML data dissemination.

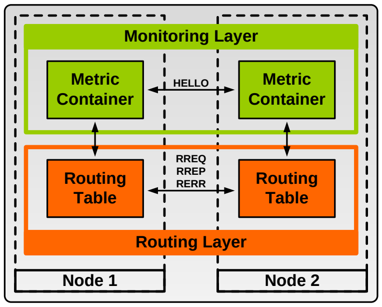


Figure 1 AAODV: Routing and Monitoring Layers

Each node sends the information about metrics for its neighbors (from the MC) in the HELLO messages, together with its node ID and timestamp. The timestamp determines the time, at which the HELLO was sent. Therefore, when a neighboring node receives the HELLO, it can calculate delay and jitter metrics. We assume that the nodes are synchronized. If the MC is empty (e.g., at the initialization of the network) the HELLO message will contain only the timestamp and an IP header. When a node receives a HELLO, it scans the message for any information addressed to it. If it finds this information it updates its MC. Afterwards, the node calculates the metrics to the source of the HELLO and updates the MC once again.

In AAODV the delay and jitter metrics are calculated according to the RFCs [17] and [18] using the HELLOs as probe messages. In our approach, the following exponential smoothing function (1) has been used for delay and jitter metrics estimation

$$d_{new} = \alpha * d_{old} + (1-\alpha) * d_{sample}, \quad \alpha \in [0;1] \quad (1)$$

where d_{old} – the old value of delay/jitter;
d_{sample} – the new sample of delay/jitter.

The ETX metric has been implemented using PLR, in the same way as in [5]. In this approach, the value of ETX per link represents in fact the probability of successful transmission of a packet, considering both the forward and reverse link delivery ratios. The following formula (2) is used for calculating per link ETX

$$ETX_{link} = P_{link} = d_f * d_r, \quad (2)$$

where d_f – forward link delivery ratio;
d_r– reverse link delivery ratio.

By default, the HELLO messages are generated every 2 seconds and the d_f and d_r are calculated by counting the successfully received HELLO messages at a node in the analyzed time window (20 s). The successful delivery per path must take into account the successful delivery on every link, therefore the path ETX is calculated as a product of link ETXs (3)

$$ETX_{path} = \prod_{i}^{n} ETX_{link_i}, \quad (3)$$

where n – number of links that constitute the path.

The path chosen from a source to a given destination in a network is the one with the highest ETX. Of course, the maximum value of the ETX is 1.

Note that the AAODV protocol is not limited to these metrics and can be implemented with any other per link calculated metric. As it was mentioned before, AAODV is based on AODV and it implements the same algorithm, which uses the sequence numbers in order to obtain loop-free paths. The main differences from AODV are as follows:

- AAODV nodes do not flood the network with RREQs when they are searching for a new route;
- AAODV nodes do not discard all the duplicate RREQs – this idea is also presented in [14].

In AAODV every time a node receives a packet, for which it does not have a route to the destination, it queues the packet and sends RREQ. If a new packet is received and needs to be routed to the same destination, the node checks two additional conditions in comparison to AODV, before it sends a new RREQ. First it checks whether another packet to the same destination exists in the queue. Then it checks if the RREQ for the packet already existing in the queue has expired or not. In the case that another packet to the same destination already exists in the queue and if the timer for the RREQ has not expired, the node does not send a new RREQ. This way, the RREQ flooding, evident in AODV, is inhibited and the overall overhead is decreased.

Moreover, as it was stated before, when an AAODV node receives a duplicate RREQ, it does not discard it immediately. The node verifies the sequence number and then checks if the metric for the path advertised in the RREQ is better than the one already existing in its routing table. If this condition is met, the node will update its routing table with the path from the RREQ, otherwise it will discard this RREQ.

Every time an AAODV node receives a RREQ, it takes the actions from the flowchart presented in Figure 2. Note that the RT entry refers to the path (stored in the routing table) to the RREQ's originator. RREQ entry refers to the path (indicated in the RREQ) to the RREQ's originator.

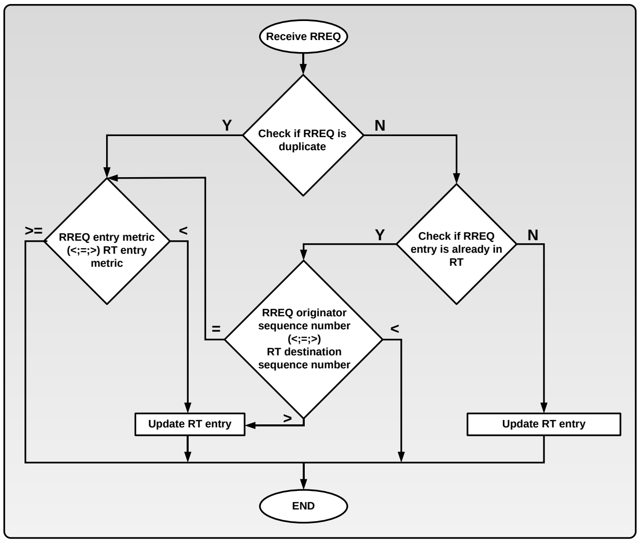

Receive RREQ
Check if RREQ is duplicate
Y
N
RREQ entry metric (<;=;>) RT entry metric
>=
<
Check if RREQ entry is already in RT
Y
N
RREQ originator sequence number (<;=;>) RT destination sequence number
=
<
>
Update RT entry
Update RT entry
END

Figure 2 Flowchart for implementing RREQ packet analysis

The algorithm depicted above proves the easy extendibility of AAODV to multipath, if instead of replacing the entry in the routing table, we will add it. This way, we will obtain more than one route to the source and RREPs will be sent on each of these paths. Multipath routing becomes effective only if the paths towards the destination are either node or link disjoint. In order to achieve this, several additional conditions should be considered.

IV. AAODV SIMULATIONS

In this section we compare the impact of the common routing metrics: hop count, delay, jitter and ETX on the network performance using ns-3 simulator.

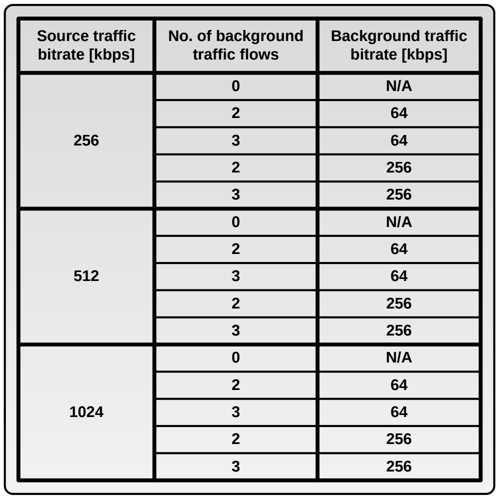
The simulated nodes were equipped with 802.11b Wi-Fi cards. The nodes used adaptive link rate that varies link bitrate from 1 Mbps to 11 Mbps. The topology was discovered by the nodes using HELLO messages. HELLO messages were broadcasted every 2 seconds, at the basic rate of 1 Mbps.

The simulation scenarios were based on eight randomly generated topologies. The nodes were distributed randomly in a square area. In order to be sure that the random topologies do not consist of isolated nodes, the possibility to communicate between any pair of nodes in the network was verified.

We considered two network sizes:

- Case 1: 16 nodes distributed uniformly in a square of 250 m x 250 m. The nodes are numbered from 1 to 16.
- Case 2: 25 nodes distributed uniformly in a square of 300 m x 300 m. The nodes are numbered from 1 to 25.

The application, which we used to measure the network performance, generated the Constant Bit Rate (CBR) traffic flow between the first node (node 1) and the last but one node (node 15 for Case 1 or node 24 for Case 2). The background traffic, Variable Bit Rate (VBR) flows, was generated between any two randomly chosen nodes that were different from the application source or the destination. The start time of the source traffic was fixed at 70 s and the start time for the background traffic was randomly chosen in the interval 40-50 s. The inter-packet interval deviation of the background traffic was equal to 1 μs. In Table 1 we summarized all the traffic simulation scenarios.



Source traffic bitrate [kbps]	No. of background traffic flows	Background traffic bitrate [kbps]
256	0	N/A
	2	64
	3	64
	2	256
	3	256
512	0	N/A
	2	64
	3	64
	2	256
	3	256
1024	0	N/A
	2	64
	3	64
	2	256
	3	256

Table 1 Traffic patterns

For each of the eight random topologies, we monitored PLR, delay and jitter of the source traffic using the traffic patterns from Table 1. In total we run over 1200 different simulations.

In Figure 3, we depicted the PLR, average delay and average jitter for the first five traffic patterns presented in Table 1 (source bitrate set to 256 kbps). We limited the delay and the jitter scales to 250 ms and 50 ms respectively. Note that the legend presented in Figure 3 is common for all the figures that follow. The averaged results from eight topologies show that AODV is outperformed by all variants of AAODV. Evaluating the network topology impact on performance, we observed that PLR, delay and jitter of source traffic can increase 5 to 6 times from one topology to another. In some cases the paths found by ETX are 2 or 3 times longer than the ones found by hop count. Despite that fact, we obtained similar results of averaged PLR, delay and jitter, regardless of the metric used. The paths discovered with AAODV and ETX have fewer retransmissions, but the length of the path affects the throughput. This leads to the conclusion that a combination between ETX and hop count should yield better results.

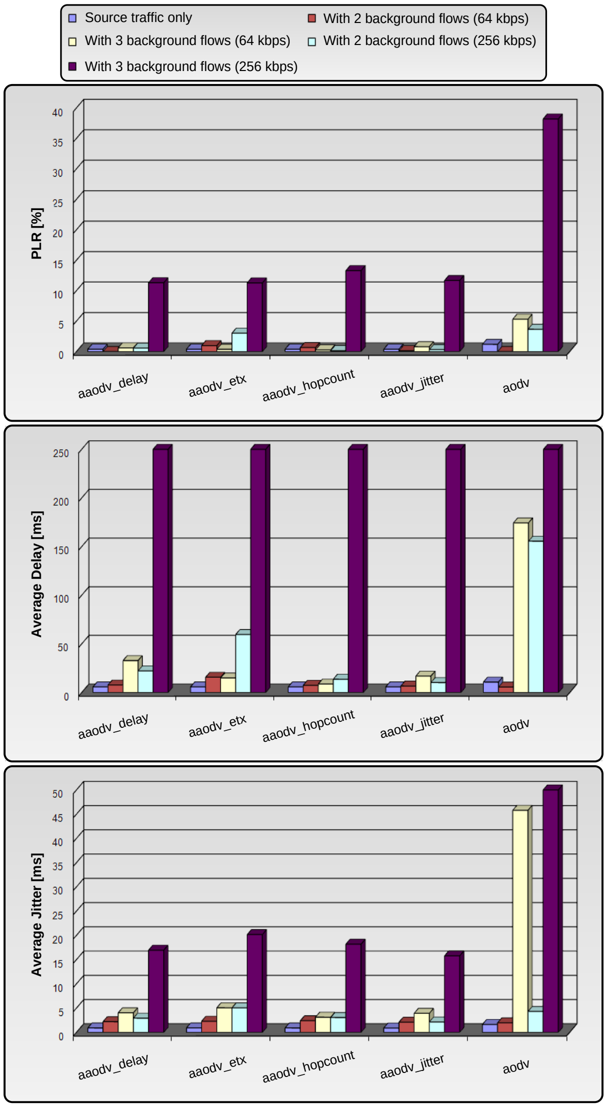


Figure 3 PLR, average delay, average jitter for the source traffic (16 nodes, 250 m x 250 m, 256 kbps)

In the next step we increased the source traffic bitrate to 512 kbps (topologies were the same). In Figure 4, we depicted the PLR, which slightly increased.

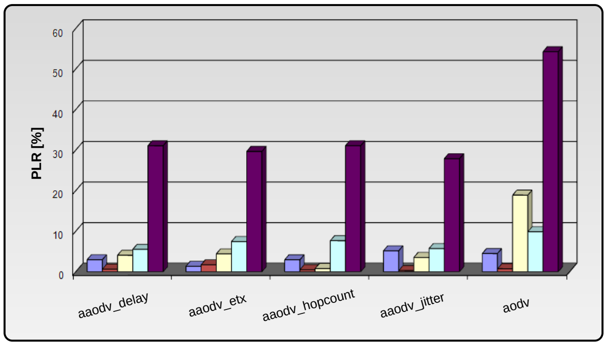


Figure 4 PLR of the source traffic (16 nodes, 250 m x 250 m, 512 kbps)

We observed that in this case the delay has dramatically increased, especially when we generated the background traffic with a bitrate of 256 kbps. The detailed analysis has shown that the source traffic delay increased very much in all cases in which the background traffic shares the paths. Note that the packets, which we used for metric calculation, were much smaller than the ones generated by the source.

In the last case, in which the source traffic was set to 1 Mbps, PLR, delay and jitter values increased more than in the previous cases. In Figure 5 we present the PLR graphic for this case. For all metrics, the delay exceeds our imposed limit of 200 ms.

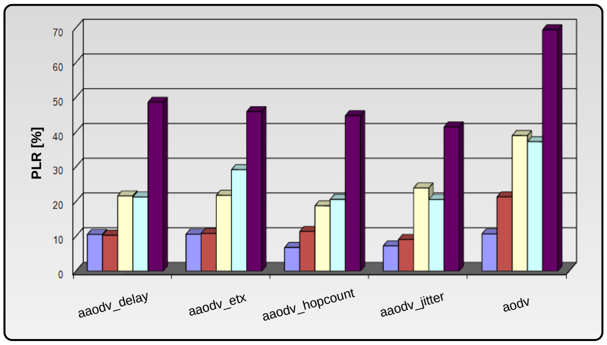


Figure 5 PLR of the source traffic (16 nodes, 250 m x 250 m, 1024 kbps)

In the second topology, in which we used 25 nodes distributed uniformly in a 300 m x 300 m square, the interferences have increased and more drastically affected the performance of the network.

In Figure 6, we depicted PLR, average delay and average jitter for the first five traffic patterns presented in Table 1, i.e., when a 256 kbps source bitrate was set. As before, we also limited the scale for the delay and the jitter to 250 ms and 50 ms respectively. Further, if we increased the source bitrate, PLR, delay and jitter have also increased until the network became unusable (according to our criteria).

The AAODV results have shown that the metrics calculation method and its usage (i.e., to determine the cost of the path) can be wrong. Firstly, the metric is used during path setup phase only, and it is calculated on a per link basis. Additionally, the metrics were calculated using HELLO messages transmitted at 1 Mbps. A direct consequence of this is that the metrics are not aware of, and do not consider the real throughputs available per links. This drawback is resolved in [19] where the HELLO messages are sent using adaptive bitrates. It also appears that routing metric estimation has to take into account the physical layer phenomena, and the path quality estimation cannot be limited to per link operations only, but should take into account the whole interference area of nodes, which constitute the path. A special care should also be given to the choice of measurements repetition frequency.

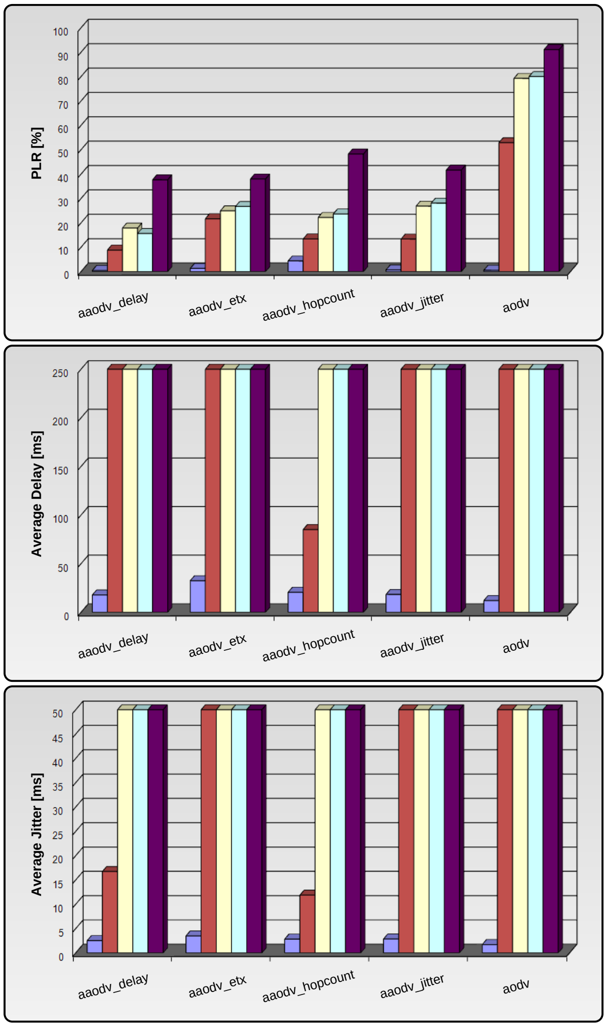


Figure 6 PLR, average delay, average jitter of the source traffic (25 nodes, 300 m x 300 m, 256 kbps)

V. AAOMDV

The previous section has shown that none of the metric behaves better than the other ones in case of single path routing. Considering this result, we propose a new routing protocol, named AAOMDV. It enables discovering of multiple paths between a source and a destination. By using RTM, AAOMDV is able to detect the degradation of the quality of the data forwarding path, and to trigger a path change based on algorithms presented later in this paper. Moreover, the functional separation of routing and monitoring mechanisms in AAOMDV make it more scalable and flexible.

A. Paths disjointness

As stated above, AAOMDV is able to find multiple link disjoint paths in the network. Please note, that the disjointness property refers strictly to the set of routes established between the same source and destination pair. The routes between different source and destination nodes can share the same links. In order to enhance the network performance, it is desirable for the traffic flows to follow paths that do not have many common links. In AAOMDV, this is accomplished by the usage of active path selection and switching algorithms described in Sections VI and VII.

In order to be link disjoint, the paths should fulfill two conditions indicated in [14]:

- For every created path the next hop must be different;
- The last hop towards the destination must differ from path to path.

B. Paths discovery algorithm

The paths discovery algorithm in AAOMDV is based on the Route Request – Route Reply (RREQ/RREP) messages exchange, present also in AAODV. The only difference is that in AAOMDV these messages contain also the information about the last hop on the path in order to be able to achieve the disjointness property.

Every time a node wants to send a packet and it does not have any available path to the destination in its routing table, it sends a RREQ message via the broadcast channel. Note that, like in AAODV and in contrast to AODV, the node sends only one RREQ. If no RREP has been received after a determined period of time, another RREQ is broadcasted. During the RREQ propagation, the reverse paths from the destination and the intermediate nodes are set up to the originator of this RREQ. The flowchart depicted in Figure 7 shows how the multiple paths are established during the RREQ broadcasting.

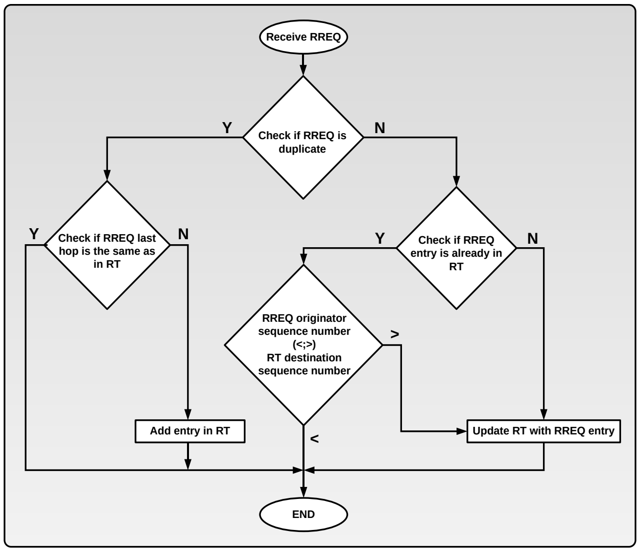

Receive RREQ
Check if RREQ is duplicate
Y
N
Check if RREQ last hop is the same as in RT
Y
N
Check if RREQ entry is already in RT
Y
N
RREQ originator sequence number (<;>) RT destination sequence number
>
<
Add entry in RT
Update RT with RREQ entry
END

Figure 7 AAOMDV paths discovery algorithm – RREQ broadcasting

The information about the last hop on the path allows accepting or rejecting the newly obtained path. This information refers to the penultimate node on the path towards the RREQ originator. Every established path has to have a unique last hop and next hop addresses, and this way the link disjoint paths towards the originator of the RREQ are established. Moreover, this approach also helps in loop avoidance.

AAOMDV can be easily adapted to support the node disjoint mode. The node disjointness property would be enabled through the rejection of the duplicated RREQ messages in the intermediate nodes. However, in a small WMN the probability of establishing multiple node disjoint paths between the same source and destination pair is quite low. For this reason, the adaptive selection of the mode can be a desired solution, i.e., if the network density is big enough and the connectivity between nodes is relatively high, then only the node disjoint paths should be allowed. This way the communication reliability would be improved. On the contrary, if each node has only few neighbors, then only the link disjoint path should be allowed.

As in AAODV, the sequence number mechanism is used to prevent the nodes from keeping the obsolete information in their routing tables. The loops can be caused by the acceptance of all the duplicated RREQs and by keeping the stale paths in the routing table. AAOMDV avoids both these situations. All the paths to the particular destination must have the same sequence number and if the one with a higher number would be found, all the previously established paths have to be deleted. When the node receives the RREQ with the same sequence number, it verifies the quality of this path and compares it with the others in order to keep the best available ones. The number of additional paths is limited. In Figure 8 an example of setting up multiple paths is presented.

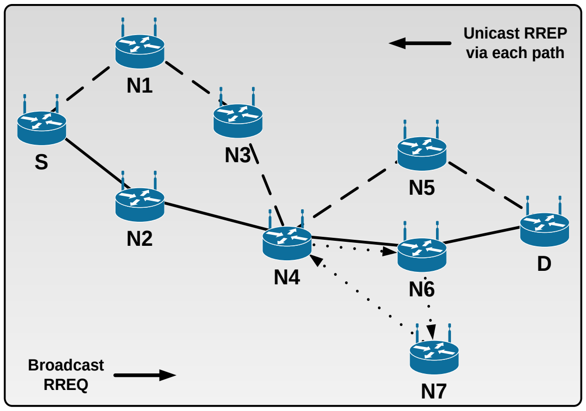
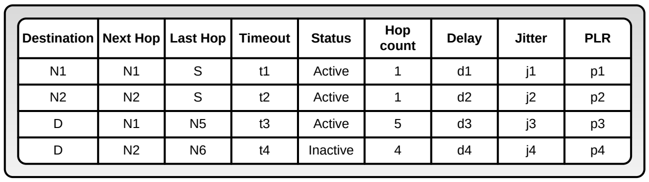


Figure 8 Setting up multiple paths

In order to establish multiple paths from the source to the destination and at the same time from the intermediate nodes to the same destination, unicast RREPs are sent on each of the paths established using the RREQ broadcasting. As we consider link disjoint paths and not node disjoint paths, an intermediate node can have more than one path to the originator of the RREQ. Hence, when an intermediate node receives the RREP, it should know onto which path this RREP message is to be forwarded. The last hop information helps to distinguish between all the paths. Please note that even if AAOMDV is implemented as a link disjoint multipath routing protocol, it is possible that it discovers only node disjoint paths. This drawback is explained in Section C.

According to Figure 8, source S broadcasts RREQ every time it searches for paths to destination D. During the RREQ propagation phase, the intermediate nodes complete their routing tables with paths towards S. These paths have different last hops (N1 and N2). When D receives the RREQ, it has to respond with a corresponding RREP. As it can be seen from Figure 8, N4 has two paths to S and when the RREP reaches N4 through the path drawn by the dashed line, the RREP should be forwarded to S on the same path. This is accomplished using last hop information to S, which is represented by N1. The dotted arrows in Figure 8 show a possible loop due to acceptance of a duplicate RREQ. N4 receives the RREQ, accepts it, completes its routing table with a path to S (drawn by a solid line) and rebroadcasts the RREQ. Node N6 receives it and performs the same operations as node N4. At the end, the RREQ will reach N4 again from node N7. Node N4 will reject this duplicate RREQ as the last hop (N2) is the same. In result, the loops are avoided. Note that multiple paths can also be set in intermediate nodes.

In Figure 9 we depicted the routing table available at node S after the route discovery procedure. As it can be seen, for every path we have more than one metric associated. The metrics are used to determine the quality of the paths as described in Section VI. Every path is associated with its own timeout that is updated when the path is used. Note that this timer will not be updated in the intermediate nodes if the path is not used.



Destination	Next Hop	Last Hop	Timeout	Status	Hop count	Delay	Jitter	PLR
N1	N1	S	t1	Active	1	d1	j1	p1
N2	N2	S	t2	Active	1	d2	j2	p2
D	N1	N5	t3	Active	5	d3	j3	p3
D	N2	N6	t4	Inactive	4	d4	j4	p4

Figure 9 The routing table structure at node S

In fact, all the paths will have the timer updated in all the nodes including the intermediate ones, because the probe messages will be sent via the inactive paths, as it is described later.

C. Link disjointness problem

The topology shown in Figure 10 is considered.

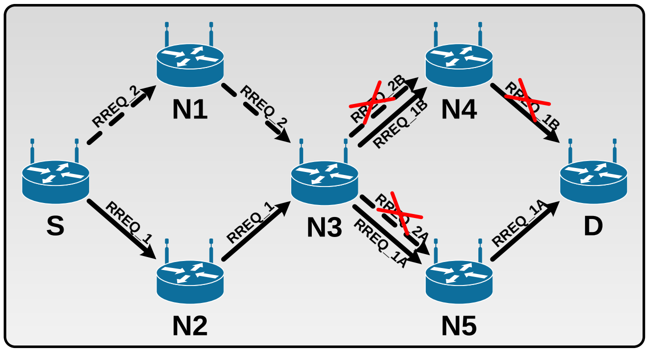


Figure 10 Link disjointness problem

Considering Figure 10, it should be possible to find two link disjoint paths from source S to destination D, namely [S, N1, N3, N4, D] and [S, N2, N3, N5, D]. However, this is not possible in some situations. At the beginning, node S broadcasts a RREQ message. The packet processing time at every node may be different and because of that the RREQ copies arrive at different times in the intermediate nodes. The first copy (RREQ_1) goes through N2 and in N3 it is forked. RREQ_1A reaches the destination D through N5, the first path [S, N2, N3, N5, D] is discovered and the RREP is sent on it. RREQ_1B gets to D via N4, but it must be discarded, as the last hop N2 is common for the already added path. In the meantime RREQ_2 arrives to N3 through N1. N3 adds the second path to S in its routing table and rebroadcasts RREQ_2. It is received in N4 and N5 nodes, but both of them must reject it, while in both cases it does not create a new link disjoint path. This happens, because RREQ_1 is still buffered and the first hop to S, i.e., N3 is the same for both RREQ copies. The result is that only one path may be established.

The described problem shows that in some cases it is hard to find more link disjoint paths. This is a drawback in a small network, as it becomes very difficult to find multiple paths between source and destination. One solution for this problem may be the reduction of the time during which the RREQ is buffered in nodes. We also expect that in more dynamic scenarios it would be possible to achieve link disjoint paths, e.g., when RREQ_1B copy will be lost between N3 and N4.

D. *Routes Limit Validation*

The wireless medium is highly unstable and the transmission quality changes in time. Due to the mobility of nodes, frequent topology changes may occur. These are the reasons to store a limited number of backup paths in the routing table. This way, the content of the routing table is more recent. After a new path is added to the routing table, AAOMDV verifies whether the number of paths does not exceed the maximum number permitted, defined by the configurable parameter – *routes_limit*. As mentioned before, it has been proven in [14] that the gain of a multipath is achievable with two/three paths for one destination. If the limit is exceeded, then the quality of all the paths is evaluated using the algorithm described in Section VI and the worst one is deleted from the routing table.

E. *Enhanced Monitoring Layer – Real Time Monitoring*

The AAOMDV nodes can have more than one path to a destination, but only one is used for data forwarding – in the routing table it has the active flag set (see Figure 9). During the route discovery phase, the nodes activate the first path that they obtain towards the destination. Note that at this step the nodes do not consider any kind of metric – the nodes start routing the data packets from the queue as soon as they obtain the first path to the destination. After a determined period, starting from the first RREP, the source sends a message named Route Activation (RACTV, see Figure 11) to activate the path to the destination that has been chosen as the best one using the path selection algorithm from Section VI (INITIAL_MODE). The mentioned period is named MULTIPATH_DISCOVERY_TIME and has a default value of 2 seconds.

The Enhanced Monitoring Layer (EML) of AAOMDV incorporates all the functionality of the AAODV ML and has some new features. The most important feature of the EML is the capability to monitor in real-time regime the multiple end-to-end paths available between the source and the destination. Using the traffic packets sent from the source to the destination, it is possible to monitor PLR, delay and jitter of the active path. Probe messages are sent in order to evaluate delay and jitter of the inactive paths.

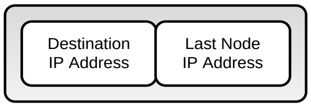


Figure 11 RACTV header

The ETX can be used to substitute the PLR metric on the inactive paths. Note that the evaluation of PLR, delay and jitter on the active path is realistic, as it considers the real traffic. The delay and jitter evaluation on the inactive paths is done by using active probing and it does not reflect the real delay and jitter, which would be achieved if we routed the data traffic on them. The probe messages, called Route Probes (RPRBs), sent on the inactive paths, have the same payload size as the averaged payload size of the data packets transmitted during the last 5 seconds between the source and the destination through the active path. Once again, it should be noted that although the size of the probes is appropriately matched, the transmission of only two packets cannot emulate the real flow of packets. Therefore, the results for the delay and jitter are estimative. The RPRB message header is depicted in Figure 12. The delay and jitter are calculated according to [17] and [18].

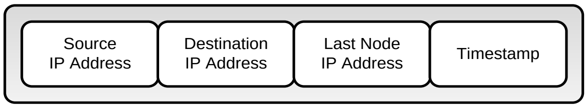


Figure 12 RPRB header

The information about all the paths between the source and the destination is evaluated at the destination and sent back to the source using a Route Report message (RRPRT), shown in Figure 13. The size of this message is variable since the number of paths obtained between the source and the destination, although limited, is not fixed.

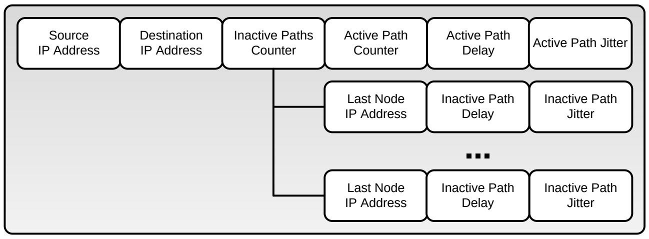


Figure 13 RRPRT header

The Active Path Counter represents the number of packets successfully received by the destination during the last 5 seconds interval. When it receives the RRPRT message, the source can calculate the PLR for this period as it knows how many packets it has sent. In fact, all the information known at the source about the end-to-end paths is 5 s old. The delay and jitter are evaluated at the destination and sent back in the RRPRT messages to the source. The sequence of the messages exchange implemented in the EML for the RTM is depicted in Figure 14.

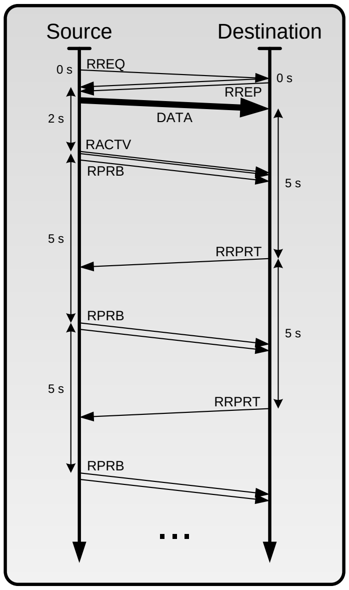


Figure 14 The exchange of messages in EML

In order to take full advantage of the multipath feature and of the RTM, it is necessary to determine the best path and to send the traffic using this path. The path selection and switching algorithms are protocol independent, and may be applied to any multipath routing protocol that does not use all the known paths simultaneously for data forwarding. Both algorithms will be described in the following sections.

VI. Active Path Selection

The active path selection algorithm should be implemented in each node in the network. Two modes of the active path selection algorithm that can be distinguished are called INITIAL_MODE and NORMAL_MODE. The first one is used to select the best path of the available ones discovered during MULTIPATH_DISCOVERY_TIME. At the beginning of the communication, the end-to-end real metrics are not available, so the per link calculated metrics, i.e., delay, jitter, ETX and hop count are considered. The second mode of the algorithm is used when a new RRPRT message is received at the source node. In this case, the real metrics, i.e., the real delay, real jitter, PLR and hop count are considered. AAOMDV deals with multi-parametric metric and it needs a special algorithm to compare the quality of two or more paths and to select the better one.

Let $M = \{m_1, m_2, m_3, ..., m_k\}$ be the set of metrics related to every path. $W = \{w_1, w_2, w_3, ..., w_k\}$ is the vector of weights, which express the importance of every particular metric.

The algorithm for paths comparison utilizes all the metrics M **simultaneously** to evaluate and confront the two paths with each other. At the initial stage of the algorithm all the metric weights are defined. These weights allow differentiating the importance of the metrics. This way it is possible, e.g., to favor the paths with a smaller packet loss level by assigning a higher value for the weight related with the PLR metric. The weights are set with respect to the application needs. Here we consider also the hop count metric. Hop count metric should be considered when building a metric as the throughput achievable in an arbitrary WMN is proportional to $\Theta(W \cdot n^{-1/d})$, where d is the dimension of the network, n the number of nodes and W is the total bandwidth. In a two dimensional network, the throughput can be as small as $\Theta(W \cdot n^{-1/2})$ [20]. The hop count metric does not cause any implementation problems, because it is already used by the AODV protocol. The next parameters that must be defined are the threshold values for all the metrics considered in the algorithm. A path that has a metric, which exceeds its threshold, is considered the worst path. If all the available paths to a destination are considered as the worst path, it is desirable to send a new RREQ to the destination and establish new paths (of course, if this is possible). Both sets of parameters, i.e., weights and thresholds, may be configured and adjusted according to operator or user requirements, e.g., in the policy based approach. After these steps, the algorithm is ready to compare the paths. For all the comparisons the Composite Metric is calculated. The paths are compared two by two. For calculating the Composite Metric formula 3 applies:

$$Composite\_Metric_{p1} = \\ = \frac{d_{p1} \cdot w_d}{d_{p1} + d_{p2}} + \frac{j_{p1} \cdot w_j}{j_{p1} + j_{p2}} + \frac{p_{p1} \cdot w_p}{p_{p1} + p_{p2}} + \frac{h_{p1} \cdot w_h}{h_{p1} + h_{p2}} \qquad (3)$$

where:

d, j, p, h – delay, jitter, PLR and hop count values;

w_d, w_j, w_p, w_h – weights correlated with metrics.

Formula 3 is applicable when the algorithm is used in NORMAL_MODE, i.e., when real metrics can be used. In INITIAL_MODE per link calculated metrics are used and the component associated with the PLR must be replaced by a corresponding component calculated for the ETX metric. The ETX indicates the packet delivery ratio and the path with a higher ETX value is considered to be better, therefore the ETX component must be subtracted from formula 3. In the NORMAL_MODE, the ETX is also used instead of the PLR in the same way for the evaluation of the Composite Metric for inactive paths. In order to improve the stability of the network and to give a priority to the currently active path, the algorithm defines one more configurable parameter, i.e., ACTIVE_PATH_MARGIN. If the evaluated path is active, then this parameter specifies how much better (in percent) the inactive path should be in order to replace the active one. The default value of this parameter is 10%.

In the case when two paths have equal Composite Metrics, the first one is indicated. This way the changes in the network configuration are avoided, because the currently active path is always preferred as it is the first one, when compared with any other route.

VII. Active Path Switching algorithm

The challenge appears when two paths are created between different pairs of source and destination nodes, but parts of these paths are common for both of them. It is also possible that the backup paths overlap. In this scenario, if both sources start to send the traffic via the common links, then the active paths performance degrades; therefore both nodes will switch to the second available path. The situation may repeat, causing the so-called flip-flop phenomena and will lead to oscillations in the network configuration. To overcome this challenge, an algorithm that controls the active paths switching from node to node is needed. The algorithm can be either centralized or distributed. In the centralized approach, the switching should be controlled by a central entity. In the second approach, the decision of path switching is distributed among nodes. In this paper we will implement a distributed algorithm for the path changing. The distributed approach is a more scalable solution. No central node is needed, so it is possible to use it regardless of the network size, which can be understood as the number of nodes, as well as the spatial extent. Nodes are able to autonomously adapt to the changes in the network and make autonomic decisions according to the results of the performance measurements. Their decisions are taken locally, but finally it should lead to the global optimization of the network configuration.

The proposed algorithm works as follows. All the source nodes that are currently sending data to their destination nodes are aware of the quality of each path to the destination stored in their routing tables. This knowledge is obtained using monitoring of both active and inactive paths. As described previously, the active path monitoring is piggybacked in the data packets and the RPRB messages are used to evaluate the inactive ones. The source nodes take their decisions based on the periodically received RRPRT messages from the destination nodes. Every time the source node receives a RRPRT message, it updates its routing table with fresh measurement results, compares all the available routes with one another and chooses the best one. If it is the same as the currently active path, nothing happens. If another path is chosen, the algorithm starts working. First, a random number (*random_number*) with uniform distribution is chosen from a range of [0; 100]. In the algorithm a configurable threshold value (*change_threshold*) is defined and if the *random_number* is smaller than *change_threshold* nothing changes and the next RRPRT message is awaited. In the opposite case, the node makes the better path active, deactivates the previously active one, applies these changes in its routing table, sends a RACTV message on the new active path and starts to send data using it. The *random_number* and the *change_threshold* values have critical impact on the algorithm behavior. Their task is to avoid the continuous changes of the active path and to stabilize the protocol functionality. It is possible that the *change_threshold* value may be inaccurate and the *random_number* may be always smaller. In this case the performance of the network will be weak, although it could be simply improved by changing the active path. For this reason a new parameter (*change_limit*) has been defined that determines the limit value of the path change cancellation. The consecutive unsuccessful attempts to switch the path are calculated. If their number exceeds the *change_limit* value, the node changes the active path regardless of the *random_number* value. The election of distribution type and threshold values has a great impact on the network performance (see Section VIII). Figure 15 shows the structure of the distributed algorithm.

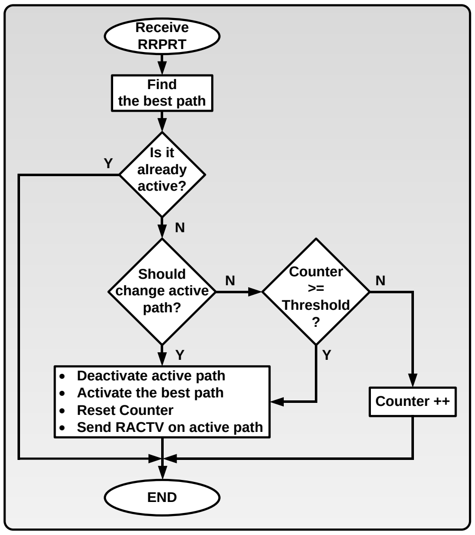


Figure 15 Distributed algorithm for active path switching

VIII. AAOMDV Simulations

In this section the performance of AAOMDV is compared to AODV. The comparison was made in a network that consisted of 16 nodes, randomly located in a square area of 300 m x 300 m. Each node was equipped with IEEE 802.11b Wi-Fi card and the communication range was ca. 175 m. We run the simulations with three different random number generator seed values, in order to get a different placement of nodes. The obtained results were averaged. In order to be sure that the random topologies do not consist of isolated nodes, the possibility to communicate between any pair of nodes in the network was verified.

The source traffic was generated between a specific pair of nodes, while the background traffic sources and destinations were chosen randomly. The maximum number of paths in the routing table for a specific destination has been set to 3.

As reference for the AAOMDV performance evaluation we used AODV. To check the behavior of both protocols with a different network load we changed the source and the background traffic throughput and the number of background traffic flows. In AAOMDV simulations we additionally checked the influence of the *change_threshold* and the *change_limit* parameters on the performance of the network. The simulation scenarios of AODV/AAOMDV are presented in Table 2.

Source traffic bitrate [kbps]	No. Of background flows	Background traffic bitrate [kbps]	change_threshold	change_limit
128 256 512	0	-	0 25 50 75 100	0 1 2 3 10 100
	1 2	64 128 256		

Table 2 Simulation parameters

We performed about 2000 simulations to obtain the results.

Firstly, we verified the proper behavior of AAOMDV using PyViz visualiser [21], which is a part of the ns-3 simulator. Figure 16 shows an example of PyViz output. It can be observed that between the same source and destination two different paths have been established by AAOMDV.

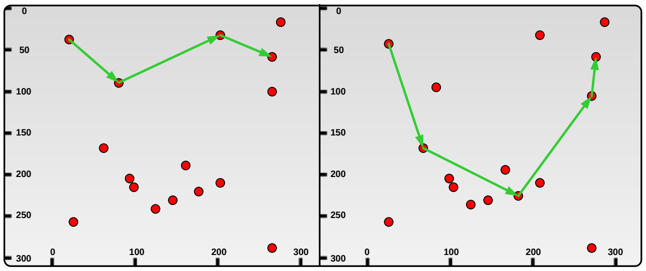


Figure 16 PyViz output – two paths found in the network

Figure 17 shows the influence of the AAOMDV *change_threshold* and *change_limit* parameters on PLR, average delay and average jitter, when only the source traffic was generated in the network. The detailed analysis of the mutual interdependencies between the above mentioned AAOMDV parameters led us to the following conclusions:

- In order to permit nodes to change their active path frequently, both parameters should be set to a low value;
- If both parameters have relatively high values then it is very hard to switch the active path;
- A node is always restricted from path changes if the *change_threshold* is set at 100% and the *change_limit* is set much higher than 0 (e.g., 100). The initially chosen path will be used until it gets lost;
- A node is always permitted to change the path if the *change_threshold* is set to 0% or the *change_limit* is set to 0. It means that no postponing of the path change is allowed.

Therefore, both parameters are dependent on each other and their values must be correlated in order to obtain the desired configuration.

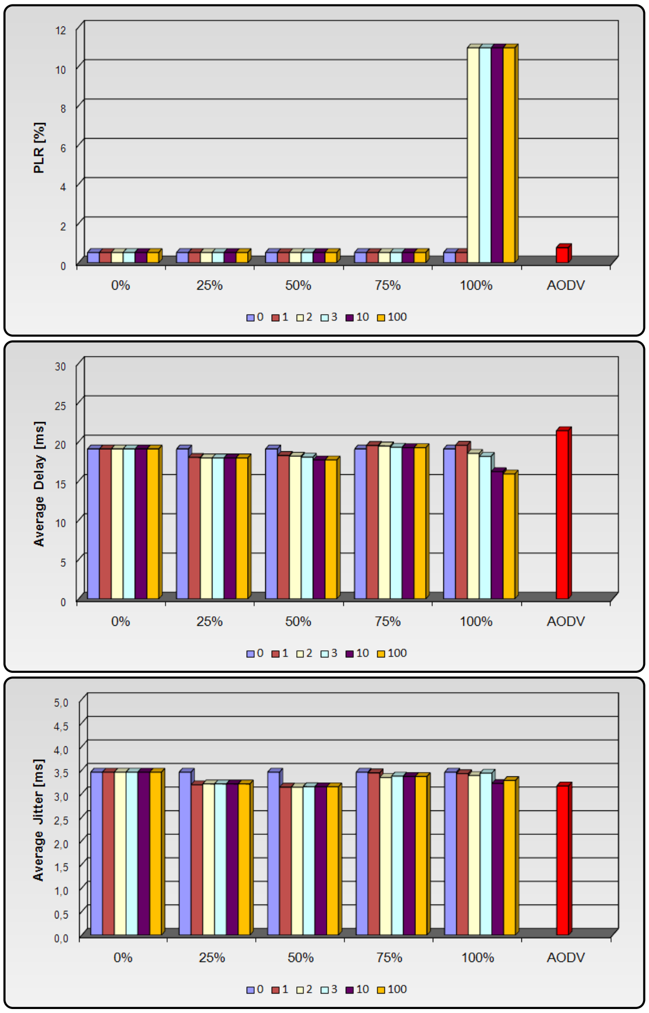


Figure 17 PLR, average delay, average jitter of the source traffic (source traffic – 128 kbps, no background traffic)

It can be observed that the frequent changes of the active path were advisable for improving the quality of transmission. If switching of the active path was not permitted, the traffic packets started to be lost. On the other hand AODV, which is a single path protocol based on the hop count metric, yielded good overall results. This means that when the network load was low, the Composite Metric did not outperform the hop count metric. When the throughput of the traffic increased and the network load grew, the nodes started to switch the paths more frequently (see Figure 18).

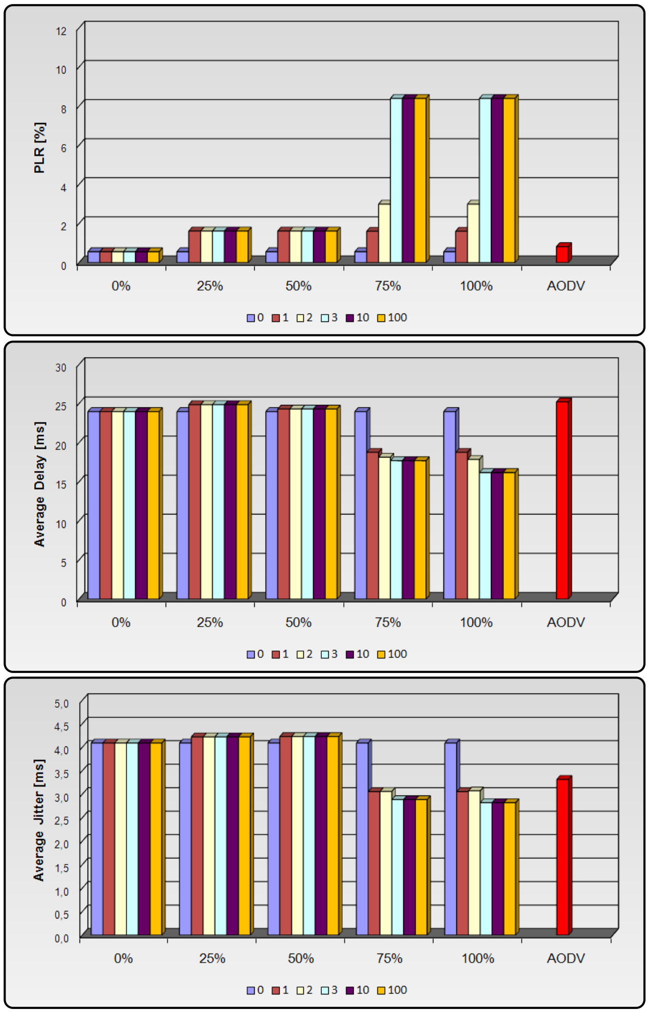


Figure 18 PLR, average delay, average jitter of the source traffic (source traffic – 256 kbps, no background traffic)

Figure 19 shows that in some cases the AAOMDV routing protocol was able to outperform the AODV significantly. In all simulated cases the AAOMDV PLR was at least three times lower than the AODV PLR. The delay and jitter had acceptable values in both cases. As we mentioned before, very high *change_threshold* and *change_limit* values cause that the first obtained path is used until it gets lost regardless of its parameters. Therefore, this configuration shows the impact of the routing metric used on the network performance. The conclusion is that when the network load increases, the Composite Metric provides better results than the hop count metric. It should be noted that when the traffic generated in the network increased, it became more viable to limit the number of active path changes. Similar results are also depicted in Figure 20. In this case, AAOMDV also outperformed AODV, although the PLR was a little bit higher. The obtained results have confirmed that the two parameters, i.e., *change_threshold* and *change_limit* have a great impact on the overall network performance, and that permanent monitoring of the paths and the dynamic paths switching provides better results than usage of AODV.

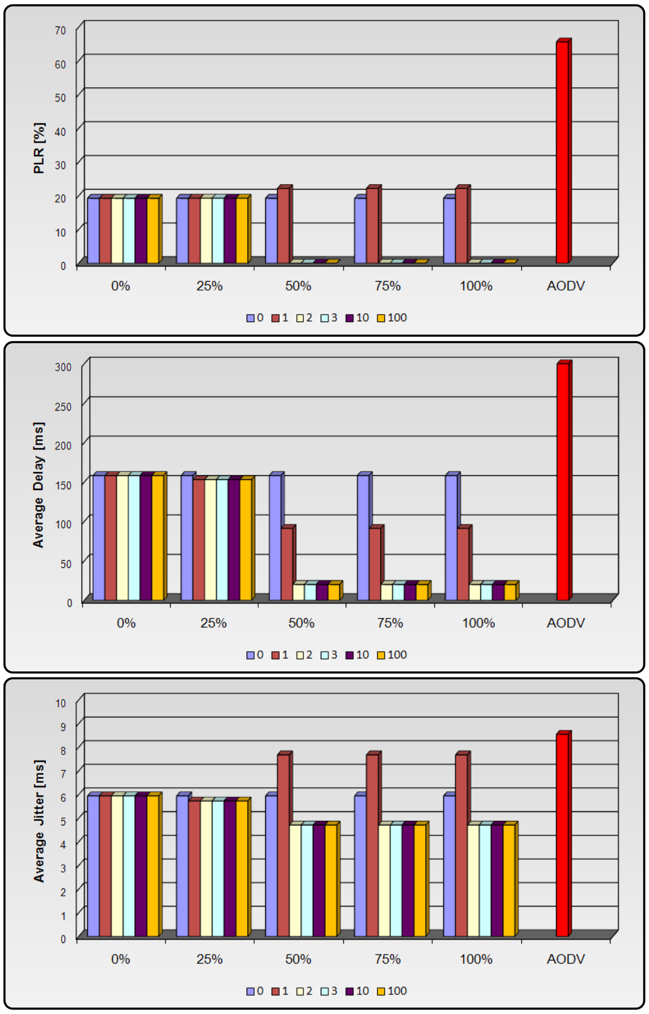


Figure 19 PLR, average delay, average jitter of the source traffic (source traffic – 512 kbps, one background traffic – 64 kbps)

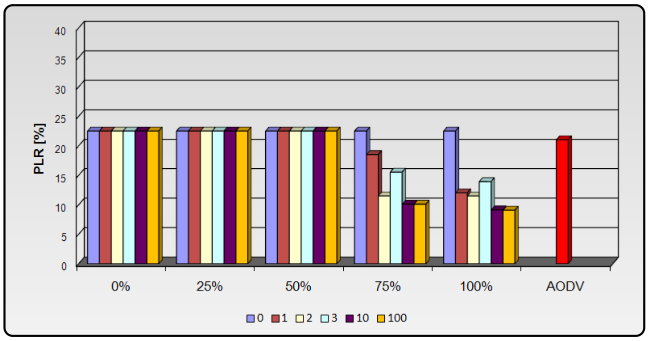


Figure 20 PLR of the source traffic (source traffic – 256 kbps, one background traffic – 128 kbps)

IX. Conclusions and Future Work

The starting point of this paper was a comparison between the most popular routing metrics, i.e., hop count, delay, jitter and ETX, and our main goal was to determine the best one to be used in WMNs. To achieve that efficiently, we designed a new protocol AAODV, able to calculate the path cost based on more than one metric. AAODV, due to the separation of routing from paths monitoring, can use any per link calculated routing metric. The results of simulations led us to the conclusion that none of the analyzed metrics behaves significantly better than the others. This conclusion was in a sense predictable as none of the tested metrics is traffic aware or fully addresses the challenges that appear in a wireless environment, e.g., interflow and intraflow interferences, exposed and hidden terminal problems, etc. The evident problem of AAODV is that the path quality is only monitored during the path setup phase. This is why we decided to add the multipath capability to AAODV and enable continuous end-to-end monitoring of all the paths. In order to find the best path we defined the Composite Metric that takes into account PLR, delay and jitter weighted appropriately to network operator preferences. In AAOMDV a distributed path switching algorithm has been implemented and the Composite Metric is used for the active path selection.

The benchmark for the AAOMDV performance evaluation was AODV. It has been observed that when the network load was low both AAOMDV and AODV yielded good, similar results. It cannot be affirmed that one of the two routing protocols outperforms the other. However, when the network load increased, AAOMDV outperformed AODV by providing about two times smaller PLR and delay of the analyzed traffic.

A weak point of AAOMDV is the necessity of fine tuning of the *change_limit* and the *change_threshold* parameters in order to optimize the network performance. This procedure should be modified in order to have self-tuning properties and we will focus on it in our future work.

References

[1] H. Ştefănescu, M. Skrocki, and S. Kukliński, AAODV Routing Protocol: The Impact of the Routing Metric on the Performance of Wireless Mesh Networks, The Sixth International Conference on Wireless and Mobile Communications (ICWMC 2010), pp. 331-336, September 20-25, Valencia, 2010.

[2] C. Perkins, E. Belding-Royer, and S. Das, Internet Engineering Task Force (IETF): Ad hoc On-Demand Distance Vector (AODV) Routing, Request for Comments (RFC) 3561, July 2003.

[3] http://www.nsnam.org/

[4] D. Johnson and G. Hancke, Comparison of two routing metrics in OLSR on a grid based mesh network, Ad hoc Networks, Volume 7 (2), pp. 374-387, 2009.

[5] D. Passos, D. V. Teixeira, D. C. Muchaluat-Saade, L. C. S. Magalhães, and C. V. N. Albuquerque, Mesh Network Performance Measurements, In 5th International Information and Telecommunication Technologies Symposium, 2006.

[6] J. Bicket, D. Aguayo, S. Biswas, and R. Morris, Architecture and Evaluation of an Unplanned 802.11b Mesh Network, ACM Mobicom Conference, pp. 31-42, September 2005.

[7] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, MACAW: A Media Access Protocol for Wireless LAN's, Proceedings of the Conference on Communications Architectures, Protocols and Applications, pp. 212-225, London, United Kingdom, 1994.

[8] S. Yin, Y. Xiong, Q. Zhang, and X. Lin, Traffic-aware Routing for Real Time Communications in Wireless Multi-hop Networks, Wireless Communication and Mobile Computing, Volume 6 Issue 6, pp. 825-843, September 2006.

[9] V. Ramasubramanian, Z. J. Haas, and E. G. Sirer, SHARP: A Hybrid Adaptive Routing Protocol for Mobile Ad Hoc Networks, In Proceedings of the ACM Symposium on Mobile Ad Hoc Networking and Computing, pp. 303-314, Annapolis, Maryland, June 2003.

[10] X. Ni, K. Lan, and R. Malaney, On the performance of expected transmission count (ETX) for wireless mesh networks, Proceedings of the 3rd International Conference on Performance Evaluation Methodologies and Tools, Athens, Greece, 2008.

[11] R. Baumann, S. Heimlicher, M. Strasser, and A. Weibel, A Survey on Routing Metrics, TIK Report 262, ETH Zürich, February 2006.

[12] M. E. M. Campista at al., Routing Metrics and Protocols for Wireless Mesh Networks, IEEE In Network, Volume 22 Issue 1, pp. 6-12, 2008.

[13] S. J. Lee and M. Gerla, AODV-BR: Backup routing in Ad Hoc networks, Proceedings of IEEE WCNC 2000, Volume 3, pp. 1311-1316, Chicago, September 2000.

[14] M. K. Marina and S. R. Das, Ad hoc on-demand multipath distance vector routing, Wireless Communications & Mobile Computing, Volume 6 Issue 7, pp. 969-988, November 2006.

[15] Z. Ye, S. V. Krishnamurthy, and S. K. Tripathi, A framework for reliable routing in mobile ad hoc networks, IEEE INFOCOM, Volume 1, pp. 270-280, Sanfrancisco, March 2003.

[16] S. J. Lee and M. Gerla, Split Multipath Routing with Maximally Disjoint Paths in Ad hoc Networks, Proceedings of IEEE ICC 2001, pp. 3201-3205, Helsinki, Finland, June 2001.

[17] G. Almes, S. Kalidindi, and M. Zekauskas, A One-way Delay Metric for IPPM, IETF RFC 2679, September 1999.

[18] C. Demichelis and P. Chimento, IP Packet Delay Variation Metric for IP Performance Metrics (IPPM), IETF RFC 3393, November 2002.

[19] D. Aguayo, J. Bicket, and R. Morris, SrcRR: A High Throughput Routing Protocol for 802.11 Mesh Networks, http://pdos.csail.mit.edu/~rtm/srcrr-draft.pdf

[20] B. S. Manoj and R. R. Rao, Wireless Mesh Networks: Issues and Solutions, In: Y. Zhang, J. Luo, and H. Hu, Wireless Mesh Networking: Architectures, Protocols and Standards, Auerbach Publications,Ch. 1, pp. 3-48, New York, USA, 2007.

[21] http://www.nsnam.org/wiki/index.php/PyViz

The Analysis of Similarities and Registration Delay in Phonebook Centric Social Networks

Péter Ekler
Department of Automation and Applied Informatics
Budapest University of Technology and Economics
Magyar Tudósok Körútja 2., 1113 Budapest, Hungary
peter.ekler@aut.bme.hu

Tamás Lukovszki
Faculty of Informatics
Eötvös Loránd University
Pázmány Péter sétány 1/C, 1117 Budapest, Hungary
lukovszki@inf.elte.hu

***Abstract*—Phonebook centric social networks provide a synchronization mechanism between phonebooks of the users and the social network which allows detecting other users listed in the phonebooks. After that, if one of their contacts changes her or his personal detail, it will be propagated automatically into the phonebooks, after considering privacy settings. We participated in the implementation of a phonebook centric social network, called Phonebookmark and investigated the structure of the network. We used the data of this network for building the proposed models. In such social networks two entities may identify the same person if some parameters are similar, e.g.: phone number, address, etc. We call such entity pairs as similarities. Previously it was shown that the distribution of similarities follows a power law. Also a model was proposed by us, which can be used to estimate the total number of similarities, which is very important from scalability point of view in such networks. However the accuracy of the model is another question, because of the infinite variance of the power law distribution, which is used for modeling the number of similarities involving a user. The paper presents interesting and practical problem of analysis of similarities in social networks with application to mobile phonebooks management. The presented contribution includes both theoretical and practical components as well. We show that using the fact that a member of the network can only be involved in a limited number of similarities results in a similarity distribution with a finite variance. By using the central limit theorem we show the accuracy of our estimation. We also highlight that this model can be used in other power law distributions which apply to the requirements. Finally we also propose a performance model which can be used during the resource requirement design of such phonebook centric social networks.**

Keywords-component; social networks, mobile phones, power law distribution, variance, central limit theorem, queue model

I. INTRODUCTION

Nowadays social based websites, like social networks are becoming increasingly popular. These solutions not only available from web browsers but there are several existing mobile clients as well. These mobile applications are mainly simple clients to the network with some additional features. The phonebooks in the mobile devices represent social relationships that can be integrated in the social networks.

The relationship between social networks and mobile phones is noticeable as the popularity of such systems increase. In [1] we analyzed such networks from similarity handling point of view. In this paper we extend those results with important models and performance evaluation. Our new model allows a much more accurate prediction about the scalability of networks where the connections follow power law distributions.

In the last decade the internet related technologies developed rapidly. As reasons of this growth new type of solutions and applications have appeared. One of the most popular solutions are social network sites (SNS). Since their introduction, social network sites such as Facebook, MySpace and LinkedIn have attracted millions of users, many of whom have integrated these sites into their daily practices and they even visit these multiple times per day. These popular online social networks are among the top ten visited websites on the Internet [2]. End of 2010 it was reported that Facebook surpasses Google as number one U.S. site [3]. The basic idea behind such networks is that users can manage personal relationships online on these networks.

According to new statistics [4] Facebook has more than 750 million users, 50% of the active users log on to Facebook in any given day, more than 35 million users update their status each day and an average user spends more than 55 minutes per day on Facebook. Facebook began in early 2004 and the above statistics show that such popular social networks can have a huge growth which has to be considered during the design of any SNS.

Mobile phones and mobile applications are another hot topic nowadays. Facebook statistics also show that there are more than 65 million active users currently accessing Facebook through their mobile devices. People that use Facebook on their mobile devices are almost 50% more active on Facebook than non-mobile users. The increasing capabilities of mobile devices allow them to participate in social network applications as well. Mobile phone support in general social networks are usually limited mainly to photo and video upload capabilities and access to the social network using the mobile web browser.

However we should consider the fact, that the phonebook of the mobile device also describe the social relationships of its owner. Discovering additional relations in social networks is beneficial for sharing personal data or other content. Given an implementation that allows us to upload as well as download our contacts to and from the social networking application, we can completely keep our contacts synchronized so that we can see all of our contacts on the mobile phone as well as on the web interface. In addition to that if the system detects that some of our private contacts in the phonebook is similar to another registered members of the social network (i.e. may identify the same person), it can discover and suggest social relationships automatically. In the rest of this paper we refer to this solution as a *phonebook centric social network* (PCSN). Discovering and handling such similarities in phonebook centric social networks is a key issue. If a member changes some of her or his detail, it should be propagated in every phonebook to which she or he is related after considering privacy settings. In addition to that, with the help of detected similarities the system can keep the phonebooks always up-to-date.

Power law distribution is quite common in social networks and similar internet related graphs as measurements and examples show in Section 2. The number of similarities in phonebook centric social networks is very important from performance and scalability point of view.

We show that the distribution of similarities can be modeled with a random variable *X* with $\Pr[X \geq x] \sim cx^{-\alpha}$, if $x \leq n$ and $\Pr[X \geq x] = 0$ otherwise, where α>1 and *n* is a relevant upper bound.

As a main contribution of this paper, we show that the distribution of similarities has a finite variance which allows us to use the central limit theorem to prove the accuracy of our estimation of the total number of similarities. This model can be used generally in other similar distributions.

As a practical result, the concept of phonebook centric social networks was applied in the *Phonebookmark* project at Nokia Siemens Networks. *Phonebookmark* is a phonebook centric social network implementation by Nokia Siemens Networks. We took part in the implementation and before public introduction it was available for a group of general users from April to December of 2008. It had 420 registered members with more than 72000 private contacts, which is a suitable number for analyzing the behavior of the network. During this period we have collected and measured different type of data related to the social network.

The rest of the paper is organized as follows. Section 2 describes related work in the field of social networks and power law distributions. Section 3 introduces the structure of phonebook centric social networks. Section 4 summarizes our previously published model related to calculating the total number of similarities in the network. Section 5 states a general theorem related to the variance of power law distribution with relevant upper bound and uses it to prove the accuracy of the model described in Section 4. Section 6 shows that the total number of similarities is close to their expected value. Section 7 shows a performance model for calculating the expected queue length for similarity processing. We also show measurements related to Phonebookmark based on this model. The model can be used during the design of any different phonebook centric social networks. Finally, Section 8 concludes the paper and proposes further research plans.

II. RELATED WORK

In [5] the authors have defined social network sites (SNSs) as web-based services that allow individuals to (1) construct a public or semi-public profile within a bounded system, (2) articulate a list of other users with whom they share a connection, and (3) view and traverse their list of connections and those made by others within the system. The nature and nomenclature of these connections may vary from site to site.

According to this definition, the first recognizable social network site launched in 1997. SixDegrees.com allowed users to create profiles, list their Friends and, beginning in 1998, surf the Friends lists. Each of these features existed in some form before SixDegrees, of course. Profiles existed on most major dating sites and many community sites. AIM and ICQ buddy lists supported lists of Friends, although those Friends were not visible to others. Classmates.com allowed people to affiliate with their high school or college and surf the network for others who were also affiliated, but users could not create profiles or list Friends until years later. SixDegrees was the first to combine these features.

After that social networks have developed rapidly and the number of features increased. Nowadays most sites support the maintenance of pre-existing social networks, but others help strangers connect based on shared interests, political views, or activities. Some sites cater to diverse audiences, while others attract people based on common language or shared racial, sexual, religious, or nationality-based identities. Sites also vary in the extent to which they incorporate new information and communication tools, such as mobile connectivity, blogging, and photo/video-sharing.

As the functions of the SNSs flared, the number of users increased rapidly. Handling the extending number of users efficiently in SNSs is a key issue as it was visible in case of Friendster. Friendster was launched in 2002 as a social complement to Ryze. It was designed to help friends-of-friends meet, based on the assumption that friends-of-friends would make better romantic partners than would strangers. As Friendster's popularity surged, the site encountered technical and social difficulties. Friendster's servers and databases were ill-equipped to handle its rapid growth, and the site faltered regularly, frustrating users who replaced email with Friendster.

Huge amount of papers and popular books, such as Barabási’s Linked [6] study the structure and principles of dynamically evolving large scale networks like the Internet and networks of social interactions. Many features of social processes and the Internet are governed by power law distributions. Following the terminology in [7] a nonnegative random variable *X* is said to have a power law distribution if $\Pr[X \geq x] = cx^{-\alpha}$, for constant *c*>0 and α>0. In a power law distribution asymptotically the tails fall according to the power α, which leads to much heavier tails than other common models.

Distributions with an inverse polynomial tail have been first observed in 1897 by Pareto [8] (see. [9]), while describing the distribution of income in the population. In

1935 Zipf [10] and Yule [11] investigated the word frequencies in languages and based on empirical studies he stated that the frequency of the *n*-th frequent word is proportional to 1/*n*.

Mislove et al. [12] studied the graph properties of several online real-world social networks. Their paper presents a large-scale measurement study and analysis of the structure of multiple online social networks. They examined data gathered from four popular online social networks: Flickr, YouTube, LiveJournal, and Orkut. They crawled the publicly accessible user links on each site, obtaining a large portion of each social network's graph. Their data set contains over 11.3 million users and 328 million links. Their measurements show that high link symmetry implies indegree equals outdegree; users tend to receive as many links as the give, the observed networks are power law with high symmetry.

In [13], the graph structure of the Web has been investigated which also can be considered as a special variant of social network [14] and it was shown that the distribution of in- and out-degree of the Web graph and the size of weekly and strongly connected components are well approximated by power law distributions. Nazir et al. [15] showed that the in-and out-degree distribution of the interaction graph of the studied MySpace applications also follow such distributions.

There has been a great deal of theoretical work on designing random graph models that result in a Web-like graph. Barabási and Albert [16] describe the preferential attachment model, where the graph grows continuously by inserting nodes, where new node establishes a link to an older node with a probability which is proportional to the current degree of the older node. Bollobás et al. [17] analyze this process rigorously and show that the degree distribution of the resulting graph follow a power law. Another model based on a local optimization process is described by Fabrikant et al. [18]. Mitzenmacher [19] gives an excellent survey on the history and generative models for power law distributions. Aiello et al. [20] studies random graphs with power law degree distribution and derives interesting structural properties in such graphs.

Detecting similar or matching parameters of users is an important part in our phonebook centric social network. There is a huge amount of work for general similarity and match detection algorithms. According to [21] typical systems have an effectiveness (accuracy) of, at best, forty percent. A new measurement [22] showed that 55 percent of the first 20 records retrieved by Google Scholar are relevant. As shown in 3.3, the precision of Google Scholar remains relatively high even after the first 50 hits. Within the first 100 search results, 39 percent of GS records are relevant. Figure 3.3 also reveals that the utility of GS could be improved if relevant results were concentrated more heavily within the first 20 or 30 hits rather than the first 50 or 100.

The key difference between other works on online social networks and our work is that we extended social networks with mobile phone support and we discovered that the distribution of similarities follows power law. We proposed a model to estimate the number of similarities and despite the infinite variance of power law distribution we proved the accuracy of our model.

III. Structure of Phonebook centric Social Networks

Phonebook centric social networks are extending the well-known social network sites, they have a similar web user interface, but they add several major mobile phone related functions to the system. Following consider social networks as graphs. In case of general social networks, nodes are representing registered members and edges between them represent social relationships (e.g. friendship). After this we should notice that each member has a private mobile phone with a phonebook (Figure 1).

On Figure 1 we can see that phonebook contacts results new type of nodes in the graph representation and the edges between these private phonebook contacts and members represent which member “owns” those private contacts.

One of the key advantages of phonebook centric social networks is that they allow real synchronization between private phonebook contacts and the social network.

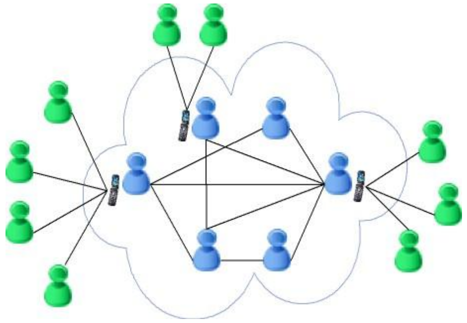


Figure 1. Phonebook-enabled social network

In order to enable such mechanism we need a similarity detecting algorithm. Such an algorithm is able to compare two person entries (members and private contacts, too) and determine how likely they represent the same person and propose a corresponding weight for the detected similarity.

Figure 2 represents the graph structure if the similarity detecting algorithm has finished comparing the relevant person entries.

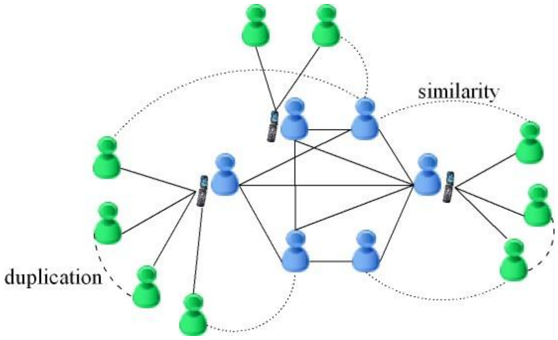


Figure 2. Detected similarities and duplications

On Figure 2 the dotted edges between member and private contacts represent detected similarities and broken lines between two private contacts illustrate possible duplications in the phonebooks. Duplications are detected as a positive side effect of the similarity detecting algorithm.

After similarities and duplications are detected there is a semi-automatic step, the members having private contacts in their phonebook, which are detected as similar to other members, have to decide whether the detected similarities are relevant ones, i.e.: accept or reject them. We call this step similarity resolution. In addition to that, members can also decide about the relevancy of detected duplications in their phonebooks. Figure 3 represents the graph structure after some of the members have resolved the detected similarities and duplication.

Besides that we can see on Figure 3 that four from the five similarities were accepted and there is still one in the system (the member has not checked it yet). Accepting a similarity means that a customized link edge is being formed between the private contact(s) in one's phonebook and the relevant member who represent the same person in the system. The private contacts that are linked to members via this type of customized links are called *customized contacts*.

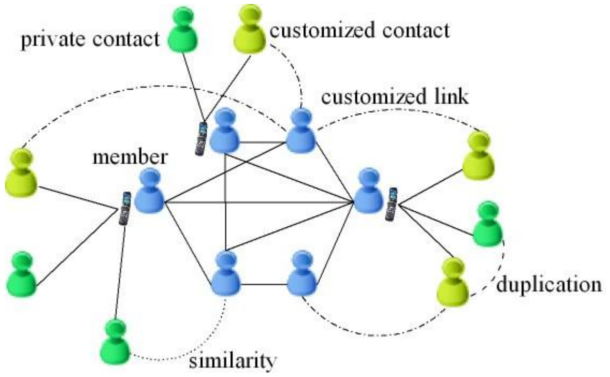


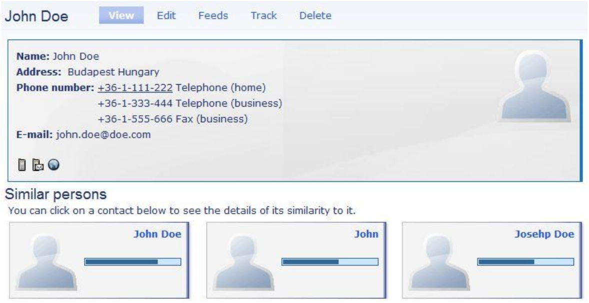
Figure 3. Resolving similarities and duplications

One of the key advantages of phonebook centric social networks are these customized links, because if a member changes his personal detail on the web user interface (adds a new phone number, uploads a new image, changes the website address, etc.) it will be automatically propagated to those phonebooks where there is a customized contact related to this member. Additional important advantages of phonebook centric social networks are:

- Private contacts can be managed (list, view, edit, call, etc.) from a browser.
- Similarity detecting algorithm realizes the user if duplicate contacts are detected in its phonebook and warns about it.
- Private contacts are safely backed up in case the phone gets lost.
- Private contacts can be easily transferred to a new phone if the user replaces the old one.
- Phonebooks can be shared between multiple phones, if one happens to use more than one phone.
- It is not necessary to explicitly search for the friends in the service, because it notices if there are members similar to the private contacts in the phonebooks and warns about it.

The detailed structure and edge rule definition was described in [23].

In [24] we have introduced a phonebook-centric social network implementation, called Phonebookmark. Phonebookmark provides a semi-automatic similarity detecting and resolving mechanism. First it detects similarities and calculates a similarity weight for them, which indicates, how likely the entries identify the same person. (Figure 4).



John Doe

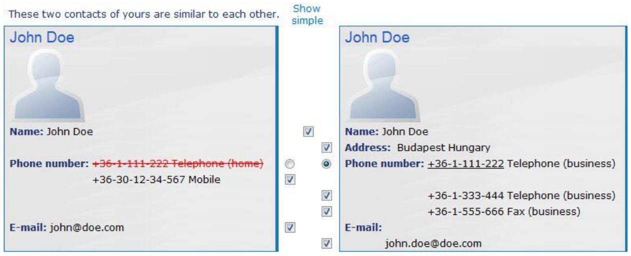
Name: John Doe
Address: Budapest Hungary
Phone number: +36-1-111-222 Telephone (home)
+36-1-333-444 Telephone (business)
+36-1-555-666 Fax (business)
E-mail: john.doe@doe.com

Similar persons
You can click on a contact below to see the details of its similarity to it.

John Doe
John
Josehp Doe

Figure 4. Dealing with multiple similarities

After a detected similarity is being selected, Phonebookmark provides a user interface where the details of the two people can be merged. Here the user can choose whether to accept or reject the similarity, which is the base of the semi-automatic behavior (Figure 5).



These two contacts of yours are similar to each other.

John Doe
Name: John Doe ☑
Phone number: ~~+36-1-111-222 Telephone (home)~~ ○
+36-30-12-34-567 Mobile ☑
E-mail: john@doe.com ☑

John Doe
☑ Name: John Doe
☑ Address: Budapest Hungary
● Phone number: +36-1-111-222 Telephone (business)
☑ +36-1-333-444 Telephone (business)
☑ +36-1-555-666 Fax (business)
E-mail:
☑ john.doe@doe.com

Figure 5. Semi-automatic similarity resolution

IV. Number of Similarities

We model the number of similarities generated during a member registration by a random variable *X*. More precisely, *X* models the number of similarities proposed by the automatic similarity detection algorithm. In [22] we showed that *X* can be well approximated by a power law distribution. Using this model we gave estimation on the total number of similarities in the system. Now we summarize this model.

The total number of accepted similarities N_S in a phonebook centric social network can be estimated with the following formula:

$$N_S = NE[X]P_R, \tag{3}$$

where N is the number of registered members and P_R is the rate of the similarities accepted by the users. Measurements in [22] showed that P_R can be approximated with 0.9. In order to estimate *E[X]*, we need the probabilities *Pr[X=x]*, which can be obtained from the complementary cumulative distribution function $\Pr[X \geq x] \sim cx^{-\alpha}$ by derivation:

$$\Pr[X = x] \sim c' x^{-(\alpha+1)}. \tag{4}$$

In order to be a probability distribution, $\sum_{x=1}^{\infty} c' x^{-(\alpha+1)} = 1$. Note, that x starts from one, because a new member registration involves at least one similarity, because the system allows registration only by invitation. Therefore, the new member is already in the phonebook of the inviting member. Thus, $c'=1/\zeta(\alpha+1)$, where $\zeta(.)$ denotes the Riemann Zeta function. Then the expected value is:

$$\begin{aligned} E[X] &= \sum_{x=1}^{\infty} x \Pr[X = x] \\ &= \sum_{x=1}^{\infty} x \frac{1}{\varsigma(\alpha+1)} x^{-(\alpha+1)} \\ &= \frac{1}{\varsigma(\alpha+1)} \sum_{x=1}^{\infty} x^{-\alpha} = \frac{\varsigma(\alpha)}{\varsigma(\alpha+1)}. \end{aligned} \tag{5}$$

The expected total number of accepted similarities N_S in a phonebook centric social network can be estimated with the following formula:

$$N_S = N_M \frac{\varsigma(\alpha)}{\varsigma(\alpha+1)} P_R. \tag{6}$$

For $\alpha>1$, $\zeta(\alpha)/\zeta(\alpha+1)$ is a finite constant. In our case, for $\alpha=1.276$, we obtain that the expected total number of similarities is

$$N_S = 2.9196 * 420 * 0.9 = 1103. \tag{7}$$

However in this model the X random variable has power law distribution which has infinite variance thus the accuracy of this model is an issue. In the next section we show how to prove the accuracy of this model by stating and a general theorem related to the variance of power law distributions with relevant upper bound.

V. Variance Model for Power Law Distribution with Upper Bound

For $\alpha\leq2$, a power law distribution has infinite variance, which prevents to apply the central limit theorem in order to obtain that the total number of similarities will be close to their expected value. However we can use the following fact

Fact: If the phonebooks do not contain duplicates then the number of similarities caused by a member is at most *2(N-1)* [23].

With other words, in the interval *[0,2(N-1)]* the distribution of similarities follows a power law and the probability of higher similarities is zero. In order to see this, note that a member u can be similar to at most one private contact of each of the other *N-1* members and, for each private contact of u, there is at most one similar member in the network.

We show that the distribution of similarities resulting from this fact has a finite variance. This allows us to use the central limit theorem to prove the accuracy of our estimation of the total number of similarities in Section 4.

Theorem 1: Let X be a random variable with $\Pr[X = x] = cx^{-\beta}$ if $x \leq n$ and $\Pr[X = x] = 0$ otherwise, where $\beta=\alpha+1$, $2<\beta<3$. In this case the variance can be estimated with $\sigma^2 X = \Theta\left(n^{3-\beta}\right)$.

For the proof we used two lemmata.

Lemma 1: Let X be a random variable with $\Pr[X = x] = cx^{-\beta}$ if $x \leq n$ and $\Pr[X = x] = 0$ otherwise, where $\beta=\alpha+1$, $2<\beta<3$. In this case the var**i**ance is $\sigma^2 X = \mathrm{O}\left(n^{3-\beta}\right)$.

Proof: From the Steiner formula, the variance is estimated as $\sigma^2 X = E[X^2] - (E[X])^2$. *E[X]* was defined previously, thus we need to estimate only the *E[X²]*. By definition:

$$\begin{aligned} E[X^2] &= \sum_{x=1}^{\infty} x^2 \Pr[X = x] \\ &= \sum_{x=1}^{n} x^2 c \frac{1}{x^{\beta}}. \end{aligned} \tag{8}$$

Now we can apply that n is an upper bound on the value of X. This way (1) can be followed as:

$$E[X^2] = c \sum_{x=1}^{n} x^{2-\beta}.$$

$$\text{Let } y = \frac{1}{c} E[X^2]. \tag{9}$$

Following we show an upper estimation for y. In order to do so we create an upper model for the function of y by using the powers of *1/2*. Let $z = 2^{\frac{1}{\beta-2}}$, then

$$z^{2i} \Pr[X = z^i] = (z^i)^{2-\beta} = \left(2^{\frac{1}{\beta-2}i}\right)^{2-\beta} = 2^{-i} \tag{10}$$

Figure 6 illustrates how we performed the estimation, with the *f1* function.

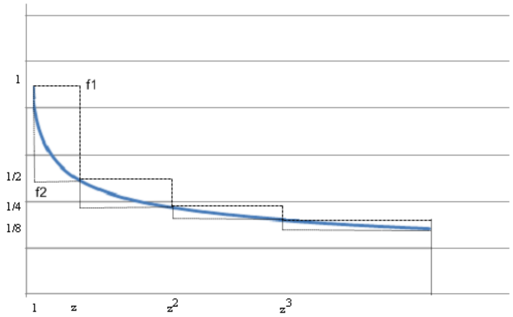


Figure 6. Staged estimation function

Now we are able to approximate y from top:

$$\begin{aligned}
y &\le \sum_{i=0}^{\log_z n}\left(z^{i+1}-z^i\right)\left(z^i\right)^{2-\beta}\\
&= \sum_{i=0}^{\log_z n}\left(z^{i+1}-z^i\right)2^{-i}\\
&= \sum_{i=0}^{\log_z n}(z-1)z^i 2^{-i}\\
&= (z-1)\sum_{i=0}^{\log_z n}\left(\frac{z}{2}\right)^i\\
&= (z-1)\left(\frac{\left(\frac{z}{2}\right)^{\log_z n+1}-1}{\frac{z}{2}-1}\right)\\
&= (z-1)\left(\frac{\frac{z}{2}n^{\frac{1}{\log_{z/2} z}}-1}{\frac{z}{2}-1}\right)\\
&= (z-1)\left(\frac{\frac{z}{2}n^{\frac{1}{1+\log_{z/2} 2}}-1}{\frac{z}{2}-1}\right).
\end{aligned} \tag{12}$$

The explanation to the last step:

$$\log_{z/2} z = \log_{z/2} 2\frac{z}{2} = 1+\log_{z/2} 2 \tag{13}$$

To continue, first we have to check the following calculation. Remember that z was described with β and $\beta=\alpha+1$. This way:

$$\log_{z/2} 2 = \frac{\log_2 2}{\log_2 z/2} = \frac{1}{\log_2\left(\frac{2^{\frac{1}{\beta-2}}}{2}\right)} = \frac{1}{\frac{1}{\beta-2}-1} = \frac{\beta-2}{3-\beta}. \tag{14}$$

Therefore:

$$n^{\frac{1}{1+\log_{z/2} 2}} = n^{\frac{1}{1+\frac{\beta-2}{3-\beta}}} = n^{3-\beta}. \tag{15}$$

This way (2) looks as follows:

$$y \le (z-1)\left(\frac{\frac{z}{2}n^{\beta-3}-1}{\frac{z}{2}-1}\right). \tag{16}$$

Next we show that the variance by applying the Steiner formula and the previous calculations is $O(n^{3-\beta})$:

$$\begin{aligned}
\sigma^2 X &= E[X^2]-(E[X])^2 = cy-\Theta(1)\\
&= c(z-1)\left(\frac{\frac{z}{2}n^{\beta-3}-1}{\frac{z}{2}-1}\right)-\Theta(1)\\
&\le c\left(\frac{(z-1)z}{z-2}n^{3-\beta}\right)-\Theta(1)\\
&= O\left(n^{3-\beta}\right).
\end{aligned} \tag{17}$$

□

Lemma 2: Let X be a random variable with $\Pr[X=x]=cx^{-\beta}$ if $x\le n$ and $\Pr[X=x]=0$ otherwise, where $\beta=\alpha+1$, $2<\beta<3$. In this case the variance is $\sigma^2 X=\Omega\left(n^{3-\beta}\right)$.

Proof: Similarly to *Lemma 1* if we give a lower bound on y using function *f2* is shown on Figure 6:

$$y \ge \sum_{i=0}^{\log_z n}\left(z^{i+1}-z^i\right)2^{-(i+1)}, \tag{18}$$

we obtain that $\sigma^2 X=\Omega\left(n^{3-\beta}\right)$. □

Proof (of Theorem 1) The proof is straightforward by applying Lemma 1 and 2:

$\sigma^2 X=\Theta\left(n^{3-\beta}\right)$, because $\sigma^2 X=\Omega\left(n^{3-\beta}\right)$ and $\sigma^2 X=O\left(n^{3-\beta}\right)$

□

In our case the upper bound n to the total number of similarities is *2(N-1)*.

VI. Applying Central Limit Theorem for the Distribution of Similarities

Following we show that the total number of similarities are close to their expected value.

Theorem 2: Let *N* be the number of members in a phonebook centric social network and $S_N=X_1+X_2+...+X_N$ where X_i , $i=1,...,N$, is a random variable representing the number of similarities raised by member *i*, i.e. $\Pr[X_i = x] = \kappa x^{-\beta}$ if $x \le n$ and $\Pr[X_i = x] = 0$ otherwise, where $n=\Theta(N)$, $\beta=\alpha+1$, $2<\beta<3$, and κ is a constant. Let $\mu = E[X_i]$. Then

$$\Pr[S_N \ge c\mu N] \approx 1-\Phi(m),$$

where $m = a(c-1)\mu N^{\frac{\beta}{2}-1}$ and *a* is an appropriate constant.

Proof: X_1, X_2, …, X_N are *N* independent and identically distributed random variables, each having finite values of expectation μ and variance $\sigma^2>0$. The central limit theorem states that the distribution of the sample average of these random variables approaches the normal distribution with a mean μ and variance σ^2/n. Let

$$Z_N = \frac{S_N - N\mu}{\sigma\sqrt{N}} \qquad (19)$$

By the central limit theorem, the distribution of Z_N approaches the standard normal distribution:

$$Z_N \to \Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{\frac{-t^2}{2}} dt \qquad (20)$$

The density function looks as follows:

$$f(x) = \frac{1}{\sqrt{2\pi}} e^{\frac{-x^2}{2}} \qquad (21)$$

Now we determine the probability that S_N is greater or equal than *c* times of its expected value, for a constant $c>1$. For $S_N= cE[S_N]=c\mu N$ then

$$\begin{aligned} Z_N &= \frac{c\mu N - N\mu}{\sigma\sqrt{N}} \\ &= \frac{(c-1)N\mu}{\Theta(\sqrt{N^{3-\beta}})\sqrt{N}} \\ &= \frac{(c-1)N\mu}{\Theta(\sqrt{N^{4-\beta}})} \\ &= \frac{(c-1)\mu}{\Theta\left(N^{1-\frac{\beta}{2}}\right)} \\ &\ge a(c-1)\mu N^{\frac{\beta}{2}-1}, \end{aligned} \qquad (22)$$

where *a* is an appropriate constant. Therefore:

$$\Pr\left[S_N \ge c\mu N\right] \le \Pr\left[Z_N \ge a(c-1)\mu N^{\frac{\beta}{2}-1}\right]. \qquad (23)$$

Let $m = a(c-1)\mu N^{\frac{\beta}{2}-1}$. Since, by the central limit theorem, the distribution of Z_N can be approximated by the standard normal distribution, we have

$$\begin{aligned} &\Pr\left[S_N \ge c\mu N\right] \le \Pr\left[Z_N \ge m\right] \\ &\approx 1-\Phi(m). \end{aligned} \qquad (24)$$

□

Theorem 3: For $m = a(c-1)\mu N^{\frac{\beta}{2}-1}$:

$$1-\Phi(m) \le \frac{1}{2\sqrt{\pi}} e^{-\gamma N^{\beta-2}},$$

where $\gamma = (a(c\text{-}1)\mu)^2/2$ is a constant.

Proof:

$$\begin{aligned} 1-\Phi(m) &= 1 - \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{m} e^{\frac{-t^2}{2}} dt \\ &= \frac{1}{\sqrt{2\pi}} \int_{m}^{\infty} e^{\frac{-t^2}{2}} dt \\ &= \frac{1}{2} erfc\left(\frac{m}{\sqrt{2}}\right) \end{aligned} \qquad (25)$$

$$= \frac{1}{2}\frac{\Gamma\left(\frac{1}{2},\left(\frac{m}{\sqrt{2}}\right)^2\right)}{\sqrt{\pi}},$$

where *Γ(a,x)* is the incomplete gamma function.

$$\Gamma(a,x) = \int_x^\infty t^{a-1}e^{-t}dt. \tag{26}$$

For an integer *r*:

$$\Gamma(r,x) = (r-1)!e^{-x}\sum_{k=0}^{r-1}\frac{x^k}{k!}. \tag{27}$$

Because, for $x \geq 1/2$, $\Gamma\left(\frac{1}{2},x\right) \leq \Gamma(1,x)$, for $m \geq 1$:

$$\begin{aligned} 1-\Phi(m) &= \frac{1}{2}\frac{\Gamma\left(\frac{1}{2},\left(\frac{m}{\sqrt{2}}\right)^2\right)}{\sqrt{\pi}} \\ &\leq \frac{1}{2}\frac{\Gamma\left(1,\left(\frac{m}{\sqrt{2}}\right)^2\right)}{\sqrt{\pi}} \\ &= \frac{1}{2\sqrt{\pi}}e^{-\frac{m^2}{2}} \\ &= \frac{1}{2\sqrt{\pi}}e^{-\gamma N^{\beta-2}}, \end{aligned} \tag{28}$$

where $\gamma = (a(c-1)\mu)^2/2$ is a constant. □

VII. MODELING PROCESSING TIME FOR SIMILARITIES

As we have highlighted, similarity detecting and handling is a key issue in phonebook centric social networks. First the similarity algorithm has to find similar persons then handle them properly. Phonebookmark uses a semi-automatic similarity resolving mechanism. First it detects similarities and calculates a probability for them, which indicates how likely the corresponding phonebook contact and the member of the network identify the same person. This detecting algorithm runs in the background continuously on server side and it has to scan the members of the network at registration or synchronization events. In case of multiple similarities, Phonebookmark uses the similarity probability values to determine the proper order. The details of the algorithm are discussed in [25].

The behavior of the similarity detecting algorithm is similar to a queuing system where the processing unit is the algorithm and the entities in the queue are the person pairs which are waiting for comparison. The responsiveness of the algorithm is critical as similarity handling is a key feature of phonebook centric social networks compared to other solutions.

In the following model we consider only the registration operation, since it can bring the most similarity, because a totally new phonebook is being uploaded in the system. This operation can be divided for two main tasks. Firstly, when a member registers, she or he should be compared to every phonebook contact in all phonebooks present in the network. If we consider the number of private contacts in a phonebook as a random variable X_{Pc}, this means $E[X_{Pc}]*N$ comparisons, where $E[X_{Pc}]$ is the expected value of the phonebook sizes, *N* is the number of members in the network before the registration and *PC* denotes to private contacts. After the initial state of the social network, when the number members *N* is high, it can be considered as a relative constant value in one processing step of the queue model.

Based on the database of Phonebookmark we were able to estimate the distribution of phonebook sizes. Figure 6 shows the tail distribution of the phonebook-sizes such that the *x*-axis has linear scale and the *y*-axis logarithmic scale. The points on this figure fit very well to a line, which means that the tail of the phonebook sizes decreases exponentially. This provides a simple empirical test for whether a random variable has an exponential distribution. In this case the gradient of the function gives the parameter of the exponential distribution (Figure 6).

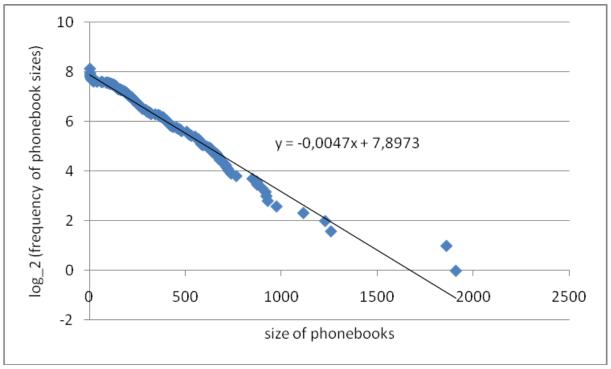


Figure 6. Size of phonebooks in Phonebookmark

In this measurement this parameter is *0.0047*, the expected value of the exponential distribution can be calculated as the reciprocal of this parameter, thus the expected value of phonebook sizes according to this measurement is *212*. Following we refer to $E[X_{Pc}]$ as *C*. This shows that the phonebook sizes can be modeled very well with an exponential distribution.

The other task during the member registration is to check, which members of the network are in the phonebook of the new member. This task requires $N*X_{Pc}$ comparisons, where the size of the new phonebook is modeled also with exponential distribution.

This way the amount of comparisons required by a member registration is modeled with the random variable X^*_{Pc}:

$$X^*_{Pc} = C * N + X_{Pc} * N = N * (C + X_{Pc}) \qquad (29)$$

Following we show that X^*_{Pc} has exponential distribution.

Lemma 4: X^*_{Pc} random variable has exponential distribution.

Proof:

Because of the linear transformation, the distribution function of X^*_{Pc} looks as follows:

$$F_{X^*_{Pc}}(x^*) = F_{X_{Pc}}\left(\frac{x^* - N * C}{N}\right), \qquad (30)$$

when N>0, which is always true in our case. This way since the distribution of X_{Pc} and X^*_{Pc} looks the same, X^*_{Pc} has also exponential distribution. □

We model the registration rate of members as a Poisson process with λ parameter and we assume that a person pair comparison is the time unit.

Theorem 4: In order to keep the stability of the similarity detecting the following is required for the rate of member arrival:

$$\lambda < \frac{1}{2CN}.$$

Proof: According to Kleinrock's model for queuing systems (Section 3.2 in [26]), when the arrival rate is modeled with a λ parameter Poisson distribution and the processing with exponential distribution with ν parameter then the requirement for stability:

$$\frac{\lambda}{\nu} < 1. \qquad (31)$$

This means that the expected value of serving time (*1/ν*) is smaller than the expected value of time between arrivals (*1/λ*). In our case the expected value of the serving time is E[X^*_{Pc}], since we considered a person pair comparison as the time unit. By applying *Lemma 4* we can see that X^*_{Pc} has an exponential distribution and the expected value of it is calculated by:

$$E[X^*_{Pc}] = E[CN + X_{Pc}N] = \\ CN + NE[X_{Pc}] = 2CN. \qquad (32)$$

In case of exponential distributions, the reciprocal of the expected value is the λ parameter of the distribution. This way the requirement of the stability looks as follows:

$$\frac{\lambda}{\frac{1}{2CN}} < 1, \qquad (33)$$

$$\lambda < \frac{1}{2CN}$$

□

This way the average number of person pairs Q waiting for comparison can be calculated (Section 3.2 in [26]) with:

$$Q = \frac{\lambda}{\frac{1}{2CN} - \lambda} \qquad (34)$$

Based on this model, the resource requirement of the similarity detecting can be calculated in real environment, considering the speed of the processing unit(s). In order to demonstrate the behavior of this queue, we have made measurements regarding to the registration of the members in Phonebookmark.

Figure 7 illustrates the queue length considering $2C*N$ and $2.5C*N$ processed person-pair comparison in one step.

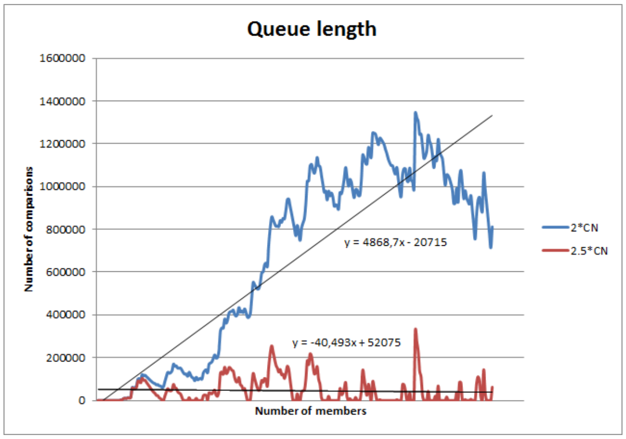


Figure 7. Queue length for similarity calculation

The *x*-axis shows as the number of members in the system increases, while the *y*-axis represents the number of comparison steps when a new member registers (sum of the remaining comparison and the new ones). It can be seen that the average queue length can be decreased significantly, when the processing speed increases.

Figure 8 illustrates the queue length normalized with the number of members.

VIII. Conclusion and Future Work

Social network sites are becoming more and more important in everyday life. Phonebook centric social networks enable to manage online and mobile relationships within one system.

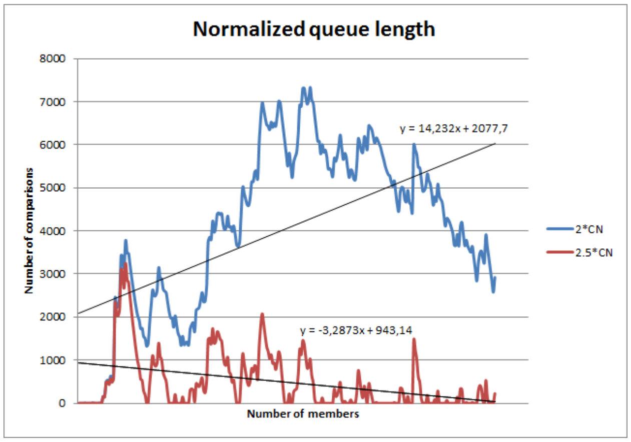


Figure 8. Normalized queue length for similarity calculation

The key mechanism of such networks is a similarity handling algorithm which detects similarities between members of the network and phonebook entries.

The number of similarities is a key parameter from scalability point of view. In our previous research we have shown how to estimate the expected number of similarities [23]. In order to show the accuracy of this model, in this paper we proved that, the distribution of similarities has a finite variance (Section V). This model can be used generally in other similar distributions.

After that, as the variance is finite, we applied central limit theorem to examine the accuracy of our estimation of the total number of similarities. We showed that the total number of similarities is close to their expected value. As a future work, the estimation, stated in Theorem 2, can be refined by taking the speed of the convergence to the limit distribution into account.

Finally we showed that in order to ensure the responsiveness of the network the similarity detecting should work quickly. We have given a queue based model for similarity detecting and we have shown how to calculate the expected queue length, assuming Poisson arrival for member registration. The results can be applied also for the resource requirement in different social networks providing synchronization with an external contact list.

Acknowledgments

This project is supported by the New Hungary Development Plan (Project ID: TÁMOP-4.2.1/B-09/1/KMR-2010-0002 and TÁMOP-4.2.1/B-09/1/KMR-2010-0003).

IX. References

[1] P. Ekler, T. Lukovszki. *The Accuracy of Power Law based Similarity Model in Phonebook-centric Social Networks*. In: 6th International Conference on Wireless and Mobile Communications (ICWMC). 2010.

[2] Alexa. http://www.alexa.com/topsites. February 2010.

[3] Comcast. http://www.comcast.net/articles/finance/20101230/BUSINESS-US-FACEBOOK-GOOGLE/, December 2010.

[4] Facebook statistics, http://www.facebook.com/press/info.php?statistics, February, 2010.

[5] D. M. Boyd, N. B. Ellison, *Social network sites: Definition, history, and scholarship*, Journal of Computer-Mediated Communication, Volume 13, Issue 1 (2007)

[6] A.-L. Barabási, R. Albert. *Emergence and scaling in random networks*. *Science*, Vol. 286, paged: 509-512, 1999.

[7] A. Fabrikant, E. Koutsoupias, and C. H. Papadimitriou. *Heuristically Optimized Trade-offs: A New Paradigm for Power Laws in the Internet*. In *Proc. of ICALP*, pages: 110-122, 2002.

[8] V. Pareto. *Course d'economie politique professé à l'université de Lausanne*, 3 volumes, 1896-7.

[9] M. Mitzenmacher. *A brief history of generative models for power law and lognormal distributions*. *Internet Mathematics*, Vol. 1, pages: 225-251, 2001.

[10] G. K. Zipf. *The Psycho-Biology of Language. An Introduction to Dynamic Philology*. Houghton Mifflin, Boston, MA, 1935.

[11] G. U. Yule. *Statistical study of literary vocabulary*, Cambridge University Press, 1944.

[12] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee. *Measurement and analysis of online social networks*. In ACM/USENIX IMC, 2007.

[13] A. Broder, R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, A. Tomkins, and J. Wiener. *Graph structure in the web*. In Proc. of the 9th international World Wide Web conference on Computer networks, 2000.

[14] F. Chierichetti, R. Kumar, S. Lattanzi, M. Mitzenmacher, A. Panconesi, P. Raghavan, On Compressing Social Networks, In: Proc. of the 15th ACM SIGKDD International Conference Knowledge Discovery and Data Mining (KDD'09), 2009.

[15] Nazir, S. Raza and C.-N. Chuah. *Unveiling Facebook: A measurement Study of Social Network Based Applications*. In: Proc. ACM Internet Measurement Conference (IMC), 2008.

[16] A.-L. Barabási, R. Albert. Emergence and scaling in random networks. *Science*, Vol. 286, 509-512, 1999.

[17] B. Bollobás, O. Riordan, J. Spencer, G. Tusnady. The degree sequence of a scale-free random graph process. *Random Structures and Algorithms*, Vol. 18(3), 279-290, 2001.

[18] A. Fabrikant, E. Koutsoupias, and C. H. Papadimitriou. Heuristically Optimized Trade-offs: A New Paradigm for Power Laws in the Internet. In: *Proc. 29th International Colloquium on Automata, Languages and Programming (ICALP)*, 110-122, 2002.

[19] M. Mitzenmacher. *A brief history of generative models for power law and lognormal distributions*. Internet Mathematics, Vol. 1, 225-251, 2001.

[20] W. Aiello, F. R. K. Chung, L. Lu. *A random graph model for massive graphs*. In: *Proc. 32nd Symposium on Theory of Computing STOC*, 171-180, 2000.

[21] Nist special publication 500-255. In The Twelfth Text rEtrieval Conference (TREC 2003).

[22] Walters and H. W., Google scholar search performance: Comparative recall and precision libraries and the academy, volume 9/1, January, 2009.

[23] P. Ekler, T. Lukovszki. *Similarity Distribution in Phonebook-centric Social Networks*. In: 5th International Conference on Wireless and Mobile Communications (ICWMC). 2009.

[24] P. Ekler, T. Lukovszki. *Experiences with phonebook-centric social networks*. In: CCNC'10, Las Vegas. 2010.

[25] Péter Ekler, Tamás Lukovszki, *Learning Methods for Similarity Handling in Phonebook-centric Social Networks*, 10th International Symposium of Hungarian Researchers on Computational Intelligence and Informatics (CINTI 2009), 2009.

[26] L. Kleinrock, *Queueing Systems. Volume 1: Theory*, Wiley-Interscience, pages 94-101, 1975.

WLAN IEEE 802.11a/b/g/n Indoor Coverage and Interference Performance Study

Sandra Sendra[1], Miguel Garcia[2], Carlos Turro[3], Jaime Lloret[4]
Universidad Politécnica de Valencia
Camino Vera s/n, 46022, Valencia, Spain
[1]sansenco@posgrado.upv.es, [2]migarpi@posgrado.upv.es, [3]turro@cc.upv.es, [4]jlloret@dcom.upv.es

Abstract—**An adequate wireless network plan is needed to replace the traditional wired LANs. A full coverage WLAN offers the flexibility to relocate people and equipment or to reconfigure and add more wireless devices to the network. Usually, an IEEE 802.11 variant is chosen based on their bandwidth and their coverage area. However, sometimes there are special cases where the best technology is not the newest one. In addition, suitable positioning of access points (AP) is crucial to determine the efficiency of the network. E.g. in the case where devices are going to transmit at a maximum of 1 Mbps, any choice is acceptable, but when it is required higher performance, other factors must be considered. In this paper, we compare IEEE 802.11a/b/g/n indoor environments to know what technology is better. This comparison will be taken in terms of RSSI, coverage area, and measuring the interferences between channels. These key factors must be optimum to have high performance in the WLAN. This study will help the researchers to choose the best technology depending of their deploying case, and we will see study the best variant for indoors.**

Keywords-WLAN; IEEE 802.11; Coverage; Interferences; Performance measurements.

I. INTRODUCTION

This paper is an extended version of the paper presented by S. Sendra et al. in [1].

One of the major issues in Wireless Local Area Network (WLAN) indoor environments is the multipath dispersion due to the influence of many signal reflectors and diffusions. Walls, floors and roofs attenuate the signal highly and provoke great variations in the mean received power. Even the furniture and the metallic structures of the walls and roofs have high impact because they enhance the scattering and diffraction. There has been many studies about the signal propagation in indoors [2][3]. Moreover, there are special challenges when designing WLANs in indoors [4].

Because the emitter and the receiver are close, the delay between echoes will enlarge the delay spread. But, temporal variations are slower because of the low mobility of the users. Temporal variations are mainly given by the presence of humans close to the antennas. Moreover, there are other features in indoor environments such as:

- Electromagnetic fields provided by electronic devices. Although the reflection and diffraction can be modeled, there are many things inside the building that introduce a certain grade of variability [3].
- Usually people walking in any corridor or facility close to the emitter or the receiver will cause significant variations [5].
- Because the distances are short, any variation of the direction of the antenna will imply high changes in the signal received.
- Metallic objects reflect the radio signal. The signal will not cross metallic walls and metallic objects will fade.
- Wood, crystal, plastic and bricks reflect part of the signal, but let pass the rest.
- The objects with high humidity have more signal absorption.

There are several indoor propagation models. They can be classified in empirical models (which are based on the measures taken and predict the signal loss), in deterministic models (that simulate the signal propagation in order to characterize the transmission channel), theoretical models, (which are based in the physical laws of the modeled medium) and stochastic models (they are modes which results have a probability distribution) [6]. The appropriate model must be chosen based in the design necessities. Empirical models are used in network design, while deterministic models are used for high precision applications. The first ones are less complex and need lower input parameters, but they do not predict instantaneous signal fainting [7].

The most well known models are the following ones:

- Log-Normal Shadowing Path Loss Model [8]
- Loss Model based in COST 231 [9]
- Linear Path Attenuation Model [10]
- Keenan-Motley Model [11]
- ITU-R Model [12]
- Dual Slope-Model [13]
- Multi-Wall Model [14]

Several authors have studied empirically each one of them providing their drawbacks and benefits.

But, when we are setting up a WLAN, it is not practical to model all wireless coverage area for each site where the access point is going to be placed, especially when we are talking about large extension areas [15]. Within the IEEE 802.11 standard [16], there are included several variants like IEEE 802.11a, IEEE 802.11b, IEEE 802.11g and IEEE 802.11n. All of them provide different coverage areas, and even different signal strength inside the coverage area.

The standard uses the CSMA/CA protocol as the medium access method. It is a carrier sense multiple access with collision avoidance used to avoid collisions between wireless data packets. In Europe, the frequency ranges from 2.401 to 2.483 GHz is divided into 13 channels of 22 MHz wide, and spaced 5 MHz between them, where channel 1 is centered on 2.412 GHz and the channel 13 is located at 2.472 GHz.

Japan adds an additional channel 14, located 12 MHz above channel 13. In an IEEE 802.11 network, the participant significantly reduces the speed of the overall wireless network. Now we are going to introduce each IEEE 802.11 variant [17, 18].

A. *IEEE 802.11a*

IEEE 802.11a was approved in 1999. Although it was born in 1999, it did not begin to be marketed until 2001. This variant works in the 5 GHz band. Its architecture is based on two types of devices: the Access Points (APs), which are the base station for the wireless network, and the wireless clients, that can be mobile devices such as laptops, PDAs, and fixed devices such as desktops and workstations equipped with a wireless network interface.

IEEE 802.11a uses OFDM (Orthogonal Frequency Division Multiplexing) modulation with 52 subcarriers. This standard has a theoretical maximum speed of 54 Mbps, but the transmission rate decreases when the signal quality decreases. 54 Mbps could be changed to 48, 36, 24, 12, 9 and 6 Mbps. There are 52 subcarriers, 48 of them are used for the data transmission and 4 for pilot tasks, with a separation of 312.5 KHz. Each subcarrier may be modulated by BPSK (Binary Phase Shift Keying), QPSK (Quaternary Phase Shift Keying), 16-QAM (Quadrature Amplitude Modulation) or 64-QAM.

IEEE 802.11a provides 12 non-overlapping channels. As it uses the 5 GHz band, the signal has less interference than the other standards IEEE 802.11. But the equipment must be in the line of sight (LOS) to gain a better efficiency in communications.

B. *IEEE 802.11b*

The 802.11b standard was approved in 1999. IEEE 802.11b data are encoded using the Direct Sequence Spread Spectrum Signal (DSSS). This technology uses CCK (Complementary Code Keying) and QPSK modulation to achieve a maximum transfer raw rate of 11 Mbps. However, it cannot exceed 6 Mbps with TCP (Transmission Control Protocol) and 7 Mbps with UDP (User Datagram Protocol) theoretically.

The protocol can be used in point-to-multipoint or point-to-point topology with links over distances proportional to the features of the antennas and output power. Furthermore, if there is any problem with the signal quality, it is possible to transmit in 5.5, 2 or 1 Mbps, using redundant methods of data encryption.

First devices appeared very quickly because this variant was an extension to the DSSS modulation of the original standard. The higher speed and the low cost of the devices achieved a fast growth of this technology in the market.

C. *IEEE 802.11g*

IEEE 802.11g appeared in 2003. It is an evolution of IEEE 802.11b. It works on 2.4 GHz frequency band and it is compatible with IEEE 802.11b. Its theoretical transfer is 54 Mbps, although it is reduced to 22 Mbps when the receiver is some meters far from the AP in a real scenario. It uses 52 subcarriers.

The modulation scheme used in 802.11g is orthogonal frequency-division multiplexing (OFDM), such as in 802.11a, with data rates of 6, 9, 12, 18, 24, 36, 48, and 54 Mbps. It reverts to CCK (like the 802.11b standard) for 5.5 and 11 Mbps and to DBPSK and DQPSK + DSSS for 1 and 2 Mbps respectively. In this standard, there is also a speed decrease according to the signal quality.

IEEE 802.11g suffers from the same interference as IEEE 802.11b in the already crowded 2.4 GHz range.

Because IEEE 802.11g uses the same radio signaling (CCK) as 802.11b, at the lower four IEEE 802.11g data rates, it is fully backward compatible with IEEE 802.11b. This enables IEEE 802.11b/g wireless networks to continue supporting only IEEE 802.11b enabled devices. IEEE 802.11g may seem to be the competence of 802.11a, but most products include both technologies because they are complementary.

D. *IEEE 802.11n*

While IEEE 802.11a/b/g WLANs provide adequate performance for today's networking applications, the wireless applications of next generation require higher data throughput and bigger coverage area. This variant sought to bring the transmission capacity of wireless data transmission at speeds of wired systems.

IEEE 802.11n is a proposed amendment to the IEEE 802.11-2007 standard [16] in order to significantly improve the network performance of the previous standards such as 802.11b and 802.11g. IEEE 802.11n is built based on previous standards of the 802.11 family, adding Multiple-Input Multiple-Output (MIMO) and binding of network interfaces (Channel Bonding). It also adds frames to the MAC layer.

It presents an increase of the theoretical maximum rate of 600 Mbps of data transfer. Currently it supports a PHY rate of 450 Mbps, using 3 spatial streams in a channel width of 40 MHz. Furthermore, IEEE 802.11n uses MIMO based on using multiple transmit and receive antennas to improve system performance. This technology requires a separated radio-frequency chain and an analog to digital converter for each MIMO antenna which increases the implementation costs compared to the systems without MIMO technology.

Table 1 summarizes the main characteristics of the four variants of the IEEE 802.11 standard.

The success of the standard has caused density problems related to crowding in urban areas. So, some issues must be studied such as interference, coverage and used bandwidth in each IEEE 802.11 variant. In our previous work, presented in a conference [1], we carried out coverage measurements of several devices that were capable to work in different IEEE 802.11 variants. In addition, the number of lost packets, bandwidth and throughput when there are interferences, for each IEEE 802.11 variant were measured. In this paper, we have added, on the one hand, some coverage measures that were not performed in the previous work and, on the other hand, we have taken more measurements of lost packets, bandwidth and throughput for each IEEE 802.11 variant. The results will be the mean values for each variant.

Table 1. Technology comparison.

	IEEE 802.11a	**IEEE 802.11b**	**IEEE 802.11g**	**IEEE 802.11n**
Frequency band	5.7 GHz	2.4 GHz	2.4 GHz	2.4 / 5 GHz
Average Theoretical speed	54 Mbps	11 Mbps	54 Mbps	600 Mbps
Modulation	OFDM	CCK modulated with QPSK	DSSS, CCK, OFDM	OFDM
Channel bandwidth	20 MHz	20 MHz	20 MHz	20 / 40 MHz
Coverage radius	35 m	38 m	38 m	75 m
Unlicensed spectrum	Yes (it depends on countries)	Yes	Yes	Yes (it depends on countries)
Radio Interference	Low	High	High	Low
Introduction cost	Medium-Low	Low	Low	High-medium
Device cost	Medium-Low	Low	Low	Medium
Mobility	Yes	Yes	Yes	Yes
Current use	Medium	High	High	High
Security	Medium	Medium	Medium	High

In this paper, we are going to show the empirical coverage area and the signal strength inside the coverage area. This let us know which is the technology that provides better coverage features, but, in this case (compared with reference [1], we will take more measures in order to extract more reliable conclusions. Moreover, we are going to compare the interferences between neighboring channels for each technology in order to know the number of available channels that can be used to plan the wireless network.

The remainder of this paper is organized as follows. Section II shows the related work on WLAN coverage designs. This section is separated in two parts. The first part shows indoor coverage studies and the second part shows papers related with performance measurements such as interferences, bandwidth and throughput. The test bench where our measurements have been performed is shown in Section III. Section IV presents the coverage measurements performed and the graphs obtained for each device, working in different IEEE 802.11 variant. Section V shows the measurements of the interferences between channels for each variant. It shows the graphs of lost packets, bandwidth and throughput when there are interferences. Finally, Section VI concludes the paper and gives our future work.

II. Related Work

In the literature, there are some works related with performance test and interference calculations. However, very few people have worked with physical devices to obtain real values.

A. Related works of indoor coverage.

There have been many studies of indoor coverage for single-transmitter and single-receiver protocols, like IEEE 802.11a, IEEE 802.11b and IEEE 802.11g [6]. Others use the measurements obtained from the signal level of a group of access points to perform the channel planning while avoiding interference [15]. There are some that locate clients by using the signal strength received by several access points [5]. IEEE 802.11 infrastructure has the advantage of being available in numerous indoor environments, and is deployed in densities that allow for the possibility of positioning with meter level accuracy.

As IEEE 802.11 networks are widely deployed, there has been a significant amount of work about planning IEEE 802.11n wireless networks. In such networks, the use of Multiple-Input Multiple-Output (MIMO) transmission scheme changes the expected behavior of signal level due to its multiple antenna use, exploiting physical phenomena such as multipath propagation to increase the transmission rate and reduce the error rate.

Foschini [19] derives theoretically that for the same SNR a 2x2 MIMO channel can hold twice the amount of bandwidth than using a single transmission and receiving antenna. As shown in [20] even further gain can be expected by the use of larger arrays of antennas in both reception and transmission.

Most of the recently published papers have modeled the MIMO channel matrix with independent and identically distributed Gaussian entries, which is an idealistic assumption, especially for indoor scenarios. More realistic MIMO channel models can be generally divided into three classes: ray-tracing, scattering and correlation models [21]. Anyway, for indoors need very large simulation time and complexity for trying to provide a good prediction of the channel behavior. On the other hand, correlation-based models don't provide detailed information about coverage, which could be needed for applications like indoor positioning [22].

There are still a few indoor IEEE 802.11n channel measurements reported in the literature. Simulation methods of these channels based on direct measurements are even fewer. At the 5 GHz band, for example, the publicly available IEEE TGn models [23] are the most convenient tools for MIMO channel simulations. However they have their own limitations; e.g., they are based on single-input single-output (SISO) channel models presented in [20] which do not reflect accurately the multipath propagation channel.

Another work that studies the coverage is presented by E. Amaldi et al. in [24]. This paper describes the optimization models with hyperbolic and quadratic objective functions. The authors propose heuristic methods that combine greedy and local search phases, and show the need of appropriate planning models and procedures that are specific to WLANs. The authors suppose that the system affects to the coverage planning process, and the incidence of overlapping regions should be taken into account in the planning procedure (beside of all the other optimization parameters). The computational results show that their heuristics provide near-optimal solutions within a reasonable amount of time.

Finally, in [25], J. N. Davies et al. measure the IEEE 802.11n signal level in a real building, but they don't make any comparison with IEEE 802.11a or IEEE 802.11g.

B. *Related works of interferences.*

From the IEEE 802.11 WLAN interference side, Nicolescu [26] proposed a model for interference in dense wireless networks that enables a low complexity procedure to collect the interference map and can be used to predict the damage from several simultaneous interferers. Unfortunately measurement of the interference map faces asymmetries in the card and channel behavior, which make the complexity still prohibitive for dense multiple card networks, requiring direct measurements in indoor deployments. Also, Fuxjäger et al. [27] show that the assumption of perfect independence between non-overlapping channels does not always hold in practice, by means of simple experiments with commercially available hardware, and found that the level of interference varies with physical distance, concurrent link-load, modulation rate, frame size, transmission power, receiver sensitivity and design, antenna patterns, etc., calling also for more direct measurements.

J. Padhye et al. present in [28] an interference measurement-based study between links in a static, IEEE 802.11, multi-hop wireless network. Then, the authors propose a simple empirical estimation methodology that can predict pair wise interference using only measurements. These tests are based on heuristics methods where the wireless links are defined by their packet loss rate. They state that this methodology could be applicable to any wireless network that uses omni-directional antennas.

Related to interference and throughput measures in WLAN, J. Jun et al. present in [29] an accurate formula to estimate the throughput in IEEE 802.11 networks, for several variants (802.11, 802.11b, 802.11a), in the absence of transmission errors and for various physical layers, data rates and packet sizes. The authors cite some applications where it is very important to know the maximum throughput in order to design them correctly. Theoretical Maximum Throughput can be used to facilitate optimal network provisioning, for example, in multimedia applications. It can influence the topological distribution of the nodes in the case of ad-hoc networks.

Although analytical studies and network simulations may provide valuable insights of the WLANs' operation, they cannot predict the actual performance of practical implementations with high accuracy. Moreover, measurements obtained from file transfer operations are also limited by the need to specify the processor type, processor speed and the network operating system. B. Bing presents in [30], an experimental study to characterize the behavior in terms of throughput and response time of two commercial AP under different degrees of network load. They are WavePOINT, from Lucent Technologies, and Spectrum24, from Symbol Technologies. The tests showed important characteristics such as throughput and response time under various network loads. The author also shows that the length of a data frame and the wireless bit rate also affects to the WLAN's transmission capabilities. But, the performance of an IEEE 802.11 WLAN is generally unaffected by the type of frame and the use of reservation frames such as RTS and CTS.

III. The Scenario Description

In this section, we describe the scenario where the measures have been taken and the hardware and software used to perform our research.

A. *Place of measurement*

In order to do the measures, we have sought a wide enclosure with an area of 91 m^2, with a length and a width of 12.5 m by 6.68 m. This building is made with walls of different thickness and materials. We have tried to find a scenario that was made from different materials, as can be found in common houses.

Fig. 1 shows the plane of garage. It has rectangular base, divided into two parts by a wall of 9 cm of thick: the garage (left) and the kitchen (right). The enclosure of the staircase is made of bricks with high consistency. All these walls have a layer of plaster and paint on both sides. The bathroom is made with hollow bricks of 9 cm. These walls are covered by ceramic tiles. All external walls are double with a thermic and acoustic insulation of polystyrene. Fig. 1 also shows the APs placement. In red we can see AP, which has been used for the coverage measurements test bench. In green we can see AP1 and AP2, which have been used for the interference test bench. Their placements have been decided randomly in order to avoid having equidistant placements.

B. *Hardware used in the test bench*

Four APs of different brands and models have been used. All of them are capable to working in different wireless technologies of IEEE 802.11 a/b/g/n (depending on the model). The models used are described below:

- Linksys WRT320N: It is a small device that is able to work in IEEE 802.11a/b/g/n standard variants. It works at frequencies of 2.4 GHz and 5GHz. It has three internal antennas, needed to work in IEEE 802.11n. Its RF power (EIRP) is 17 dBm.
- Dlink DWL-2000AP+: It works on IEEE 802.11g. It can be configured to work as a wireless AP, as a point to point bridge with another access point, as a wireless bridge point to multi-point or as a wireless client configured. Its output power is 16 dBm.
- Cisco Aironet 1130AG: It has been built to provide wireless coverage in offices and workplaces for their services. It is designed to be hung on the wall in a vertical position and it can works in IEEE 802.11a/g but it is also compatible with 802.11 b. It output power in IEEE 802.11a is 17 dBm, and its output power in IEEE 802.11g is 20 dBm.
- Linksys WRT54GL: This device is capable of working in the variants IEEE 802.11b/g, therefore, is only capable of emitting at a frequency of 2.4 GHz. It has 2 external antennas, which are used to correct the multipath effect. His RF power (EIRP) is approximately 18 dBm.
- Linksys WUSB600N: This is a wireless USB interface device that has been used as the capture device for all Laptops and PCs used in the test bench. This wireless card is able to capture IEEE 802.11a/b/g/n signals. It has a transmitting power of 16 dBm in all variants and its receiver sensitivity is approximately -91 dBm.

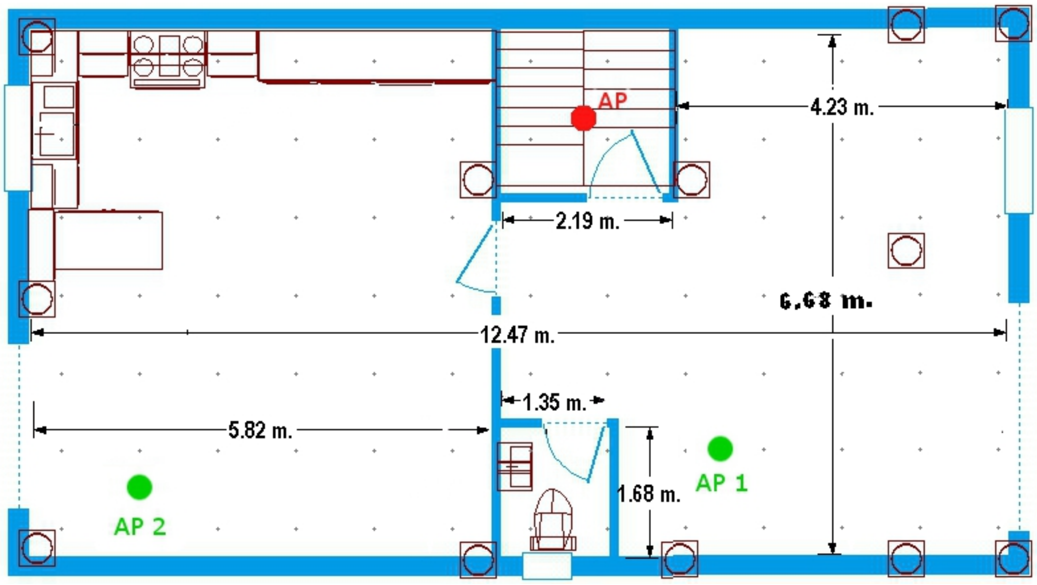


Figure 1. Scenario and APs situation.

In order to take the coverage measurements, we have used a laptop with dual core processor at 1.67 GHz per core and 1 GByte of RAM. In addition, two desktop PCs with an AMD 1700MHz CPU and 1 GByte of RAM memory have been used to take the interference measurements.

C. *Software used*

This subsection describes the software used to perform our test bench.

- InSSIDer is a freeware that can detect wireless networks and manage, in a graphics mode, the intensity of these signals. This program let us detect all wireless networks in the test area on the computer screen and lists all of their details: SSID, MAC address, channel, Radio Signal Strength Indicator (RSSI), type of network, security, speed and signal intensity and allows monitoring the signal quality via a chart using the received RSSI [31].
- MS-DOS commands. The MS-DOS shell presents some utilities and commands, which allow checking the status of the network connection.
- Net Meter monitors the network traffic used by all network interfaces [32]. It displays in real-time graphical and numerical downloading and uploading bandwidth rates.

IV. COVERAGE MEASUREMENTS

This section describes the strategies carried out to do the coverage measurements and the measures obtained.

A. *Process to gather the coverage measurements*

First we measured the wireless coverage offered by each device, working on various wireless technologies. These signal values depend mainly, on the losses suffered due to the walls traversed and the multipath effect.

In order to perform this work, we draw a grid in the garage floor. It allowed us to take measurements of all devices in the same place. The position of the measure points is seen in Fig. 1. The equidistant points shown in the figure are separated 1m from each other.

Each access point has been located in the stairwell (marked in red on Fig. 1), at a height of 50 cm of the floor. The signal power levels received at each measure point is collected by a laptop running the application software InSSIDer. The used capture device was a WUSB600N wireless card for all computers in order to avoid adding some sort of error taking the measurements. The laptop was located at a height of 50 cm above the ground.

B. *Results of coverage measures*

In Fig. 2, we can see the legend used for all coverage graphics. All values shown are measured in dBm, with an absolute error of 1dBm.

Fig. 3 shows the level of coverage obtained with Linksys WRT320N when it is configured to work only in 802.11a. As the figure shows, the best coverage is located in the stairwell. The signal is propagated out of the walls of the stairwell to the outer walls. Then, there signal strength is quickly decreased with some low peaks in the coverage area.

Fig. 4 shows the coverage obtained with the Cisco Aironet 1130AG when it is configured to work only in 802.11 a. This device presents the lowest signal level. This may be due to the antenna radiation direction (we place all the devices in the same position, independently of the placement of the antenna inside of them).

Fig. 5 shows the level of coverage obtained with Linksys WRT320N configured to work only in 802.11b. In this case, the best coverage is located in the staircase, but the signal is decreased quickly as it is propagated to the garage. The kitchen area has a lower signal level than the garage.

Fig. 6 shows the level of coverage obtained with the Linksys WRT54GL configured to work only in 802.11b. It has been the device that provides higher signal strength in the coverage area.

Fig. 7 shows the level of coverage obtained with the Linksys WRT320N configured to work only in 802.11n. Although there is high signal strength close to the access point, there are suddenly low values in the coverage area.

Fig. 8 shows the level of coverage obtained with the Dlink DWL-2000AP configured to work only in 802.11 g. This device presents the highest signal levels in almost all the garage surface, and the kitchen's area.

Fig. 9 shows the level of coverage obtained with the Cisco Aironet 1130AG configured to work only in 802.11 g. This device is the one that presents the lowest signal level. This may be due to the antenna radiation direction (we place all the devices in the same position, independently of the placement of the antenna inside of them).

Fig. 10 shows the level of coverage obtained for the Linksys WRT320N configured to work only in 802.11g. As we can see, when it is working in IEEE 802.11b, its coverage is better than in IEEE 802.11g . It is even more significant in closest distances.

Fig. 11 shows the coverage obtained with the Linksys WRT54GL when it is configured to work only in 802.11g. It has been one of devices that provides the lowest higher signal strength in the coverage area.

🟦 -75--70 🟪 -70--65 🟦 -65--60 🟦 -60--55
🟩 -55--50 🟩 -50--45 🟨 -45--40 🟨 -40--35
🟧 -35--30 🟥 -30--25 🟫 -25--20

Figure 2. Colour legend.

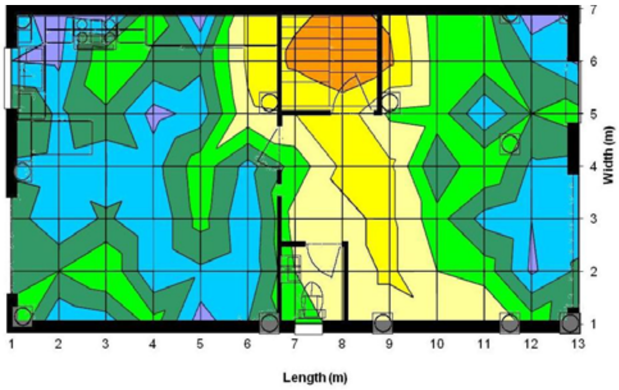


Figure 3. Coverage to Linksys WRT320N in 802.11a.

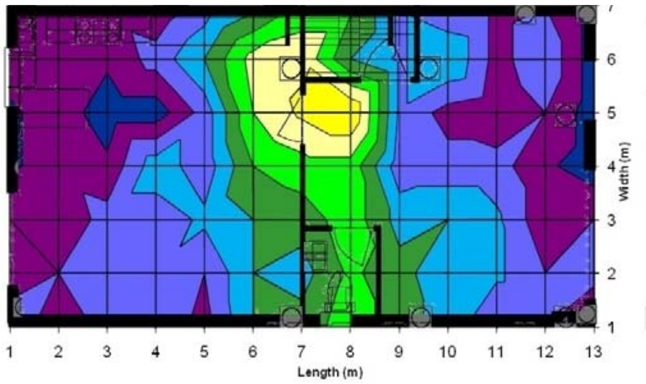


Figure 4. Coverage to Cisco aironet 1130AG in 802.11a

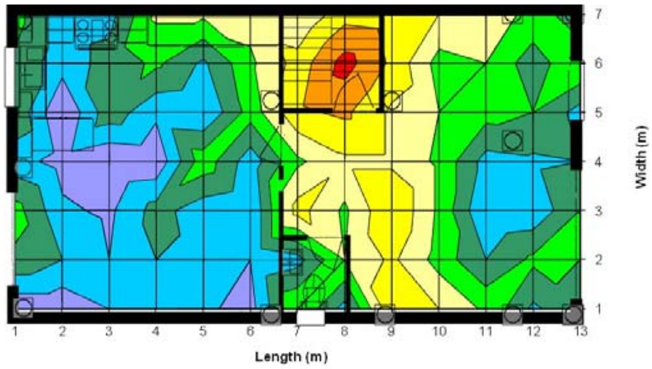


Figure 5. Coverage to Linksys WRT320N in 802.11b.

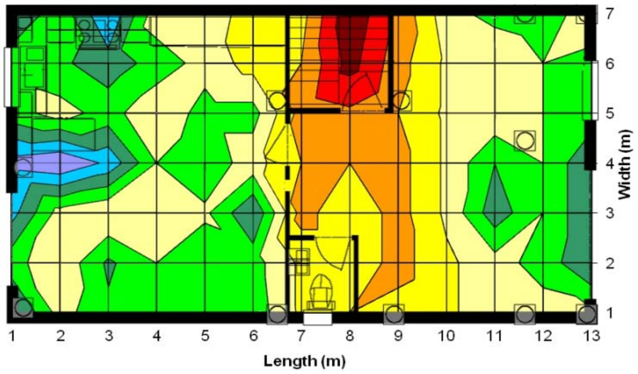


Figure 6. Coverage to Linksys WRT54GL in 802.11b.

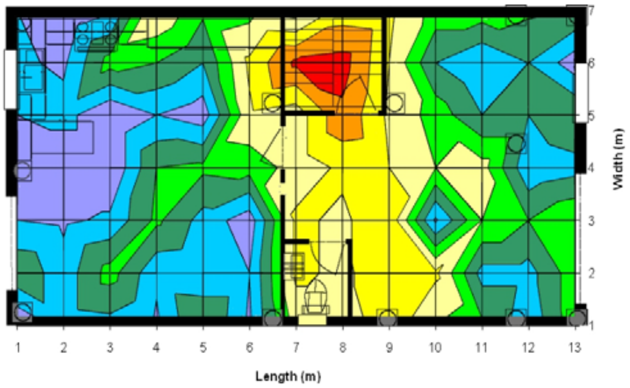


Figure 7. Coverage to Linksys WRT320N in 802.11n.

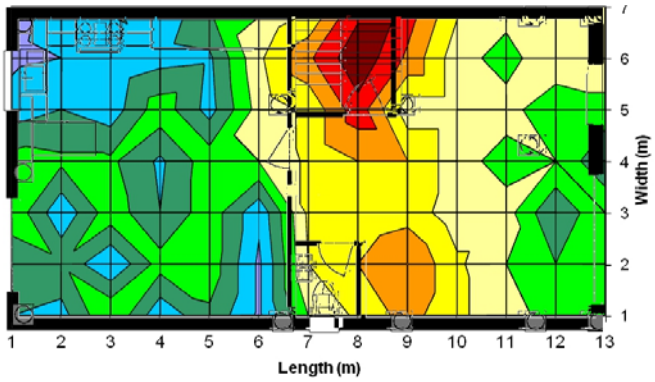


Figure 8. Coverage to Dlink DWL-2000AP in 802.11g.

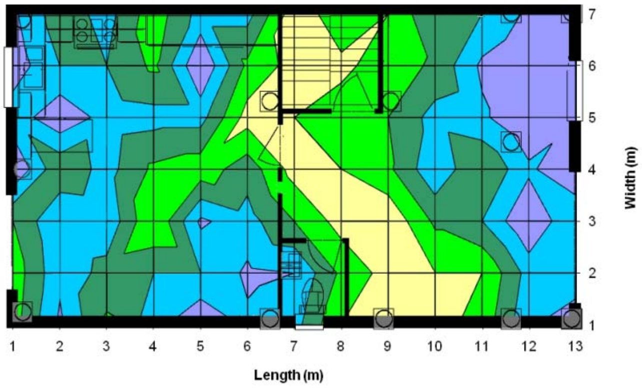


Figure 9. Coverage to Cisco Aironet 1130AG in 802.11g.

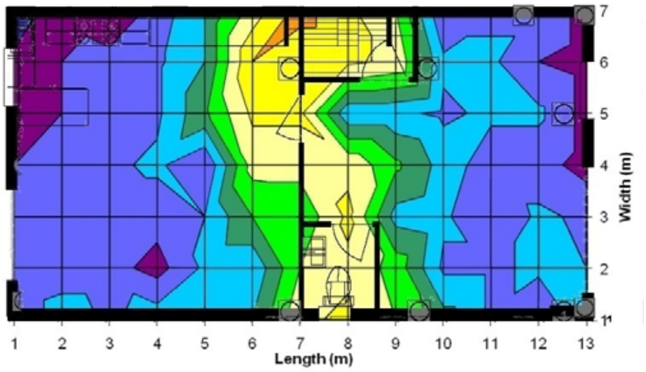


Figure 11. Coverage to Linksys WRT54GL in 802.11g

V. INTERFERENCE MEASUREMENTS

This section describes the process used to make the interference measurements. It also shows the topology of PCs and APs.

A. Scenario

In this test, we have used four PCs and two APs of the same brand and model. Fig.1 shows the location of AP1 and AP2 (marked in green). These sites are chosen to ensure that there are walls between the two small wireless networks.

First, we used channel 6 for both wireless devices. Then, we configured different IP networks in order to perform our test. Now we are able to measure the effects of the interference generated by another network working in the same channel. In order to take the measurements, we changed the working channel in one device while the other remained fixed. The measurements were taken for each channel until there was a difference of 5 channels. With the collected data, the average value of lost packets, throughput and bandwidth was estimated. This let us know the behavior of each variant based on the number of overlapping channels.

Fig. 12 shows the topologies. The PCs are situated at a distance of approximately 1 m from the AP which is associated to. A large file is transmitted between the computers associated to the AP2. Meanwhile, measurements of the packet loss, throughput and bandwidth consumed are carried out in the Wireless Network of the AP1.

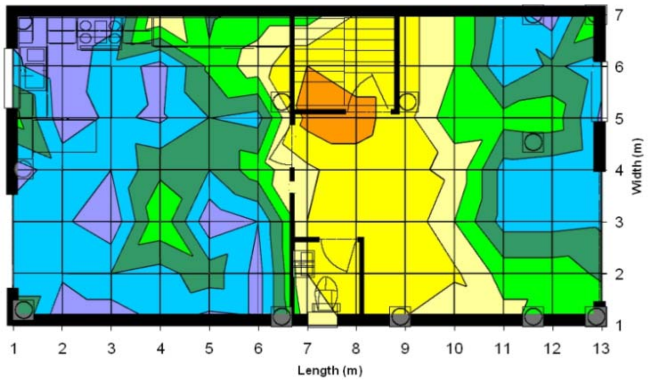


Figure 10. Coverage to Linksys WRT320N in 802.11g.

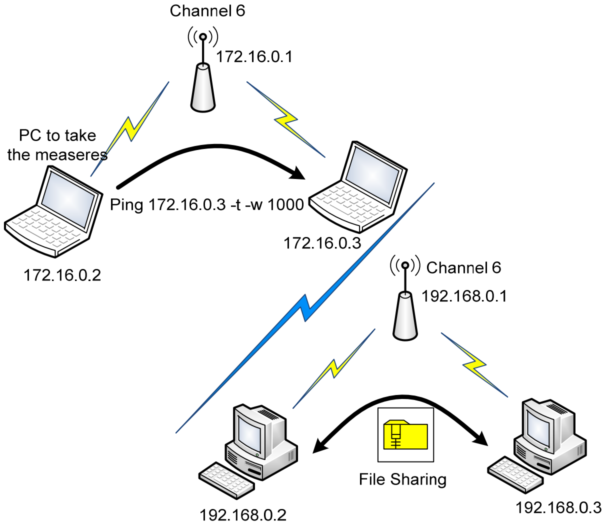


Figure 12. Network topology.

In order to present our results, we have grouped the collected data of the variants IEEE 802.11 b/g/n (all of them work at 2.4 GHz) in the same figures, because the distribution channels were the same, while IEEE 802.11a, which works at 5GHz, was displayed in separate figures.

The measures were made with all devices under test, working on different variants. Later, we computed the average value of lost packets, throughput and bandwidth.

B. Lost Packets

Based on the physical properties of radio-frequency signals, when we have two frequency components, represented in the same spectral domain, the interference should be low (which results in fewer lost packets), because overlapping spectra is lower. But, in the following simulations we have found that the measurements do not always follow this behavior.

In order to know the lost packets, a ping is transmitted between the PCs associated to AP1. We fixed the maximum time to 1000 milliseconds. After this time, the packet will be considered lost. We choose a small time, because it is a small network, without a large number of intermediate devices that may introduce delays. Measurements were taken during 3 minutes in each the devices.

The obtained results as a function of the amount of channel separation are shown in the following figures.

Figure 13 shows the number of lost packets for IEEE 802.11 b/g/n, when both devices are working on the same channel. As we can see b variant has higher number of lost packets (around 44%) than g variant, while devices working in IEEE 802.11n do not have lost packets.

Figure 14 shows the number of lost packets for IEEE 802.11 b/g/n, where there is one channel between them. In this case, b variant records around 55% of lost packets and g variant has lost 42%. IEEE 802.11n does not have lost packets.

Fig. 15 shows the number of lost packets for IEEE 802.11 b/g/n, where there are two channels of separation. The number of lost packets, when the devices operate in IEEE 802.11b/g variants, was between 42 and 46%. In this case, IEEE 802.11n variant has 1% of lost packets.

Fig. 16 shows the number of lost packets for IEEE 802.11 b/g/n, when there are three channels of separation. The number of lost packets for IEEE 802.11b/g was between 40 and 44% and the IEEE 802.11n variant does not have lost packets.

Figure 17 shows the number of lost packets for IEEE 802.11 b/g/n, when there are four separation channels. As we can see the IEEE 802.11b variant has a higher number of lost packets (around 42%) than the IEEE 802.11g variant (around 37%). While IEEE 802.11n, does not have lost packets.

Figure 18 shows the number of lost packet for IEEE 802.11 b/g/n, when there are five separation channels. The number of lost packets for IEEE 802.11b/g was between 38 and 42%, and IEEE 802.11n didn't report lost packets.

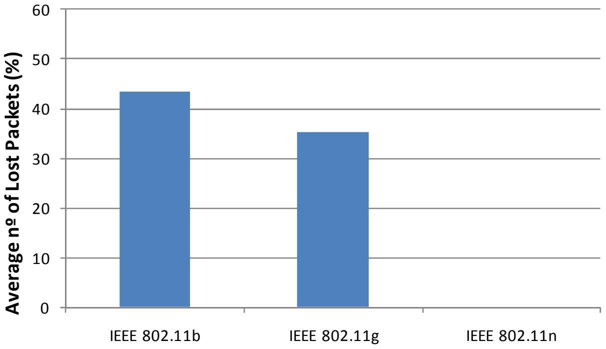


Figure 13. Lost packets with overloaping channels.

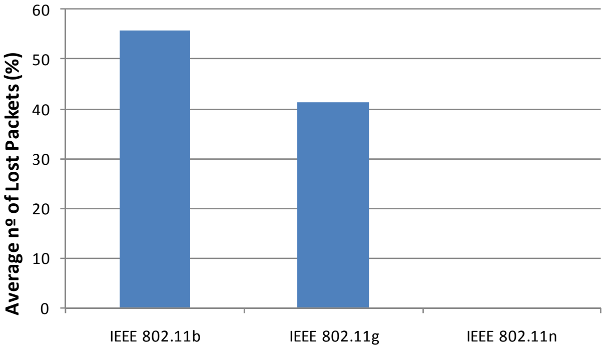


Figure 14. Lost packets with one channel of difference.

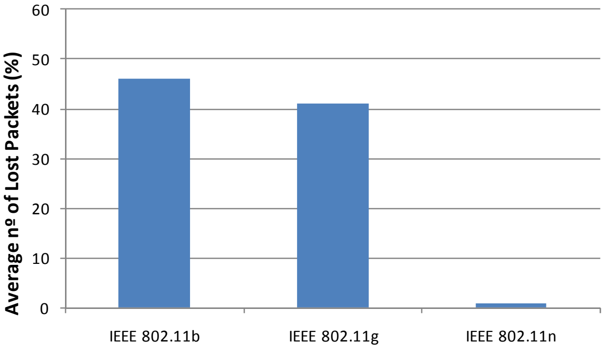


Figure 15. Lost packets with two channels of difference.

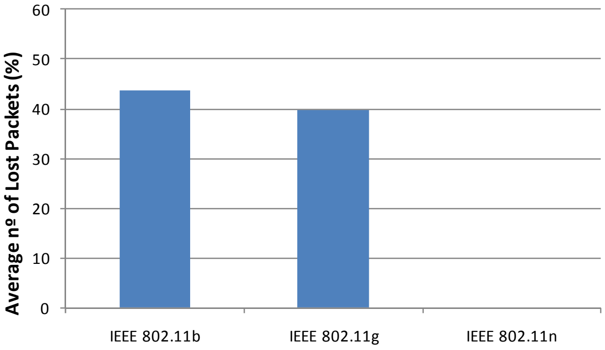


Figure 16. Lost packets with three channels of difference.

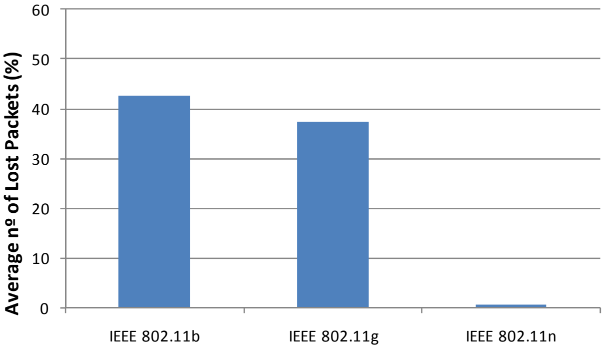


Figure 17. Lost packets with four channels of difference

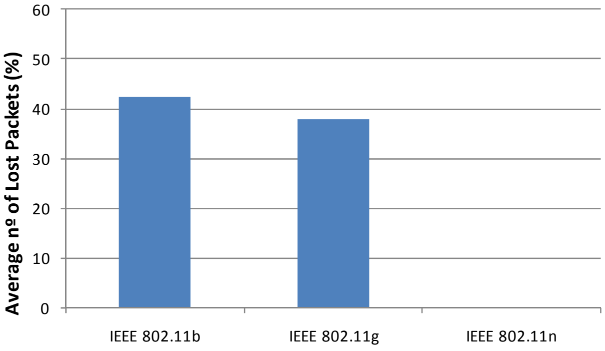


Figure 18. Lost packets with five channels of difference.

The measurements show that devices operating under the IEEE 802.11n variant does not have lost packets, while the IEEE 802.11b and IEEE 802.11g variants could have 40% of lost packets (depending on the amount of separation channels). This is mainly because IEEE 802.11n uses MIMO technology, where both transmitter and receiver have multiple antennas reducing the interferences. Generally, in traditional wireless transmission the signal is affected by reflections, causing self-degradation and therefore data loss. MIMO takes advantage of physical phenomena such as multipath propagation to increase the transmission rate and reduce the error rate. That is, MIMO increases the spectral efficiency of wireless communication system through the use of the space domain.

In IEEE 802.11b and IEEE 802.11g variants there is a slight tendency to record fewer lost packets when the channel separation is an even number than when there is an odd channel of separation. These losses can be approximated in both cases by a fifth degree polynomial with high accuracy. In particular, taking the measurements gathered, IEEE 802.11b follows expression 1:

$$y = 0.09x^5 - 1.14x^4 + 5.5x^3 - 12.98x^2 + 14.54 + 35.25 \quad (1)$$

And IEEE 802.11 g follows expression 2:

$$y = 0.35x^5 - 4.95x^4 + 25.96x^3 - 59.43x^2 + 0.31x + 43.5 \quad (2)$$

Where x is the separation between working channels and y represents the value in % of lost packets.

In order to test the performance of IEEE 802.11a variant we used two different devices. The first one was Cisco Aironet 1130AG working in IEEE 802.11a, which uses dynamic frequency selection (DFS). This system does not allow us to select different channels (such as we did in IEEE 802.11 b/g/n). The other device was the WRT320N working in IEEE 802.11a, which only works in the channels 36, 40, 44, 48. Although in the 5GHz frequency band, devices could work theoretically with 8 non-overlapping channels simultaneously, this device only allows us to work with 4 non-overlapping channels. Then, we estimated the average bandwidth for each device when there are no overlapping channels. The results are represented in fig. 19.

As fig19 shows, WRT320N working in IEEE 802.11a presents five times less packet loss than the Cisco Aironet 1130AG device (which has around 40%).

C. Throughput and Bandwidth consumption measurements

In order to measure the bandwidth offered by each technology, we performed the following test. First, 2 PCs were associated to the AP2 and were transmitting large files consuming all the bandwidth available in this network. Then, there were 2 PCs associated to the AP1, which were transmitting a large file too. The Net Meter captured the consumed bandwidth in one of these PCs. The measures are carried out during 3 minutes.

The result of the average bandwidth consumed by each IEEE 802.11 variant for different the number of separation channels is shown in the following figures.

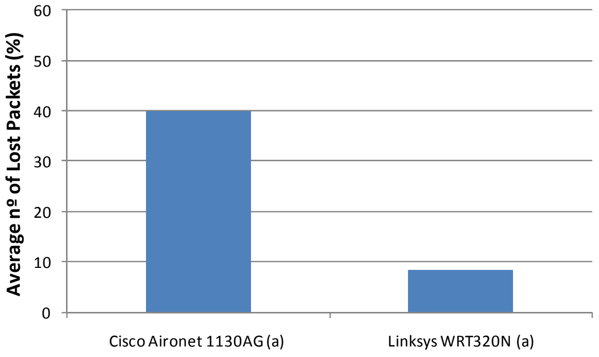


Figure 19. Lost packets with non overlapping channel

This average has been estimated taking into account the measurements taken from all devices for each variant.

Fig. 20 shows the bandwidth consumed for IEEE 802.11b/g/n, when both devices are working in the same channel. As we can see, IEEE 802.11b variant has approximately an average of 3Mbps, while IEEE 802.11g and n variants, provide around 10-12 Mbps.

Fig. 21 shows the bandwidth for IEEE 802.11b/g/n, when there is one separation channel. In this case, the IEEE 802.11b variant shows a mean bandwidth of 2Mbps and IEEE 802.11g and n variants provide approximately 11Mbps.

Fig. 22 shows the bandwidth for IEEE 802.11b/g/n, when there are two separation channels. The average bandwidth values for IEEE 802.11g and n are between 10-11Mbps. Furthermore, IEEE 802.11b is using an average bandwidth of 2.5Mbps.

Fig. 23 shows the bandwidth for IEEE 802.11b/g/n, when there are three separation channels. In this case, IEEE 802.11g variant has higher bandwidth (11Mbps) than the IEEE 802.11n variant (10Mbps). IEEE 802.11b variant has an average value of 3Mbps.

Fig. 24 shows the bandwidth for IEEE 802.11b/g/n, when there are four separation channels. We can see that IEEE 802.11b variant has an average value of 3 Mbps, while IEEE 802.11g variant has 10.5 Mbps and IEEE 802.11n variant has 12 Mbps.

Fig. 25 shows the bandwidth for IEEE 802.11b/g/n, when there are five separation channels. The IEEE 802.11b variant maintains its average value around 3 Mbps, while IEEE 802.11g/n variants have their average values very close to 10Mbps.

As it happens in lost packets measurements, there is a trend in the bandwidth consumption that is related to the separation of the working channels. On the one hand, IEEE802.11n and b variants, present higher mean values when the separation between the working channels is even, while IEEE802.11g variant, has its maximum values when the number of separation channels is odd.

Fig. 26 shows the bandwidth for IEEE 802.11a variant when channels are not overlapped. As we can see, both devices show a similar average bandwidth, with an average value of 11.4 Mbps.

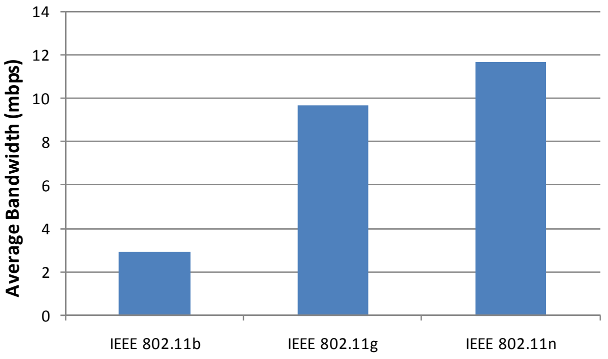


Figure 20. Average bandwidth with overloaping channels.

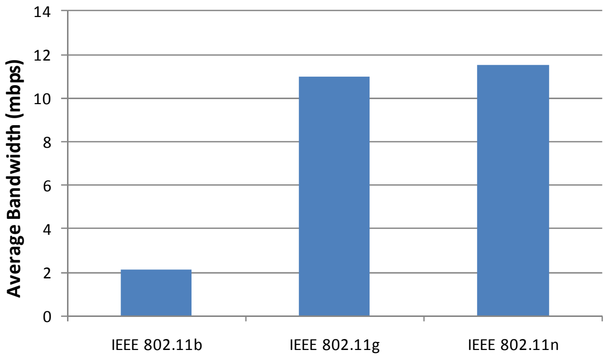


Figure 21. Average bandwidth with one channel of difference.

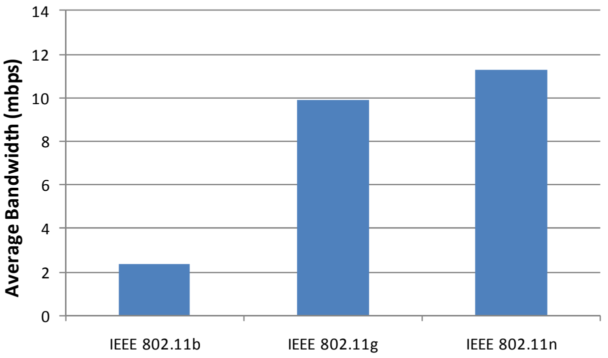


Figure 22. Average bandwidth for two channel of difference.

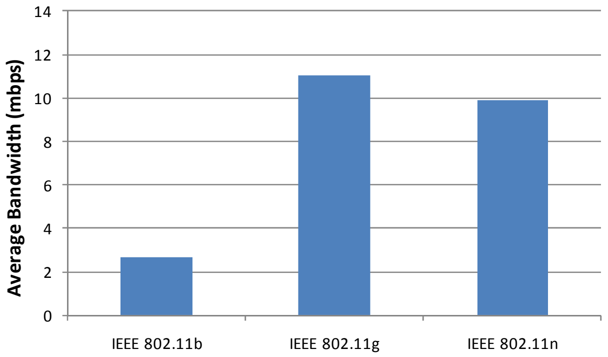


Figure 23. Average bandwidth for three channel of difference.

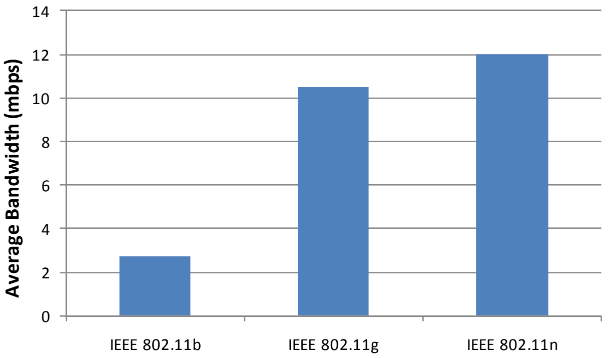


Figure 24. Average bandwidth for four channel of difference.

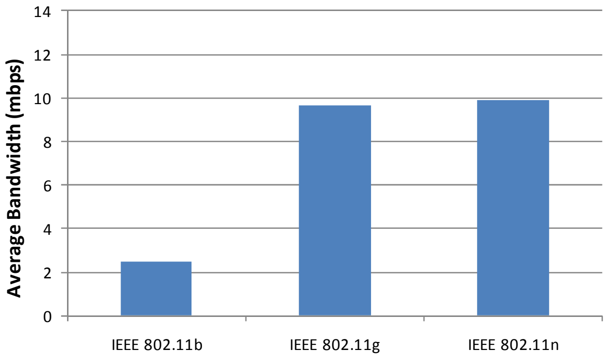


Figure 25. Average bandwidth for five channel of difference.

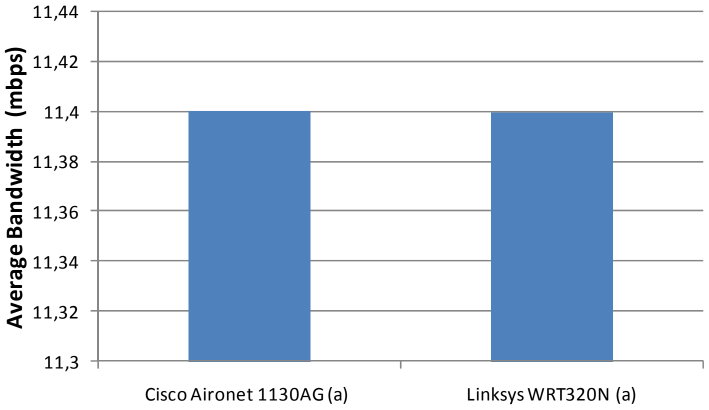


Figure 26. Average bandwidth when there are no overlapping channels in IEEE 802.11 a variant.

Finally, in order to see the use of the link capabilities for each IEEE 802.11 variant, we have measured the average throughput. These values have been obtained by dividing the average bandwidth consumption by the theoretical bandwidth of the IEEE 802.11 variant. It provides us the percentage of throughput consumption for each variant. For the IEEE 802.11n variant, we have used 320 Mbps as a reference to compute the percentage, because the used devices were limited to this speed. The results of the throughput average, as a function of the distance between channels, are shown in the following figures:

Fig. 27 shows the throughput average of IEEE 802.11b/g/n, when both devices are working in the same channel. IEEE 802.11b variant has an average throughput of 27%, meanwhile IEEE 802.11g variant maintains its average value around 18% and IEEE 802.11n variant has an average value of 4%.

Fig. 28 shows the average throughput of IEEE 802.11b/g/n variants, when there is a separation of one channel. In this case, IEEE 802.11b/g variants show values around 20% and IEEE 802.11n maintains its average value in 4%.

Fig. 29 shows the average throughput of IEEE 802.11b/g/n variants, when there are two separation channels. Again, IEEE 802.11n variant shows the lowest value, while IEEE 802.11g has its average value around 18% and the IEEE 802.11b version presents an average throughput of 22%.

Fig. 30 shows the average throughput of IEEE 802.11b/g/n, when there are three separation channels. In this case, IEEE 802.11b and g have increased their average values, locating them between 21% and 24%, but IEEE 802.11n has an average throughput of 4%.

Fig. 31 shows the average throughput of IEEE 802.11b/g/n, when there are four separation channels. The IEEE 802.11b variant has a throughput around 24.5% and IEEE 802.11g variant presents an average value of 19%. IEEE 802.11n variant maintains its value.

Fig. 32 shows the average throughput of IEEE 802.11b/g/n, when there are five separation channels. In this case, all values have decreased slightly. They have 22.5% for IEEE 802.11b, 17% for IEEE 802.11g and 3% for EEE 802.11n.

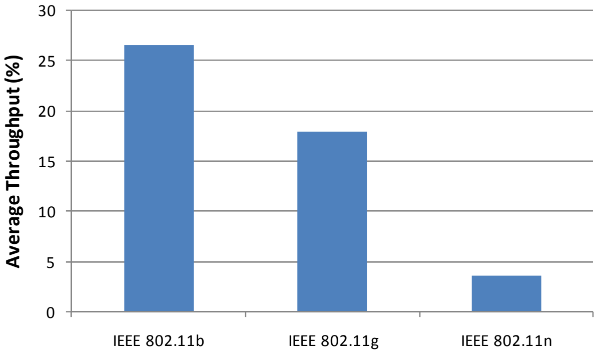


Figure 27. Average throughput with overloaping channels.

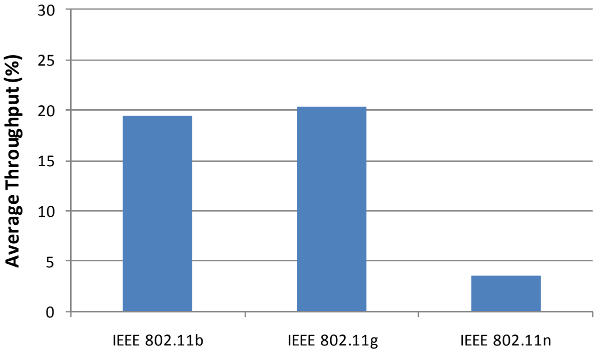


Figure 28. Average throughput with one channel of difference.

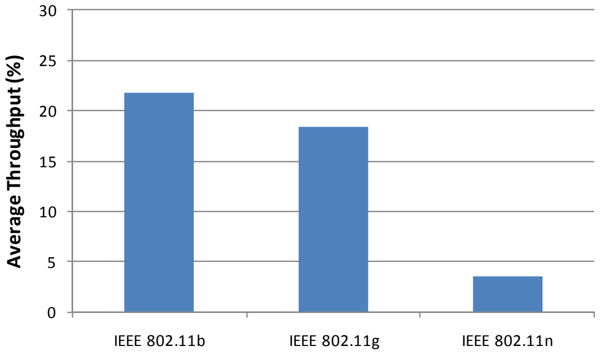


Figure 29. Average throughput with two channels of difference.

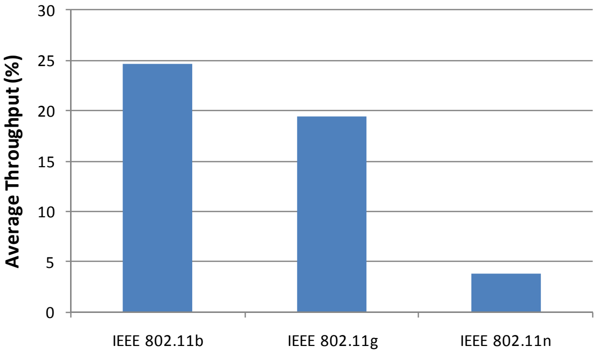


Figure 30. Average throughput with three channels of difference.

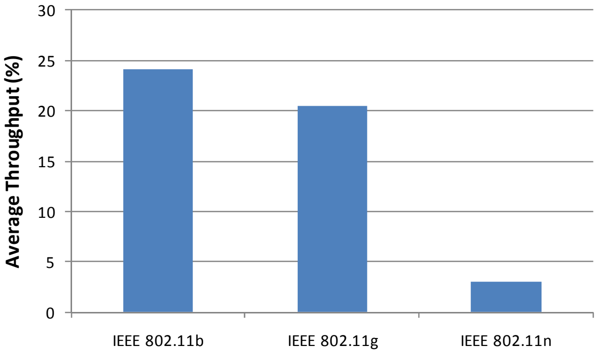


Figure 31. Average throughput with four channels of difference.

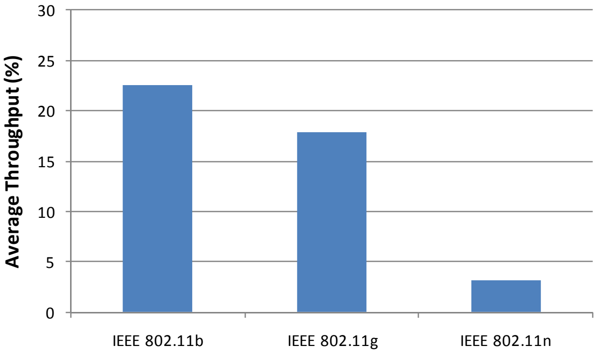


Figure 32. Average throughput with five channels of difference

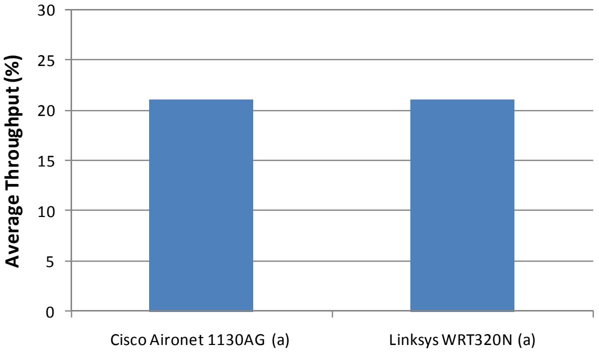


Figure 33. Average throughput with non overlapping channels

In general, we can see similar effects in all figures. The IEEE 802.11 variant that has lower average throughput is IEEE 802.11n, which in no case exceeds 5%. In IEEE 802.11g, the average throughput values is around 20%, while the variant that best uses its available bandwidth is IEEE 802.11b, which has average values very close to 25% of its total capacity.

For IEEE 802.11a variant, the average throughput measures are shown in fig. 33. Both devices have a similar average throughput (approximately 21%).

In this case it is clear that the interference highly affects the performance of the wireless network variant. IEEE 802.11a, IEEE 802.11b and IEEE 802.11g have higher throughput than IEEE 802.11n, (which average values are between 21% and 25%).

VI. Measurement discussion

Throughout this work we have performed different test benches in order to characterize the behavior of wireless signals in indoor environments. These tests have allowed us to check some statements realized by some papers related to this issue. We have also seen some other issues.

Moreover, we analyzed the frequency spectrum between 2.4 and 2.5 GHz, which includes all channels used in IEEE 802.11 b/g/n. Because of their physical properties, we could think that if there is no overlap between channels, there should not be any interference between them. But our results did not reflect this fact. In [27], authors show in their simulation that non-overlapping channels do not have interference. In fact, their tests were different from ours. They analyzed these losses depending on the distance between nodes. Despite of this fact, the conclusions are similar. As we have seen, the devices working at 2.4 GHz register fewer lost packets due to interference, when the channels are fully overlapping (Fig. 27), than when there is one channel of separation between the devices (Fig. 28). We also see that the number of lost packets, maintaining approximately the same value when there are 3 channels of separation (Fig. 30), when there are 4 (Fig. 31) or 5 channels (fig.35). In the last two cases, the number of lost packets should be very low or zero, since the overlap between the spectral is virtually nonexistent. In this case, after having performed different tests, we state that there is a slight tendency to register fewer lost packets when the channel separation is an even number than when the channel separation is an odd number. These measurements enabled us to characterize this behavior to a fifth degree polynomial with a correlation value close to the unity.

Moreover, we can extrapolate the analysis about the number of lost packets to the bandwidth measurements, where the behavior is identical. That is, when the channel separation is an even number, there is a greater bandwidth than when the channel separation is an odd number. This fact corresponds to the values of lowest packet losses. Therefore, although we have shown that non-overlapping channels does not mean less interference level, the analysis shows that a greater number of lost packets corresponds to a lower useful bandwidth in the network.

Some published papers define the throughput, as the volume of information that traverses the network over time. And others define the throughput as the channel performance. We have taken the second meaning of this concept. It relates the amount of information flowing through the channel and the theoretical maximum capacity offered by technology. Another factor that draws the attention of this analysis is that despite of the packet losses registered in IEEE 802.11n is low; the value of throughput and channel performance is quite low. As we can see in [33] the theoretical maximum throughput and data rates for IEEE 802.11 networks in a and b variants are different compared with real throughput and data rates.

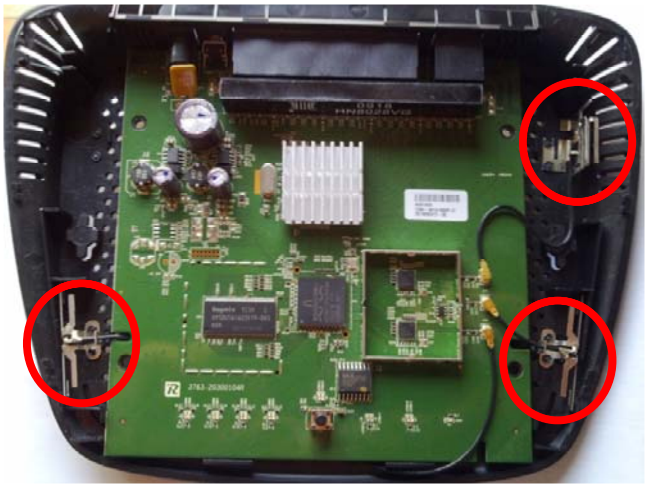


Figure 34. Internal view of WRT320N

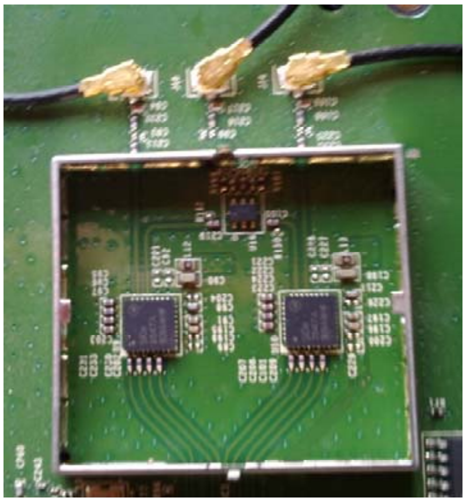


Figure 35. Integrated circuits for WRT320N

Although this phenomenon has been seen in the 4 variants measured, the n variant is the one that has the biggest difference among its theoretical value and real value. In order to analyze this, first we have analyzed the hardware characteristics of wireless device. In this case, WRT320N has 3 antennas, as shown in Fig. 34, and it should use the 3 antennas in order to work in IEEE 802.11n with MIMO.

However, as we can see in Fig. 35, only two out of three antennas are controlled by the SE2547A circuit. It allows a dual stream for both antennas. A review of this device is shown in a specific forum of wireless technology [34]. There are also other cases where the devices only work with 2 antennas, enabling a maximum effective flow rate of 315 Mbps compared to the 600 Mbps specified in the standard. In addition, the theoretical maximum data rate specified in the WRT320N datasheet is 320Mbps.

Moreover, the USB device used as a receiving interface is WUSB600N. It has 2 internal antennas and when it works in IEEE 802.11n, MIMO is also used. The network performance operating under IEEE 802.11n standard should be better than the other analyzed standards because the technology used (that is MIMO) allows it. Therefore, the most probable cause of the discrepancy between the number of lost packets and the information flowing through the channels is due to the hardware characteristics and the low performance of one of the two devices.

We demonstrated that this behavior is also observed in other variants. Other authors also analyzed IEEE 802.11 a and b theoretically [33] and observed this behavior.

VII. CONCLUSION

In this paper, we have measured the signal strength inside the coverage area of several WLAN variants (concretely in the IEEE 802.11a/b/g/n).

On the one hand, the measurements have been carried out under specific conditions, inside a house with a particular form and size. We chose an isolated place, free of wireless signals, in order to not having distorted results. We were pursuing accurate measurements. On the other hand, a specific antenna has been used, with a particular sensitivity. May be the same experiment performed in other conditions, may vary the results slightly. However, due to the results and other previous tests we had made, we believe that the results obtained are a good sign of the technology behavior. Similar results would be obtained under the same conditions

We can see that in the closest zones, the best technologies have been IEEE 802.11b and IEEE 802.11n, while the worst ones have been IEEE 802.11g and IEEE 802.11a. The one with highest signal strength in larger distances has been IEEE 802.11b and the worst ones have been IEEE 802.11g and IEEE 802.11n.

We have also measured the interferences between neighboring channels for each variant. We have observed different effects. On the one hand, we observed that the hypothesis, which told us that if we increase the separation of working channels, we should record lower losses, so it would not be always true. May be this effect happens because the measurements have been taken in closed zones, and the signal reflections may affect to the received signal strength. We think that this is a key factor when we are going to set up an IEEE 802.11 WLAN. Moreover, we have proved that packet looses have a fifth degree polynomial function of the channel separation (it matches this function almost exactly).

In general, although we have seen that the hardware used is more significant in the packet loss than the chosen IEEE 802.11 variant, we think that the variants IEEE 802.11b and IEEE 802.11g seem to be better for installations in closest zones.

Because there is an increasing number of wireless devices, and the presence of wireless networks working under the IEEE 802.11 technology is increasing, the likelihood to create interference is greater.

In a future work we will use the studies we have done in order to estimate the best position for a wireless sensor device inside a network, based on the received signal strength and the frequency interferences, in order to avoid having random sensor placements.

REFERENCES

[1] S. Sendra, P. Fernandez, C. Turró, J. Lloret," IEEE 802.11a/b/g/n Indoor Coverage and Performance Comparison" in procedings of The Sixth International Conference on Wireless and Mobile Communications, (ICWMC'10). Valencia, (Spain), September 20-25, 2010.

[2] D. Molkdar, "Review on radio propagation into and within buildings", IEE proceedings-H, 138(1):61-73, February 1991.

[3] H. Hashemi, "The indoor radio propagation channel", Proceedings of the IEEE, 81(7):943-967, July 1993.

[4] J. Lloret, J. J. López, C. Turró and S. Flores, "A Fast Design Model for Indoor Radio Coverage in the 2.4 GHz Wireless LAN", 1st International Symposium on Wireless Communication Systems 2004 (ISWCS'04), Port Louis (Mauricio Island), September 20-22, 2004.

[5] J. Lloret, M. Garcia, F. Boronat and J. Tomás, "The Development of Two Systems for Indoor Wireless Sensors Self-location", Ad Hoc & Sensor Wireless Networks: An International Journal, VOL: 8, Issue: 3-4, Pp. 235-258, June 2009.

[6] N. García, "Modelo de cobertura en redes inalámbricas basado en radiosidad por refinamiento progresivo", PhD. Thesis. Computer Science Department, University of Oviedo, Spain, March 2006.

[7] S. J. Flores, "Caracterización del canal radio móvil en el interior de edificios con múltiples plantas mediante técnicas de lanzado de rayos", PhD. Thesis. Department of Communications, Polytechnic University of Valencia, Spain, May 1998.

[8] T. S. Rappapport. "Wireless Communications: Principles and Practice". Prentice Hall. 2002

[9] D. J. Cichon and T. Kurner, "Propagation Prediction Models," COST 231 Final Rep. [Online], ch.4, pp. 17–21. 1996. Available: http://www.lx.it.pt/cost231/ [Last access: 6th June 2010]

[10] D. Andelman, "5 GHz WLAN Indoor Coverage Range: Truths and Misconceptions", White Paper 2005, Envara Inc, Israel. http://www.freeweb.hu/tordaif/HIF/RLAN/WSD.pdf [Last access: 6th June 2010]

[11] J.M. Keenan and A.J. Motley, "Radio coverage in buildings", British telecom technology Journal, 8(1):19-24, January 1990.

[12] R. S. Saunders, "Antennas and Propagation for Wireless Communication System", Ed.Wiley, 1999.

[13] J. Beyer and R. Jakoby, "Two Semi-Empirical and Fast Prediction Models for Urban Microcells Compared with Measurements at 919 and 1873 MHz", European Personal Mobile Communication Conference (EPMCC'97), Bonn, Germany, September 1997.

[14] P. Pechač and M. Klepal, "Empirical Models for Indoor Propagation in CTU Prague Buildings", Radioengineering, vol. 9, pp. 31-36. 2000

[15] J. Lloret, J. J. Lópezt and G. Ramos, "Wireless LAN Deployment in Large Extension Areas: The Case of a University Campus", Communication Systems and Networks 2003. Benalmádena, Málaga (España). September 8-10, 2003.

[16] IEEE Std 802.11 (2007) IEEE Standard for Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. Pp.1-1184. Institute of Electrical and Electronics Engineers, Inc. New York, USA.

[17] IEEE 802.11 Working Group, At http://www.ieee802.org/11/index.shtml [Last access: 3rd july 2011]

[18] IEEE 802.11 Working Group, At http://standards.ieee.org/about/get/802/802.11.html [Last access: 3rd july 2011]

[19] G. Foschini and M. Gans, "On Limits of Wireless Communications in Fading Environments when Using Multiple Antennas", Wireless Personal Communications, vol. 6, no. 3, pp. 311–335, Mar. 1998.

[20] M. A. Maddah-ali and A. S. Motahari, "Communication over mimo x channels: Interference alignment, decomposition, and performance analysis", IEEE Transactions on Information Theory. , Vol. 54, p.p. 3457 - 3470, August 2008.

[21] H. Xu, D. Chizhik, H. Huang, and R. Valenzuela, "A generalized space-time multiple-input multiple-output (MIMO) channel model," IEEE Trans. Wireless Commun., Vol. 3, 966–975, May 2004.

[22] 18 W. Weichselberger, M. Herdin, H. Ozcelik and E. Bonek, "A stochastic MIMO channel model with joint correlation of both link ends", IEEE Trans. Wireless Commun., Vol. 5, 90–100, Jan. 2006.

[23] IEEE 802.11-03/940r4: TGn Channel Models. IEEE. [Online]. Available: IEEE ftp://ieee:wireless@ftp.802wirelessworld.com/11/03/11-03-0940-02-000n-tgn-channel-models.doc [Last access: 6th June 2010]

20 J. Medbo and P. Schramm, "Channel models for hiperlan/2 in different indoor scenarios", ETSI BRAN 3ERI085B. ETSI UMTS 30.03 V3.2.0, March 1998.

[24] E. Amaldi, A. Capone, M. Cesana, F. Malucelli, F. Palazzo, WLAN Coverage Planning: Optimization Models and Algorithms, in proceedings of the IEEE Vehicular Technology Conference (VTC-Spring 2004), 17–19 May 2004, Volume: 4, Page(s): 2219–2223

[25] J.N. Davies, V. Grout and R. Picking, "Prediction of Wireless Network Signal Strength within a Building", 7th International Network Conference (INC 2008), Plymouth, UK, 8-10 July 2008.

[26] D. Niculescu, "Interference map for 802.11 networks". In Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement IMC '07. San Diego, California, USA, October 24 - 26, 2007.

[27] P. Fuxjager, D. Valerio, F. Ricciato, "The myth of non-overlapping channels: interference measurements in IEEE 802.11". 4th Annual Conference on Wireless On demand Network Systems and Services (IEEE/IFIP WONS'07). Oberguyrgl, Austria, Pp. 1–8. Jan. 2007.

[28] J. Padhye, S. Agarwal, V. Padmanaban, L. Qiu, A. Rao, and B. Zill, "Estimation of Link Interference in Static Multi-Hop Wireless Networks," In proceedings of the 5th Conference on Internet Measurement 2005, Berkeley, California, USA, October 19-21, 2005.

[29] J. Jun, P. Peddabachagari, and M. Sichitiu. "Theoretical Maximum Throughput of IEEE 802.11 and its Applications". In Proceedings of the IEEE International Symposium on Network Computing and Applications, pages 249-257, Cambridge, MA, April 2003.

[30] B. Bing, "Measured performance of the ieee 802.11 wireless LAN," in proceedings 26th Conference on Local Computer Networks, Lowell, Massachusetts, USA, 17-20 October, 1999.

[31] 24 inSSIDer website, at http://www.metageek.net/products/inssider [Last access: 10th May 2010]

[32] Net Meter website, at http://www.hootech.com/NetMeter/ [Last access: 10th May 2010]

[33] J. Jun, P. Peddabachagari, and M. L. Sichitiu, "Theoretical maximum throughput of IEEE 802.11 and its applications," in Proc. Second IEEE International Symposium on Network Computing and Applications (NCA 2003), (Cambridge, MA), pp. 249–256, Apr. 2003.

[34] WRT320N review. In RedesZone Forum website, at http://www.redeszone.net/routers/linksys-wrt320n-review-de-este-router-neutro-gigabit-con-wifi-n-a-doble-banda-24ghz-y-5ghz-no-simultanea/ [Last access: 3rd july 2011]

Fast Retrieval from Image Databases via Binary Haar Wavelet Transform on the Color and Edge Directivity Descriptor

Savvas A. Chatzichristofis[1] Yiannis S. Boutalis[1,2] Avi Arampatzis[1]
[1]Department of Electrical & Computer Engineering,
Democritus University of Thrace, Xanthi, Greece
[2]Department of Electrical, Electronic and Communication Engineering,
Chair of Automatic Control, University of Erlangen-Nuremberg, Germany
{schatzic,ybout,avi}@ee.duth.gr

***Abstract*—In this paper, we are evaluating several accelerating techniques for content-based image retrieval, suitable for the Color and Edge Directivity Descriptor (CEDD). To date, the experimental results presented in the literature have shown that the CEDD achieves high rates of successful retrieval in benchmark image databases. Although its storage requirements are minimal, only 54 bytes per image, the time required for retrieval may be practically too long when searching on large databases. The proposed technique utilizes the Binary Haar Wavelet Transform in order to extract from the CEDD a smaller and more efficient descriptor, with a size of less than 2 bytes per image, speeding up retrieval from large image databases. This descriptor describes the CEDD, but not necessarily the image from which it is extracted. The effectiveness of the proposed method is demonstrated through experiments performed on several known benchmarking databases.**

***Keywords*-CEDD; Binary Haar Wavelet Transform; Content-Based Image Retrieval;**

I. Introduction

As the use of computers, internet and cameras is getting more popular, efficient content-based image retrieval is more essential than ever. Any technology that, in principle, helps to organize digital image archives by their visual content is defined as content-based image retrieval (CBIR). By this definition, anything ranging from an image similarity function to a robust image annotation engine falls under the purview of CBIR [1].

Online image repositories such as Flickr contain hundreds of millions of images and are growing quickly [2]. The requirements of modern retrieval systems are not limited to providing good retrieval results, but extend to their ability for quick results. The majority of internet users would compromise the partial reduction of result accuracy in order to save time from searching.

Image retrieval, as well as text retrieval, may be described by the similarity search paradigm [3]. Efficient approaches that allow application on generic similarity search problems still need to be investigated [4]. A promising direction to address this issue is the *approximate* similarity search paradigm [5], [6], [7], [8]. Approximate similarity search provides an improvement in search performance at the price of some imprecision in the results. An interesting approach of approximate similarity search was proposed in [4]. The idea at the basis of this technique is that when two objects are very close to each other they 'see' the world around them in the same way.

In order to achieve image retrieval from large databases, the representation of images by Latent Dirichlet Allocation (LDA) [9] models is studied in [2]. Image representations are learned in an unsupervised fashion, and each image is modeled as a mixture of its depicted topics or object parts.

The present paper proposes a different approach for searching in large databases. First of all, in order to ensure quality of the results, the Color and Edge Directivity Descriptor (CEDD), proposed in [10], [11], is utilized. The size of this descriptor is 54 bytes/image. Subsequently, the Binary Haar Wavelet Transform [12] is used for the extraction of a second descriptor, we call Binary CEDD (B-CEDD). This second descriptor is employed during the retrieval procedure instead of the image. In this way, reduced retrieval times are achieved. A preliminary version of this work has been presented in [13]. Details concerning the CEDD and the Binary Haar Wavelet Transform are given in Sections II and III, respectively.

In order to shape B-CEDD, we follow and evaluate three different approaches. First, we consider CEDD as a single vector and apply the Binary Haar Wavelet Transform up to 15 coefficients. Second, we consider CEDD as a result of early fusion of two independent vectors, one capturing color and the other texture information. Also in this case, the resulting compact descriptor consists of 15 coefficients. In final third approach, we again consider CEDD as a result of early fusion of 6 independent vectors and apply Binary Haar Wavelet Transform to each of the vectors separately. The length of the resulting descriptor is now 18 coefficients. In Section IV, we will describe these three approaches in detail.

During the search process, an image query is entered and the system returns images with a similar content. Initially, the similarity/distance between the query and each image in the database is calculated with the proposed descriptor, and only if the distance is smaller than a predefined threshold, the comparison of their CEDDs is performed. The entire retrieval procedure is described in Section V. In order to estimate the appropriate threshold value, efficient techniques, described in Section VI, were used. The experimental results are presented in Section VII, and the conclusions of this study are drawn in Section VIII.

II. The Color and Edge Directivity Descriptor

The descriptors, which include more than one features in a compact histogram, can be regarded that they belong to the family of Compact Composite Descriptors. A typical example of CCD is the CEDD descriptor. The structure of CEDD consists of 6 texture areas. In particular, each texture area is separated into 24 sub regions, with each sub region describing a color. CEDD's color information results from 2 fuzzy systems that map the colors of the image in a 24-color custom palette. To extract texture information, CEDD uses a fuzzy version of the five digital filters proposed by the MPEG-7 EHD [14], [15]. The CEDD extraction procedure is outlined as follows: when an image block (rectangular part of the image) interacts with the system that extracts a CCD, this section of the image simultaneously goes across 2 units. The first unit, the color unit, classifies the image block into one of the 24 shades used by the system. Let the classification be in the color $m, m \in [0, 23]$. The second unit, the texture unit, classifies this section of the image in the texture area $a, a \in [0, 5]$. The image block is classified in the bin $a \times 24 + m$. The process is repeated for all the image blocks of the image. On the completion of the process, the histogram is normalized within the interval [0,1] and quantized for binary representation in a three bits per bin quantization.

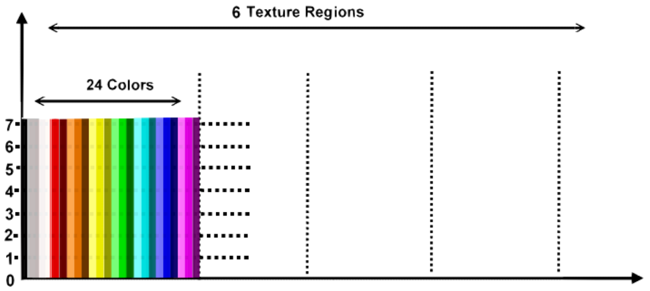


Figure 1. The structure of the CEDD.

The most important attribute of CEDDs is the achievement of very good results that they bring up in various known benchmarking image databases. Table 1 shows the ANMRR [14] results in 3 image databases, along with those obtained by MPEG-7 descriptors. The ANMRR ranges from '0' to '1', and the smaller the value of this measure is, the better the matching quality of the query. ANMRR is the evaluation criterion used in all of the MPEG-7 color core experiments. Evidence shows that the ANMRR measure approximately coincides linearly with the results of subjective evaluation of search engine retrieval accuracy. More details on the ANMRR are given in section VII. The ANMRR values for the MPEG-7 descriptors in WANG's [16] database as well as the ground truths that were used are available at [17]. Since MPEG-7 descriptor results are not available for the UCID [18] and NISTER [19] databases, an implementation of CLD, SCD and EHD in img(Rummager) [20] and LIRe Demo [21] retrieval systems is used. Details regarding the experimental results, the implementation of the MPEG-7 descriptors, as well as the ground truths that were used, are available online.

Another important attribute of CEDD is its small size requirements for indexing images. The CEDD length is 54 bytes per image.

Table I
ANMRR results in three benchmark image databases.

Descriptor	WANG	UCID	NISTER
CCD			
CEDD	**0.25283**	**0.28234**	**0.11297**
MPEG-7			
DCD MPHSM	0.39460	-	-
DCD QHDM	0.54680	-	-
SCD	0.35520	0.46665	0.36365
CLD	0.40000	0.43216	0.2292
CSD	0.32460	-	-
EHD	0.50890	0.46061	0.3332
HTD	0.70540	-	-

The img(Rummager) and LIRe Demo retrieval systems use these descriptors to create index files from which they carry out the search. Img(Rummager) makes XML-type index files, while LIRe utilizes a Lucene index to store the descriptors.

III. Binary Haar Wavelet Transform

The Binary Haar Wavelet Transform coefficients of the histogram are calculated with the use of following Haar Wavelet Transform [12]:

$$\psi(x) = \begin{cases} 1 & 0 \le x < 0.5 \\ -1 & 0.5 \le x < 1 \\ 0 & else \end{cases} \qquad (1)$$

Figure 2 shows the four basis functions of the Haar wavelet of length eight. The Haar wavelet coefficients are obtained by taking the inner product of the basis functions with the given histogram. This transformation is very fast as it does not involve multiplications.

The Haar coefficients capture the qualitative aspects of the histogram [22]. For example, the second coefficient (from the basis function 2 in Figure 2) is positive if the sum of the left half of the histogram bins is greater than the right half and negative otherwise. Similarly, the third coefficient is positive if the sum of the first quarter of the histogram bins is greater than the second quarter and negative otherwise. In the Binary Haar descriptor, each of these coefficients is quantized to ’0’ or ’1’, depending on whether its value is negative or positive, hence a binary representation is obtained.

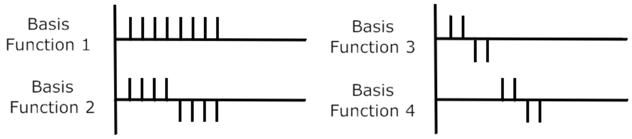


Figure 2. Four basis functions of the Haar wavelet of length eight.

At the first level, the k bins of the histogram are divided into two halves. If the sum of the histogram values in the left half is greater than the sum of the histogram values in the right half then the second bit of descriptor is ’1’, while is ’0’ otherwise. Note that the first coefficient corresponds to the sum of all probabilities in a histogram and it is always positive. Therefore is quantized to 1 and is not used in similarity matching.

At the second level, the $k/2$ bins of each half of the histogram are divided into two halves. If the sum of the histogram values in the first half is greater than the sum of the histogram values in the second half then the second bit of descriptor is ’1’ else it is ’0’. Similar, if the sum of the third half is greater than the sum in the fourth half, then the third bit of descriptor is ’1’ else it is ’0’. This is repeated recursively for the third and the fourth level.

IV. Binary CEDD (B-CEDD)

In order to describe the contents of CEDD with another descriptor significantly smaller in storage needs, we followed 3 different approaches with the use the Binary Haar Wavelet Transform.

A. Approach 1: B-CEDD$_1$

The first approach considers CEDD as a 144-dimensional vector, in which we apply 4-level Binary Haar Wavelet Transform. In the first level, the 144 elements are split in two groups of 72 elements. If the sum of the coefficients of the first group is greater or equal to the corresponding sum of the second group, then the first output coefficient of the transform is 1; otherwise, it is 0.

In the second level, the 144 elements are split in groups of 36 elements, and compared in consequent pairs using the same method as in the first level. The output of the transform consists of two coefficients. The third level of the transform compares consequent pairs of 18 elements, producing another 4 output coefficients.

Finally, the fourth level compares consequent pairs of 9 elements, producing another 8 coefficients. Overall, the output vector of the transform consists of 15 coefficients. We chose to apply the transform four times in order to produce an output vector of 2 bytes. In the rest of this paper, the result obtained from the application of the Binary Haar Transform on the CEDD descriptor will be referred as Binary CEDD with index 1 (B-CEDD$_1$).

B. Approach 2: B-CEDD$_2$

The second approach considers CEDD as the results of early fusion of two independent vectors; a vector capturing the color information of the image described by the CEDD, and a vector describing the texture information. To apply the Binary Haar Wavet Transform, we work as follows: First, we isolate the color and texture information from the descriptor in two independent vectors, the CEDD_Color vector consisting of 24 elements and the CEDD_Texture vector consisting of 6 elements. The separation is straightforward since each information item is distinctively placed in the descriptor. The separation pseudocode is the following:

```
for (int i = 0; i < 6; i++)
  {
   for (int j = 0; j < 24; j++)
      {
          CEDD_Color [j] += CEDD[24 * i + j];
      }
   }
for (int i = 0; i < 6; i++)
  {
   for (int j = 0; j < 24; j++)
    {
     CEDD_Texture [i] += CEDD[24 * i + j];
    }
   }
```

The Binary Haar Transform is applied on the CEDD_Color vector up to the third level resulting in 7 coefficients (1 coefficient from the first transformation level, 2 coefficients from the second transformation level, and 3 from the third level).

Regarding the CEDD_Texture vector, given that the resulting 2 halves include 3 elements, the problem arising is that the transform may be applied only once. In order to overcome this constraint, whenever this is met during the transform application, we propose the following solution: During the first transform level, the 6 elements are divided in 2 triads. To apply the second level, the middle element of each triad is cloned. The 2 identical elements replace the original element from which they came from. In this way, each triad is replaced by a quartet of elements, which now the transform may be applied on. In the third level of the Binary Haar Transform the cloned elements are removed and the transform is applied directly on the vector, comparing this time the elements per pair. On the whole, 8 elements are created (1 from the first transform level, 4 from the second level and 3 from the third). The complete extraction procedure of the Modified Binary Haar Wavelet Transform from CEDD_Texture is illustrated in Figure 3.

At the end of the procedure, the 2 resulting vectors are placed consecutively. The size of the produced descriptor is

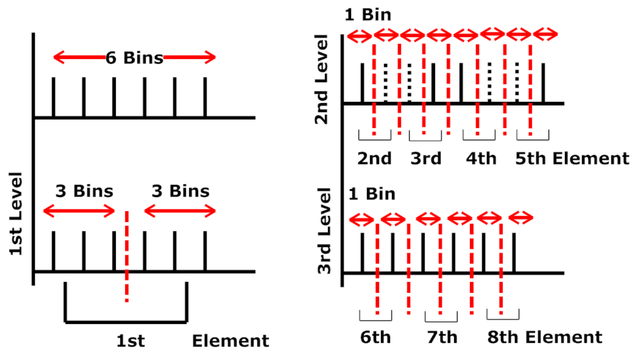


Figure 3. Extraction procedure of the Modified Binary Haar Wavelet Transform from CEDD_Texture.

limited to 15 binary bins and its storage requirements are much smaller than 2 bytes per image (15 bits). In the rest of this paper, this descriptor will be referred to as Binary CEDD with index 2 (B-CEDD$_2$).

C. Approach 3: B-CEDD$_3$

The third approach splits CEDD to 6 independent vectors in 24 dimensions. Each vector corresponds to one of the 6 texture areas used by the descriptor. The transform is applied twice to each of the vectors, splitting the vectors to groups of 12 and then to 6. From each vector, 3 coefficients are produced. We produce B-CEDD$_3$ by taking consequently the 18 output coefficients ($3 \times 6 = 18$). This approach results to the largest descriptor size and is closer to the essence of Compact Composite Descriptors where CEDD belongs to, but as we will see in Section VII it does not lead to the best end-results.

V. System Overview

One of the most important attributes of the Binary CEDD is that it is extracted directly from the CEDD with no intervention of the described image. This results in its immediate extraction from the already existent index files.

The search procedure based on the use of the 2 descriptors, CEDD and B-CEDD, is illustrated in Figure 4. The user enters a query image in the system. From this image, both the CEDD and the B-CEDD descriptors are extracted. The system uses an image database in which the indices are described by both descriptors. During data retrieval from databases, the length of the retrieved information is of great significance [23]. For this reason, in a first phase the system retrieves only the B-CEDD descriptor, which, due to its small storage requirements as well as its small length, is retrieved practically in an instance.

For each database image, the distance between the B-CEDD descriptor with the corresponding B-CEDD descriptor of the query image is calculated by a simple X-OR gate. In the case of 2 identical descriptors, the X-OR output is equal to 0, while in the worst scenario the obtained distance is 15 (equal to the B-CEDD length). The logic gate X-OR requires much less computing resources than the Tanimoto coefficient. The Tanimoto coefficient is used to calculate the distance between CEDD descriptors.

If the distance of the 2 descriptors is found to be smaller than a threshold T, then the CEDD descriptor is retrieved from the database and its distance from the corresponding CEDD descriptor of the query image is calculated. The procedure is repeated for all the database images.

After the completion of the procedure, the classified results are shown to the user in ascending order of the distance obtained during the CEDD descriptors comparison.

The most important issue that should be taken into consideration during the aforementioned procedure is the determination of the threshold T. This threshold defines whether an image is potentially similar to the query image. If it is, then the retrieval of the image's CEDD is requested and the process for its comparison with the corresponding CEDD of the query image is activated. The T value investigation is described in detail in the following section.

VI. Investigating the T Value

In order to determine an appropriate T value, we work as follows: We choose 35 images from the Wang database and regard them as historical queries. The historical queries idea comes from the text retrieval area and has been used to normalize retrieval scores of documents in cases of fusion and distributed information retrieval [24], [25].

For each one of the historical queries, searching is performed in the database from which they originate. In particular, the distances between the B-CEDD descriptors of each historical query and each image of the database are calculated and a ranking list, in which the database images are ordered according to their distance from the historical query, is obtained. The procedure is repeated for all the historical queries. At the end of the process, 35 ranking lists emerge. Since the Wang database includes 1000 images, 35000 values (35 ranking list × 1000 images) are finally obtained. By plotting these values the distance distribution is obtained. As depicted in Figure 5, which shows score distributions from the three B-CEDD approaches, the first two approaches produce similar distributions.

Subsequently, the set of these 35000 values for each approach, enters in a Gustafson Kessel fuzzy classifier [26]. The Gustafson Kessel is an extension of the Fuzzy C-Means algorithm. The Gustafson Kessel parameters are selected as: Clusters=4, Repetitions=3000, $e = 0.001$ and $m = 2$.

The four resulting classes were used to form a single input fuzzy system for each approach. The fuzzy system includes four membership functions which are labeled as: 'Low', 'Medium', 'High' and 'Highest' The centers of the classes, as they result from Gustafson Kessel classifier, correspond to the tops of the membership functions. Given that the score distribution of the B-CEDD$_1$ and B-CEDD$_2$ are similar to

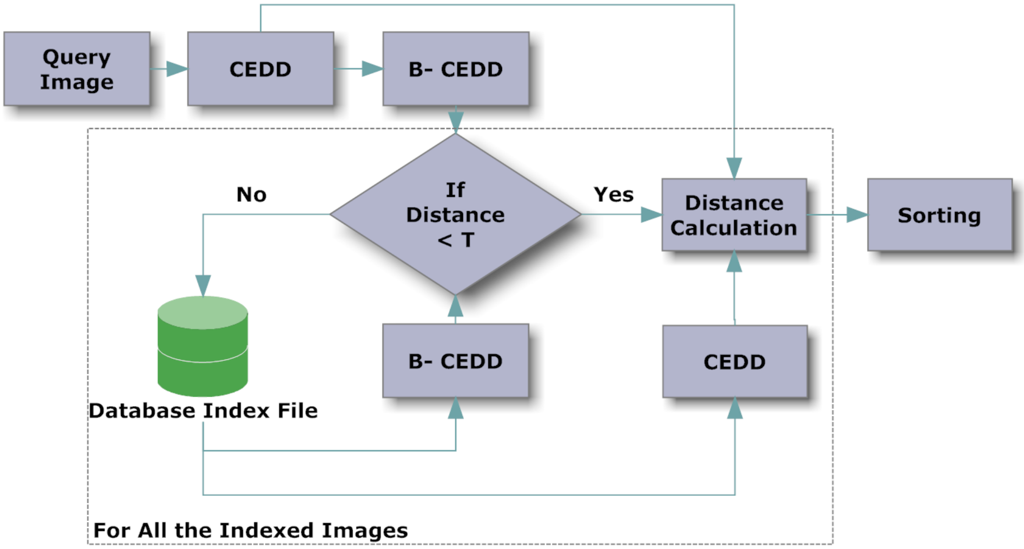


Figure 4. Searching using the 2 descriptors.

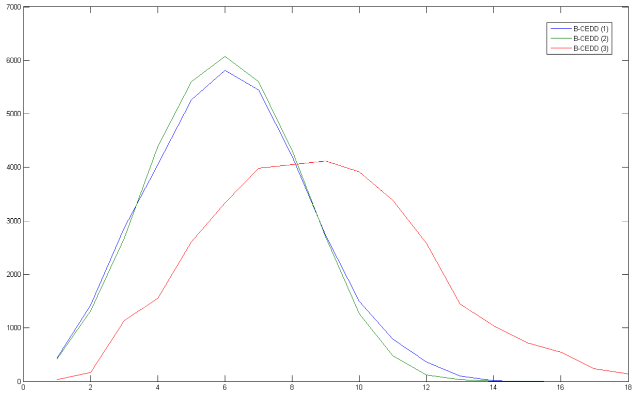


Figure 5. Distribution of the B-CEDD distances for 35 historical queries.

each other, the fuzzy systems that come from the Gustafson Kessel classifier output are identical.

The fuzzy system that was shaped operates as follows: The system gets as input the distance between the B-CEDD descriptors of the query image and any other image. The vertical axis shows the distance that may be obtained during the comparison of the 2 B-CEDD descriptors, while the horizontal axis shows the activation degree for the membership function of each class. Consider, for instance, that the distance between 2 B-CEDD descriptors was found to be equal to 4 (see Figure 6). This value activates both the first and the second membership function by 0.5.

For the simplest scenario of the model usage we should specify that if the 'Low' membership function is activated with a value greater than the activation value of any other membership function, then the image under study is likely to be visually similar to the query image. Thus, the CEDD descriptor should be also retrieved in order to perform the comparison.

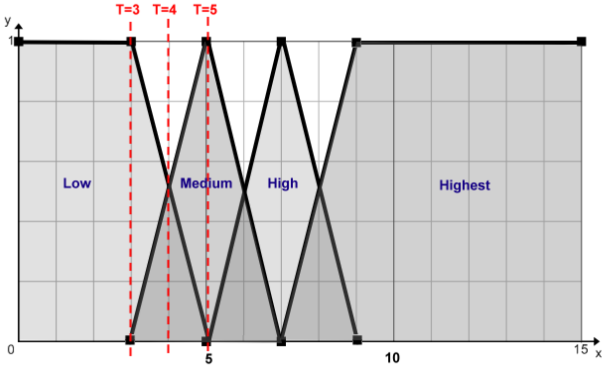


Figure 6. Outcome of the historical queries distances distribution classification in 4 classes.

Similarly, if the 'Low' activation degree is smaller than that of another membership function, the image is discarded. In the next section, the experimental results of a threshold tuning attempt are presented.

VII. Experimental Results

To the best of our knowledge, there is no large scale image database with ground truths data available that could be used for the performance evaluation of the proposed method. For this reason, experiments have been performed on two known small scale benchmarking databases.

For the performance evaluation the following measures were used:

1) Recall at n, where n is the number of the retrieved through the proposed method images. This measure

presents the number of relevant documents retrieved by a search divided by the total number of existing relevant documents.

2) ANMRR at n. The ANMRR ranges from 0 to 1. The smaller the value of this measure is, the better the retrieval quality. This measure captures both precision and recall in one value.

The ANMRR computation requires the average rank computation. The average rank AVR(q) for query q is:

$$AVR(q) = \sum_{k=1}^{NG(q)} \frac{Rank(k)}{NG(q)} \tag{2}$$

where

- $NG(q)$ is the number of ground truth images for the query q.
- K is the top ranked retrievals examined where:

$$K = min(X \times NG(q), 2 \times GMT) \tag{3}$$

$$GMT = max\left(NG\left(q\right)\right) \tag{4}$$

If $NG(q) > 50$ then $X = 2$ else $X = 4$.

$Rank(k)$ is the retrieval rank of the ground truth image. For a query q, suppose that as a retrieval result the k^{th} ground truth image is found at a position R. If this image is in the first K results then $Rank(k) = R$ else $Rank(k) = K + 1$.

The modified retrieval rank is:

$$MRR(q) = AVR(q) - 0.5 - 0.5 \times N(q) \tag{5}$$

The normalized modified retrieval rank is computed as follows:

$$NMRR(q) = \frac{MRR(q)}{K + 0.5 - 0.5 \times N(q)} \tag{6}$$

The average of NMRR over all queries is defined as:

$$ANMRR(q) = \frac{1}{Q} \sum_{q=1}^{Q} NMRR(q) \tag{7}$$

The proposed method is implemented in the retrieval system img(Rummager) [20]. For better time measurements, each experiment was repeated 10 times. All the experiments were performed on an Intel Core 2 Quad Q9550 @2.83GHz PC with 3GB of RAM.

As elaborated in the previous section, the necessity for accurate estimation of the T threshold value which defines whether the CEDD descriptor of an image should be retrieved, is imperative. In order to determine the appropriate threshold, we experiment with the following scenarios: Initially, for each of the approaches of B-CEDD, we consider that the two images may be potentially similar (and calculate the distance between the two CEDD descriptors) if the distance of their B-CEDD descriptors activates the membership function ‘Low’ with value equal to 1. This threshold is defined as $T1$.

According to the second scenario, two images may be potentially similar if the distance of their B-CEDD descriptors activates the membership function ‘Low’ with value greater than 0.5. This threshold is defined as $T2$. At the end, according to the third scenario, two images may be potentially similar if the distance of their B-CEDD descriptors activates the membership function ‘Low’ with value greater than 0. This threshold is defined as $T3$.

The values of T for all the approaches are given in Table II, while in the case of B-CEDD$_1$ and B-CEDD$_2$ the threshold values are depicted in Figure 6 with a dashed line.

Table II
THRESHOLD VALUES

	$T1$	$T2$	$T3$
B-CEDD$_1$ Thresholds	3	4	5
B-CEDD$_2$ Thresholds	3	4	5
B-CEDD$_3$ Thresholds	5	7	8

Initially, our experiments were performed using Wang database. The Wang database is a subset of 1000 manually-selected images from the Corel stock photo database and forms 10 classes of 100 images each. In particular, the queries and ground-truth proposed by the MIRROR[17] image retrieval system are used. MIRROR separates the WANG database into 20 queries.

For each of these queries, the time of the retrieval through the proposed system is measured, as well as the time required when only the CEDD descriptor is used. Table III illustrates the mean results for the 20 queries of the Wang database for all the approaches and all the T values. The Recall index represents the ratio of the correct image retrievals for each query (images that belong to the ground truth of the query) to the size of the ground truth. Therefore, the Recall @ n describes the percentage of the correct images that were retrieved for all the queries. On the other hand, the ANMRR index evaluates the order in which the results were placed after the completion of the procedure. Thus, in order to assess the systems effectiveness properly both measures should be taken into account.

Considering these results, it can be readily observed that the proposed method improves the searching procedure times significantly. For $T1$, all approaches are capable of retrieving almost 112,000 images per second. But in all three approaches both Recall @ n and ANMRR have a very small value. This means that a lot of images from the ground truth were absorbed during the retrieval procedure. By comparing the three approaches, B-CEDD$_1$ seems to perform better.

The threshold $T2$, which retrieves almost 83,333 images

Table III
EXPERIMENTS ON THE WANG DATABASE

CEDD Time	45.2ms		
	$T1$	$T2$	$T3$
B-CEDD$_1$ Time	10.0ms	13.1ms	19.0ms
B-CEDD$_2$ Time	9.1ms	12.2ms	19.0ms
B-CEDD$_3$ Time	9.1ms	13.7ms	19.1ms
B-CEDD$_1$ Retrieved Im.	242.95	404.35	562.20
B-CEDD$_2$ Retrieved Im.	218.40	381.45	563.15
B-CEDD$_3$ Retrieved Im.	212.20	445.75	599.65
B-CEDD$_1$ Recall @ n	**0.6476**	**0.8188**	**0.9234**
B-CEDD$_2$ Recall @ n	0.5441	0.7074	0.8675
B-CEDD$_3$ Recall @ n	0.4971	0.7715	0.8704
CEDD ANMRR	0.2528		
B-CEDD$_1$ ANMRR	**0.4011**	**0.3100**	**0.2636**
B-CEDD$_2$ ANMRR	0.4836	0.3747	0.2902
B-CEDD$_3$ ANMRR	0.5322	0.3397	0.2824

Table IV
EXPERIMENTS ON THE UCID DATABASE

CEDD Time	60.3ms		
	$T1$	$T2$	$T3$
B-CEDD$_1$ Time	22.7ms	25.2ms	34.0ms
B-CEDD$_2$ Time	19.5ms	25.9ms	33.3ms
B-CEDD$_3$ Time	16.0ms	18ms	27.4ms
B-CEDD$_1$ Retrieved Im.	550.13	785.68	1007.61
B-CEDD$_2$ Retrieved Im.	472.08	825.04	987.37
B-CEDD$_3$ Retrieved Im.	379.14	618.90	866.44
B-CEDD$_1$ Recall @ n	**0.8920**	**0.9457**	**0.9734**
B-CEDD$_2$ Recall @ n	0.7990	0.8926	0.9478
B-CEDD$_3$ Recall @ n	0.8349	0.9424	0.9653
CEDD ANMRR	0.2823		
B-CEDD$_1$ ANMRR	**0.2971**	**0.2833**	**0.2823**
B-CEDD$_2$ ANMRR	0.3258	0.2905	0.2831
B-CEDD$_3$ ANMRR	0.3140	0.2874	0.2848

per second and its performance is found satisfactory for both the Recall @ n and the ANMRR, could be considered as the 'golden-mean' solution. Compared to the CEDD method, the proposed method with $T2$ retrieves almost four times more images per second. Also in this case, B-CEDD$_1$ seems to advance, although it requires slightly more time than the B-CEDD$_2$ (8.33% more time / 1 ms), it improves the Recall @ n by 16.6% compared to the performance of B-CEDD$_2$. Better performance is observed also in ANMRR where B-CEDD$_1$, having the value of 0.31, is better than B-CEDD$_3$ by 8.74%.

With $T3$, good results were achieved from all the approaches for both the Recall @ n and the ANMRR but the searching time was over doubled in comparison to the corresponding time for $T1$. Also in this case B-CEDD$_1$ performs better than the other two approaches, with its value of Recall @ $n = 0.9234$ approaching the CEDD performance, consider that the value of ANMRR is smaller by 4% from the corresponding value that the CEDD descriptor presents.

In the sequel, experiments were performed using the UCID Database. The UCID database was created as a benchmark database for CBIR and image compression applications. This database currently consists of 1338 uncompressed TIFF images on a variety of topics including natural scenes and man-made objects, both indoors and outdoors. All the UCID images were subjected to manual relevance assessments against 262 selected images, creating 262 ground truth image sets for performance evaluation. The results for all the three B-CEDD extraction approaches, for all the T values are illustrated in Table IV.

In this database we observe that even the smallest values of T retrieve more images compared to the images that were retrieved for the corresponding values of T on Wang database. We also observe that even for very small T, as for $T1$, there are good results for Recall @ n from all the three approaches that we used. Also in this case, B-CEDD$_1$ performs way better than the other two approaches, showing improvement from the B-CEDD$_3$ by 6.83%, while the improvement from B-CEDD$_2$ equals to 11.64%. B-CEDD $_1$ ANMRR value is by 5.24% smaller than the corresponding value of the CEDD descriptor, but the system retrieves twice as many images in the same time.

For $T2$, B-CEDD$_1$ performance is approaching the performance of CEDD. With Recall @ n value at 0.9457, ANMRR presents deviation from the corresponding CEDD value by just 0.35%. Significant improvements also presented by the other two approaches.

Finally, for $T3$, the performance of B-CEDD$_1$ is identical to the CEDD performance. At the same time, the other two methods also improve their performance, with B-CEDD$_2$ to behave slightly better than B-CEDD$_3$.

Observing the results in the two databases, is obvious that the B-CEDD$_1$ approach performs better than the other two approaches. Considering the threshold value, is obvious that by $T2$, a satisfactory trade-off between the acceleration rate of the retrieval procedure and the performance rate of the system is ensured. But how important is the reduction in the results? In order to quantize the reduction that is observed, we use a significance test. Significance tests tell us whether an observed effect, such as a difference between two means or a correlation between two variables, could reasonably occur 'just by chance' in selecting a random sample [27]. We used a bootstrap test, one-tailed, at significance levels 0.05(*), 0.01 (**), and 0.001 (***), against the CEDD results baseline in UCID database . The significance test was applied on Mean Average Precision (MAP):

$$MAP = \frac{1}{|Q|}\sum_{q\in Q} AP(q) \tag{8}$$

where Q is the set of queries q.

$$AP(q) = \frac{1}{N_R}\sum_{n=1}^{N_R} P_Q(R_n) \tag{9}$$

where R_n is the recall after the nth relevant image was retrieved. N_R is the total number of relevant documents for the query.

Table V
SIGNIFICANCE TEST RESULTS

CEDD MAP	0.6748		
	$T1$	$T2$	$T3$
B-CEDD$_1$ MAP	**0.6598****	**0.6720-**	**0.6748-**
B-CEDD$_2$ MAP	0.6324***	0.6639**	0.6730-
B-CEDD$_3$ MAP	0.6747***	0.6685**	0.6720-

Observing the results of Table V, we conclude that all three approaches for $T3$ have non-significant reductions in their results. On the other hand, for $T2$ B-CEDD$_1$ is the only approach which has a non-significant reduction. The significance test results support the conclusion that the best method is B-CEDD$_1$ for $T2$.

Finally, given that the proposed descriptor is an MPEG-7 like descriptor, the schema of the B-CEDD as an MPEG-7 extension is described as follows:

```
<?xml version ="1.0" encoding="UTF-8"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema"
       xmlns:mpeg7="urn:mpeg:mpeg7:schema :2004"
        xmlns:SpCDNS="B-CEDDNS"
                     targetNamespace="B-CEDDNS">
  <import namespace="urn:mpeg:mpeg7:schema:2004"
               schemaLocation="Mpeg7-2004.xsd"/>
  <complexType name="B-CEDDType" final="#all">
    <complexContent>
      <extension base="mpeg7:VisualDType">
       <sequence>
         <element name="value">
          <simpleType>
           <restriction>
            <simpleType>
              <listitemType="mpeg7:Binary"/>
            </simpleType>
             <length value="15"/>
           </restriction>
          </simpleType>
         </element>
       </sequence>
      </extension>
    </complexContent>
  </complexType>
</schema>
```

VIII. CONCLUSIONS

We proposed an extension of the Color and Edge Directivity Descriptor which improves the speed efficiency of the CEDD. Through the application of the Modified Binary Haar Wavelet Transform on the CEDD, the proposed method achieves the extraction of a second, smaller (15 bits length), descriptor. Essentially, each CEDD descriptor is described by another compact binary descriptor. During the image searching process, the compact versions of the descriptors are employed, and only when their distance is smaller than a given threshold the searching continues with the CEDD. The distance between the B-CEDD descriptors is calculated by using a simple X-OR gate. The logic gate X-OR has much less computational cost than the Tanimoto coefficient. One of the most important attributes of the Binary CEDD (B-CEDD) is that it is extracted directly from the CEDD, without the need of the described image. This enables its immediate extraction from pre-existing index files. The effectiveness of the proposed method was demonstrated through experiments. Finally, it is worth noting that the proposed method can be applied to all Compact Composite Descriptors [23], [28], [29].

REFERENCES

[1] R. Datta, D. Joshi, J. Li, and J. Wang, "Image retrieval: Ideas, influences, and trends of the new age," *ACM Computing Surveys*, vol. 40(2), pp. 1–60, 2008.

[2] E. Hörster, R. Lienhart, and M. Slaney, "Image retrieval on large-scale image databases," in *Proceedings of the 6th ACM international conference on Image and video retrieval*. ACM, 2007, pp. 17–24.

[3] H. Jagadish, A. Mendelzon, and T. Milo, "Similarity-based queries," in *Proceedings of the fourteenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems*. ACM, 1995, pp. 36–45.

[4] G. Amato and P. Savino, "Approximate similarity search in metric spaces using inverted files," in *Proceedings of the 3rd international conference on Scalable information systems*. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008, pp. 1–10.

[5] P. Zezula, P. Savino, G. Amato, and F. Rabitti, "Approximate similarity retrieval with m-trees," *The VLDB Journal*, vol. 7, no. 4, pp. 275–293, 1998.

[6] P. Ciaccia and M. Patella, "Pac nearest neighbor queries: Approximate and controlled search in high-dimensional and metric spaces," in *ICDE*. Published by the IEEE Computer Society, 2000, p. 244.

[7] G. Amato, F. Rabitti, P. Savino, and P. Zezula, "Region proximity in metric spaces and its use for approximate similarity search," *ACM Transactions on Information Systems (TOIS)*, vol. 21, no. 2, pp. 192–227, 2003.

[8] H. Ferhatosmanoglu, E. Tuncel, D. Agrawal, and A. El Abbadi, "Approximate nearest neighbor searching in multimedia databases," in *Proceedings Of The International Conference On Data Engineering*. Citeseer, 2001, pp. 503–514.

[9] D. Blei, A. Ng, and M. Jordan, "Latent dirichlet allocation," *The Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.

[10] S. Chatzichristofis and Y. Boutalis, "Cedd: Color and edge directivity descriptor: A compact descriptor for image indexing and retrieval," *LNCS, Computer Vision Systems*, pp. 312–322, 2008.

[11] S. A. Chatzichristofis, K Zagoris, Y. S. Boutalis and N. Papamarkos, “Accurate image retrieval based on compact composite descriptors and relevance feedback information,” *International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI)*, vol. 24 (2), pp. 207–244, 2010.

[12] Z. Struzik and A. Siebes, “The haar wavelet transform in the time series similarity paradigm,” *Principles of Data Mining and Knowledge Discovery*, pp. 12–22, 1999.

[13] S. A. Chatzichristofis, Y. Boutalis, and A. Arampatzis, “Accelerating image retrieval using binary haar wavelet transform on the color and edge directivity descriptor,” in *The Fifth International Multi-Conference on Computing in the Global Information Technology (ICCGI 2010)*, 2010, pp. 41–47.

[14] B. Manjunath, J. Ohm, V. Vasudevan, A. Yamada *et al.*, “Color and texture descriptors,” *IEEE Transactions on circuits and systems for video technology*, vol. 11, no. 6, pp. 703–715, 2001.

[15] C. Won, D. Park, and S. Park, “Efficient use of mpeg-7 edge histogram descriptor,” *Etri Journal*, vol. 24, no. 1, pp. 23–30, 2002.

[16] J. Wang, J. Li, and G. Wiederholdy, “Simplicity: Semantics-sensitive integrated matching for picture libraries,” *Advances in Visual Information Systems*, vol. 1929/2000, pp. 171–193, 2000.

[17] K. Wong, K. Cheung, and L. Po, “Mirror: an interactive content based image retrieval system,” in *IEEE International Symposium on Circuits and Systems*, vol. 2. IEEE; 1999, 2005, pp. 1541–1544.

[18] G. Schaefer and M. Stich, “Ucid-an uncompressed colour image database,” *Storage and Retrieval Methods and Applications for Multimedia 2004*, vol. 5307, pp. 472–480, 2004.

[19] D. Nister and H. Stewenius, “Scalable recognition with a vocabulary tree,” in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 2. Citeseer, 2006, pp. 2161–2168.

[20] S. A. Chatzichristofis, Y. S. Boutalis and M. Lux, “Img(rummager): An interactive content based image retrieval system,” in *2nd International Workshop on Similarity Search and Applications (SISAP)*, 2009, pp. 151–153.

[21] M. Lux and S. Chatzichristofis, “Lire: lucene image retrieval: an extensible java cbir library,” in *Proceeding of the 16th ACM international conference on Multimedia*. ACM, 2008, pp. 1085–1088.

[22] S. Krishnamachari and M. Abdel-Mottaleb, “Compact color descriptor for fast image and video segment retrieval,” in *IS&T/SPIE Conference on Storage and Retrieval of Media Databases*, 2000, pp. 581–589.

[23] S. A. Chatzichristofis, Y. S. Boutalis and M. Lux, “Spcd - spatial color distribution descriptor. a fuzzy rule based compact composite descriptor appropriate for hand drawn color sketches retrieval,” in *2nd International Conference on Agents and Artificial Intelligence (ICAART)*, 2010, pp. 58–63.

[24] A. Arampatzis and J. Kamps, “A signal-to-noise approach to score normalization,” in *Proceeding of the 18th ACM conference on Information and knowledge management*. ACM, 2009, pp. 797–806.

[25] M. Fernández, D. Vallet, and P. Castells, “Using historical data to enhance rank aggregation,” in *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2006, pp. 643–644.

[26] D. Gustafson and W. Kessel, “Fuzzy clustering with a fuzzy covariance matrix,” in *1978 IEEE Conference on Decision and Control including the 17th Symposium on Adaptive Processes*, vol. 17, 1978, pp. 761–766.

[27] D. Moore and L. Sorenson, *Introduction to the Practice of Statistics SPSS Manual*. WH Freeman, 2005.

[28] S. Chatzichristofis and Y. Boutalis, “Fcth: Fuzzy color and texture histogram-a low level feature for accurate image retrieval,” in *Image Analysis for Multimedia Interactive Services, 2008. WIAMIS’08. Ninth International Workshop on*, 2008, pp. 191–196.

[29] Chatzichristofis, S.A. and Boutalis, Y.S., “Content based radiology image retrieval using a fuzzy rule based scalable composite descriptor,” *Multimedia Tools and Applications*, vol. 46, pp. 493–519, 2010.

Combining Biometrics Derived from Different Classes of Nonlinear Analyses of Fronto-Normal Gait Signals

T. K. M. Lee
School of IT, Monash University
Sunway Campus, Malaysia
School of EEE, Singapore Polytechnic,
Singapore
tlee@sp.edu.sg

S. Sanei
Faculty of Eng. & Physical Sciences,
University of Surrey, Surrey, UK
s.sanei@surrey.ac.uk

M. Belkhatir
Faculty of Computer Science,
University of Lyon & CNRS, France
mohammed.belkhatir@univ-lyon1.fr

Abstract- With the advent of low cost high powered computing, cameras need not just be used to record multimedia data. Cameras become sensors as we process waveforms of gait signals from the video content of humans walking towards these cameras. This sensory data allows cameras to be incorporated into networks that monitor humans and their movements. This work introduces a novel analysis of gait for human recognition which uses and can be used for surveillance. Current approaches in human gait analyses employ linear signal decomposition techniques to obtain features such as frequency and phase. In contrast, we establish the nonlinear nature of fronto-normal (FN) gait. This motivates for the use of nonlinear analyses on FN gait as a biometric and opens up new avenues for research in gait recognition. Using these nonlinear analyses to derive features, we show that by themselves they may not provide sufficient discriminating ability. But by a novel combination of two different nonlinear measures, one exploiting chaosity and another representing regularity, this can be used to identify a person using gait. We apply this in a multi-biometric experiment to demonstrate its effectiveness.

Keywords-gait, nonlinear; chaos; Hilbert Huang Transform; EMD

I. INTRODUCTION

Due to the current security climate, the presence of multimedia devices such as low cost webcams and security cameras are well nigh ubiquitous, whether in points of access or traffic. But these cameras may not just be used to capture image and video data as the information can be used to *sense* the environment and be processed to produce sensory data. This is important as cameras are being deployed in networks to so that multiple views and interpretations of a scene can provide a more robust analysis of the same. For example, the static images that come from a camera can be used for face recognition which in effect, senses the presence of a particular person. Considering the video component of the data, it is more than just a stream of static images. They incorporate a temporal dimension which can be used to derive time-based features such as frequency of the movement of the limbs while walking which comprises the gait of a person. Effectively, this makes the camera a gait sensor, which does not require attachment to a person. Gait or the manner of walking of a person, is a biological feature - its fundamental properties have been established in medical studies.

Recently, gait has been considered as a biometric which is a registered biological trait, used in human identification. Gait includes static features such as height, stride length and silhouette bounding box lengths. Some dynamic features of gait are frequency domain parameters like frequency and phase of a walk. As a biometric, gait has desirable properties, primarily because it is hard to disguise, as in normal circumstances gait movement is involuntary. Furthermore, it can be used at long distances, and it is non-intrusive and non-invasive. In the literature, the main gait recognition approaches analyze walking which proceeds in a plane parallel to a camera, the so-called fronto-parallel (FP) view. This gives the largest variation in a silhouette from which time series data is obtained for analysis. From a far distance, this is advantageous. However, being able to obtain these silhouette images from a far distance require a clear, uncluttered field of view.

As a contrast, a very common scenario is when people queue up to access a facility. In a corridor like structure, we assume that a subject is approaching a camera. In such situations gait can be used as a supporting biometric because as the subject draws nearer, other biometrics such as face or iris can be used for robust recognition. Motion in this plane which is perpendicular to the FP view, is the fronto-normal view (FN) which is considered as a special case of FP gait. Depending on the type of analysis need, in a FP walk, at least two cycles or four steps are needed. For more robust estimation of the period of walking, about 8 m is recommended [1]. To capture this movement, the camera distance required is about 9 m [2]. This is because current video cameras typically have a focal length and sensor size of 8 mm and 1/2" respectively. Practically, having such a wide uncluttered space is difficult, since whenever we want to measure a person's gait, many people and objects will be present.

In a FN view, we can still use the 8 m. But this time, we cover twelve steps and we only need a corridor-like structure, the width being about that of a human body. Therefore, a considerable amount of space is saved as shown in Figure 1 in this case, by 2/9. Besides the considerable advantages in savings of physical space and better viewpoint, Lee et al. have put forth the advantages of the monocular FN non-silhouette approach as [3]:

i) Smaller physical space is needed.
ii) Multiple subjects can be tracked.
iii) Other biometrics can be easily combined.
iv) Wide variety of time analysis including non-periodic motion analysis can be used.

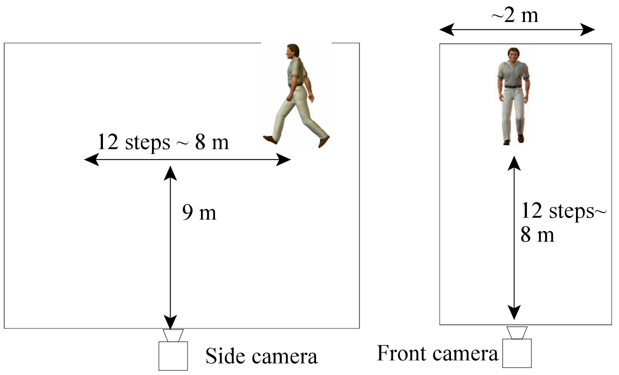


Figure 1. FP vs FN - physical dimensions for video capture

This approach has its own unique challenges when fast and reliable recognition is necessary. There are some recent surveys on gait recognition such as that done by Lee et al. [4] Most gait recognition approaches use a combination of static and dynamic gait features. Dynamic features are usually linear, like frequency of walking. Using time series parametric models for gait recognition is fairly recent, for example by Veeraghavan et al. in [5]. Lee et al. showed that chaotic measures may be used to help identify people by their gait [6]. In another paper, they show that FN gait data is mainly stationary [7]. Recently, they have also combined nonlinear measures to provide a more robust gait recognition process [8]. With respect to that publication [8] this paper renders a more complete treatment of the approach.

Many analyses of gait data assume the property of linearity without testing for it. Linearity refers to how the data may be generated by the scaled linear sums of input signals. Elaborating on this concept leads ultimately to convolution, a linear operation much used in signal processing. In the main, dynamic features of gait have been obtained by linearly decomposing gait signals via the Fourier transform, which is extensively used and has a good mathematical foundation. However, most analyses do not check that the signal is linear and stationary in nature. The gait signal is assumed to be statistically stationary. However, most biological signals are not so well specified, many studies showing that they are nonlinear and nonstationary especially in the FN. Based on biological evidence and using our FN gait dataset, Lee et al. have shown that dynamic gait data is in fact nonlinear and thus should be analyzed using nonlinear methods [9]. Applying Fourier-based decomposition to nonlinear and nonstationary signals produce mathematically correct functions, but these may not have any physical meaning at all. These signal constituents serve only to accommodate the lack of linearity and stationarity. This limits the use of such analyses in processing the signal. There are many methods to decompose data so it can be expressed in terms of components that combine in a linear way. For each of them, there are much more ways to do so nonlinearly as described by Tong, which gives rise to a rich source of features to be used in pattern recognition [10]. In this paper, we look at two of the more prominent methods of nonlinear analyses to derive features for recognition. We combine these features to increase their robustness for this task. The novelty of our approach lies in the use *and* fusion of nonlinear features for the recognition task. We demonstrate the efficacy of our approach in an experiment. In Section II, we look at the current state of temporal gait analysis. Section III covers our setup and preliminary results, Section IV describes the theory behind using Chaos Theory and the Hilbert Huang Transform (HHT) for analyzing FN gait. Section V shows some results with preliminary analysis and we conclude with Section VI.

A. Overview of temporal Gait Analysis

Psychophysical experiments using Moving Light Displays (MLDs) attached to humans have shown the possibility of using gait for identification purposes. Some time ago, Johansson showed how the patterns traced by MLDs can be perceived as that of people walking [11]. Cutting and Kozlowski showed that identification was possible from MLDs [12]. Recently, Troje has shown that the task of recognizing gender from MLDs has a lower error rate using a frontal view [13].

Gait as a biometric can be used at long distances, is non-intrusive, non-invasive, and is hard to disguise. From the medical literature (such as [1]), gait information is obtained via sensors directly attached to the body. With image processing, gait is derived from the 2D image projected on a camera sensor. In this section, we provide a brief overview of human recognition using gait with time-based features. Gait includes static features such as height, stride length and silhouette bounding box lengths. Some dynamic features of gait are frequency domain parameters like frequency and phase of a walk, which also includes the bounding box of the walking silhouette. In the literature, the area of gait analysis and recognition has involved medical analyses looking for exact movement of body parts to detect pathological conditions. Rather than standard approaches which use body silhouettes as ably described by Nixon et al., we consider the motion of individual body parts like hands and feet [14]. These produce biologically based spatio-temporal signal features which can be used as a biometric.

Much of the current gait analyses use silhouettes in the FP view because of the large changes in shape and most of these analyses assume that the signal derived from gait are linear and stationary for the sake of simplicity. Linearity refers to how the data may be described by the scaled, linear sums of input signals. In what follows, we consider a signal, which can be considered a set of time series data $\{x(t_i)\}$ for $i = 1..N$ sample points. In general, we say s is in a linear signal space if :

$$s(t) = \sum_{k=-\infty}^{\infty} h(k)\ x(t-k) \qquad (1)$$

where *h(k)* are constants and the inputs $x(t)$ may be generated by functions $f(t)$. That is, the data may be generated by the scaled, linear sums of input signals. The equation is cast as a convolution which is a common linear operation.

Stationarity entails having the statistical properties of the signal up to the second order to be constant in time. That is, $k_p = E[x^p(t)]$ where k_p is a constant, $E[\cdot]$ is the statistical expectation operator and p is the order. Usually the first order statistic is the mean, and the second order statistic the variance but sometimes the autocovariance or autocorrelation function of the signal. These measures have been used to analyze gait by decomposing the signal into its constituents. A common linear example, which has been well developed is the Fourier expansion, where the $f(t)$'s are sinusoidal functions. Current approaches have used this successfully because the FP view of gait is particularly amenable to linear analysis in contrast to the FN gait, as shown by Lee et al. [8] This motivates a search for nonlinearity and nonstationarity in descriptions of data which can be used as biometrics. In this work, they show that FN gait can be characterised by nonlinear measures. A variety of time series analyses from the fields of econometrics and physics may be employed to further characterize the gait. In contrast, FP gait yields mainly periodic measures.

Ibrahim et al. have used Empirical Mode Decompostion (EMD) (described in Section III.C) to detect the *type* of gait of a subject using a 3D accelerometer using the energy of its Intrinsic Mode Functions (IMF) [15]. From the same research group, Wang et al. have looked into various features based on IMFs and the features associated with Hilbert spectra for *clinical* gait analysis [16].

Kuchi et al. have used EMD for gait recognition. But we note that they use motion capture equipment, where the coordinates of markers attached to the body are computed in-camera at data rates of 120 samples/s [17]. Thus, the cameras are not designed to give video information. They analyze the signal for one walk cycle and for one marker, giving encouraging results. However, they do not analyze their data to provide justification for using nonlinear, nonstationary analyses. We also feel that extending the results to ordinary video cameras that can be used in security checkpoints is difficult.

II. Experimental Setup

We used a commercial video camera with a capture rate of 25 frames/s at 720 by 480 resolution. In gait recognition from video, we use feature points that have more motion in the camera plane. This would be the hands, feet, and knees for a FP walk. For a FN walk this is also true, although the motions are smaller in magnitude. For the two kinds of walk, we show the coloured marker set up in Figure 2. The marker designations are: *lh/rh* - left/right hand: *lf/rf* - left/right foot and *lk/rk* - left/right knee. Two additional discs of the same colour are attached at the waist and face level which are used for distance normalization. They are: *tm/bm*, the top/bottom markers. The markers are tracked using the CAMSHIFT algorithm [18]. We take video clips of twelve subjects and a further three for testing. Since in a FN walk, there is the looming effect caused by the subject approaching the camera. This causes the movements to grow larger and show a definite trend in the data as will be seen in Figures 3 and 4. The data trend is immediately removed and normalized in the following way:

i) Use the coordinates of the *bm* marker as the origin of the markers.
ii) The length between the *tm/bm* markers are used to divide the distance between the *bm* marker and the other marker coordinates.

Thus every subject will have 12 time series associated with the *x* and *y* movements of the 6 markers attached to the body, for a FN walk. We have 12 subjects giving a total of 144 time series. In a FP walk, we have only 6 time series from 3 markers and 2 sequences, giving 12 time series. This is because analysis using FP data are well documented in the literature.

Of the gait datasets currently available, most are of the FP view taken at low resolution. Features for recognition include frequencies of motion from Fourier-based decomposition of the motion signal. As described by Lee et al. [19] these gait datasets are not suited for our use and there are few substantial video sequences of FN gait available. Thus we create and explore the use of a dataset that focuses on and exploits the advantages of the FN view. The experience gained in using this smaller dataset will serve to prepare for larger scale work. We have FP gait sequences which are used for confirmatory tests of linearity only.

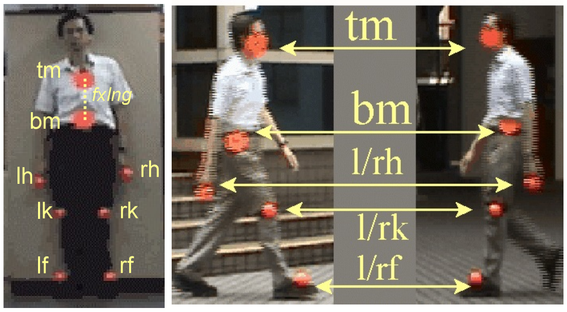


Figure 2. Marker positions: Left - FN view Right - FP view

III. Theoretical Considerations of Nonlinear Signal Analysis and Data Analysis Procedure

This section looks at the theory used to analyze the linearity of gait signals. The plot of the autocorrelation function (*ac.f*) gives a quick visual indicator of the nature of the signal. Of the several analytical methods available, there are those based on frequency domain approaches like that of the bispectrum or higher order moments which do not need parameters. In the time domain, Autoregressive (AR) and Moving Average (MA) models are popular. These methods model the human walk using a set of computed parameters.

A. Testing for nonlinearity using nonparametric methods

The nonparametric method of surrogate data introduced by Theiler et al. [20] uses a more general form of statistical hypothesis testing where we postulate the null hypothesis of linearity. Then *simulated* data are generated from the processes which are known to have linear and stationary properties. A discriminating statistic is computed on the simulated data and a *critical* value determined, based on various levels of significance for which the hypothesis holds.

Using the *experimental* data now, the discriminating statistic is again computed and compared with the critical values which now act as threshold values. We seek to reject the null hypothesis so that the alternative is true - i.e., the presence of nonlinearity. Surrogate data is one way to obtain *simulated* data conforming to the null hypothesis. The steps are:

i. A Fourier transform (FT) is applied to the data.
ii. In the transformed data, the phase is randomized.
iii. The data set is converted back to the time domain.

The surrogate data has linear properties and maintains the stationary properties of the original data, like the variance and autocorrelation. Various modifications on the basic algorithm refine on how well the stationary properties are kept. Thus the surrogate data are Gaussian, linear and stationary. Many sets of surrogate data can be generated by changing the random seed. Using surrogate data requires a suitable discriminating statistic to be determined from *both* experimental and surrogate data and a comparison made. Schreiber and Schmitz showed that a statistic based on nonlinear predictor errors (NPE) gives good results for detecting nonlinearity as compared to several others [21]. The null hypothesis of linearity postulates that the NPE computed from the original data lies within normal variation limits with the NPE obtained from sets of surrogate data [22]. Nonlinearity points the way to novel methods of analysis such as that used in chaos theory.

For the sake of completeness, we include a brief discussion about the use of prediction as a test of linearity - a fuller account may be found in [23]. Assuming a signal with a deterministic structure, its *predicted* values in the short term may be expressed as a *linear* weighted sum of its previous values in the time domain as described in (1). Of course, we will use values already in the time series to compute the prediction error. If the previous values are shuffled around as in the case of surrogate data, there will be a large variation in the short term predicted values of the surrogate data. However if we use a suitable nonlinear prediction method which does not depend on linear computations, the predicted values should not vary so much in the surrogate data. This approach uses nonlinear prediction in phase space as explained in Section III.B.

Of the many ways of characterizing nonlinear behaviour, we select the most widely used from two major categories. The first consists of examining its overall behaviour using phase space approaches. A widely used method invokes deterministic chaos theory. Another set of approaches is similar to linear analysis, but this time the signal is split into constituent nonlinear functions. For the sake of completeness, we describe an earlier work of ours using a measure of chaos, namely the Largest Lyapunov Exponent, or λ_1 to characterize gait [8].

B. Measuring Chaos with Lyapunov Exponents

To test for nonlinear chaotic behaviour, a *scalar* time series is subjected to dynamical analysis which assumes that the time series data x is generated by a vector valued process. The actual state vectors describing this process may never be known. But we can create a set of *phase space* vectors which are topographically equivalent, and can be considered to be a reconstruction of them. Takens' "method of delays" is an established method for doing this [24]. He also shows that if the dimension of the phase space vectors m is larger than the dimension of the *chaotic* attractor D, we can say that the phase vectors *embed* the state vectors and $m > 2D + 1$. Thus the reconstructed trajectory of X is made up of several phase space vectors as follows:

$$X = [X_1\ X_2 \dots\ X_m]^T$$

where X_i is the state of the system at sample i. Each row of X is a phase-space vector with a length of the embedding dimension m. That is, for each X_i,

$$X_i = [\ x_i\ x_{i+\tau} \dots\ x_{i+(m-1)\tau}\ ]$$

where τ is the time lag for a time series $x = \{x_1, x_2, \dots, x_N\}$ with N points. So X is an M by m matrix, and we have M the number of phase space vectors being $N - (m - 1)\tau$. The set of phase vectors describes a path or a trajectory in m dimensional space, and analyzing its behaviour gives a measure of chaos.

For parameter τ, the standard method is to take the time when the autocorrelation plot first goes to zero. But in Figure 7 we see that it never reaches zero until the end of the walk, so we use the time delayed mutual information measure as proposed by Fraser and Swinney [25]. For parameter m, we use the method of false nearest neighbours (FNN) proposed by Kennel et al. in [26].

In characterizing chaotic behaviour, the largest Lyapunov exponent λ_1 is the most useful and commonly used measure. If the system equations generating the data are known, it is quite straightforward to calculate it. λ_1 describes how quickly trajectories approach or come together, given different initial conditions. This comes directly from a definition of chaos. Then λ_1 is the mean exponential rate of divergence of two initially close trajectories from an initial time t_0 to t_i. The divergence d_j between the j^{th} set of points on the two trajectories is the Euclidean distance between them.

$$\lambda_1 = \frac{1}{t_i - t_0} \sum_{k=1}^{i} \log_2 \frac{d(t_k)}{d(t_{k-1})} \qquad (2)$$

One of the more recent methods to calculate λ_1 was formulated by Rosenstein [27] and independently, by Kantz [28]. This method is suitable for small and noisy data sets. Assume a fixed sampling time period Δt and that at t_i the sample number is i so that $t_i - t_0 = i\Delta t$. We substitute the subscripted time t_i by its index i. Taking logarithms on both sides of (2), we have:

$$\log_2 d_j(i) \ = \ \lambda_1 i \Delta t + \log_2 d_j(0)$$

The initial separation $\log_2 d_j(0)$ is constant, so we have a group of $j = 1$ to M (phase space vectors) approximately parallel lines for the sample number i. The main feature of this method is that we average the $\log_2 d_j(i)$ values for all j pairs of sample points at each sample j. Then

$$\langle \log_2 d_j(i) \rangle / \Delta t = \lambda_1 i + \langle \log_2 d_j(0) \rangle / \Delta t \tag{3}$$

where <·> is the average operator. We average further by fitting a line using Least Squares to the "average line" of (3) after which, λ_1 is the slope of the fitted lines. This will be shown in Section V.

C. The Hilbert Huang Transform

Recently, the EMD technique has been used for signal analysis and decomposition. Huang et al. pioneered its use in ocean wave studies [29]. It was motivated by the need to study nonlinearity and nonstationarity by obtaining the instantaneous frequency and amplitude of a signal as defined in (5). These allow us to see where the signal is changing, but the difficulty lies in the *scale* of the change, for example intermittent background noise in a larger say, audio signal. Traditional time-frequency signal processing methods like the Fourier Transform wavelet analysis do not provide a sharp distinction between the various harmonic components of the signal [30].

The HHT attempts to overcome this problem in a two-step process. Firstly, EMD decomposes a signal into a set of constituent functions, which are the IMFs at suitable scales of the signal. These functions are then subject to the Hilbert Transform which gives amplitude and phase information over the duration of the signal from which we obtain the instantaneous frequency and amplitude.

1) IMFs and the sifting process

Assuming the signal is oscillatory, IMFs have two special properties - firstly, the number of extrema and zero crossings must be equal or differ by one. Secondly, the envelope of a signal touching the local maxima and the envelope touching the local minima of the IMF has a local mean value of zero. The signal $x(t)$ is decomposed into its IMFs through the process of sifting. Rather than fitting a predefined mathematical procedure, this works with the signal data directly. For the first function IMF_1:

i. locate all the extrema of $x(t)$
ii. generate the envelope signals touching the maxima and minima $e_{max}(t)$ and $e_{min}(t)$ respectively
iii. obtain the mean signal $m(t) = (e_{max}(t) + e_{min}(t))/2$
iv. from the original and mean signal, obtain the residual signal $r(t) = x(t) - m(t)$
v. iterate steps i to iv by substituting $r(t)$ into $x(t)$ until a given criterion is met. The residual signal is IMF_1.

The next function IMF_2 is derived by using $x(t) - m(t)$ in place of $x(t)$ above. The whole process stops when a monotonic IMF is obtained. IMFs may or not have constant amplitude and frequency and can be used to reconstitute the original signal, or for further processing. In our case, the Hilbert Transform is applied to each IMF obtain the instantaneous frequency.

2) The Hilbert Transform

The Hilbert Transform computes the conjugate function $y(t)$ of any real valued function $x(t)$. By doing so, an analytic function $z(t) = x(t) + iy(t)$ is defined. In polar form:

$$\begin{gathered} z(t)=a(t)e^{i\theta(t)} \text{ where } a(t)=\sqrt{x(t)^2+y(t)^2} \\ \text{with } \theta(t)=\arctan\frac{y(t)}{x(t)} \text{ so that } \omega(t)=\frac{d\theta(t)}{dt} \end{gathered} \tag{4}$$

where $\omega(t)$ and $a(t)$ are the instantaneous frequency and amplitude, respectively at time t. From this, other measures like the mean instantaneous frequency (*MIF*) and the weighted mean instantaneous frequency (*WMIF*) can be derived for *each* IMF of the original signal. Then using quantities defined in (4)

$$MIF=\frac{1}{N}\sum_{t=1}^{N}\omega(t) \qquad WMIF=\frac{1}{N}\sum_{t=1}^{N}a(t)\omega(t) \tag{5}$$

for N samples, for a given IMF.

3) Analysis procedure

Since there is such a wide range of data, we perform a simple data reduction operation for ease of analysis. We use the simple average of the *WIMF*s of all the markers of a subject, looking for those which remain relatively constant for separate gait sessions. This is done to use as much idiosyncratic information as possible.

IV. Results From Prior Experiments

This section covers the waveforms obtained from tracking body parts and the results required for nonlinear analyses of signals from previous publications and have been reproduced here for the sake of completeness. The results concern the tests for linearity or the lack of it and the derivation of parameters required for proper embedding of data as explained in Section III.B.

As described earlier, the subjects in our dataset are designated by symbols such as *s01*, *s02*, *s04* and so on. Those having the suffix '*a*' are the second video sequence of the subject, as *s02a* is the second video from *s02*. The unnormalized and normalized plots for a FP walk are shown in Figures 3 and 5 respectively. In this figure, the x-axis motion would seem to swamp out that of the y-axis. This can be visualized for example, that the horizontal left to right motion of an arm swing is larger than that of the vertical motion.

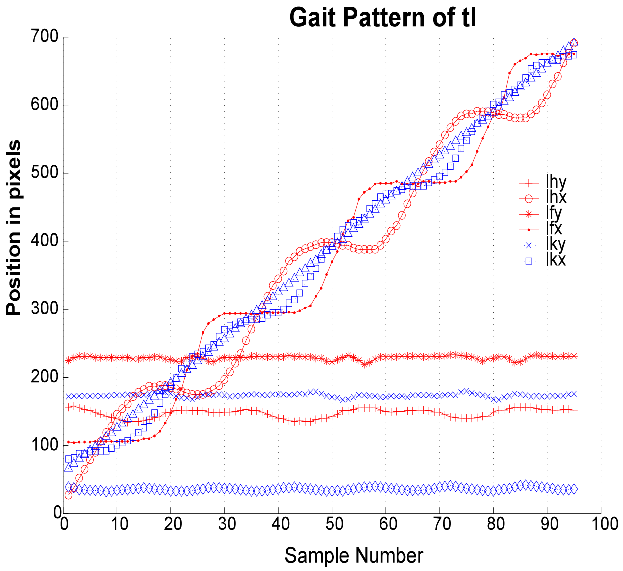


Figure 3. Unnormalized plot of a FP walk

In Figure 3 we note the motion of the body parts, in particular the x-axis coordinates which show an increase with a linear trend, reflecting the steady walking speed of the subject. By normalizing, we obtain the periodic waveform shown in Figure 5.

As a comparison, the corresponding plots for the FN walk are shown in Figures 4 and 6. In Figure 4 the coordinates increase with a nonlinear trend, a consequence of the physics of a thin lens. Here the dimensions of an object in the lens' focal plane varies inversely with the object distance from the lens [31]. However, the normalized plot in Figure 6 gives a semblance of a periodic waveform.

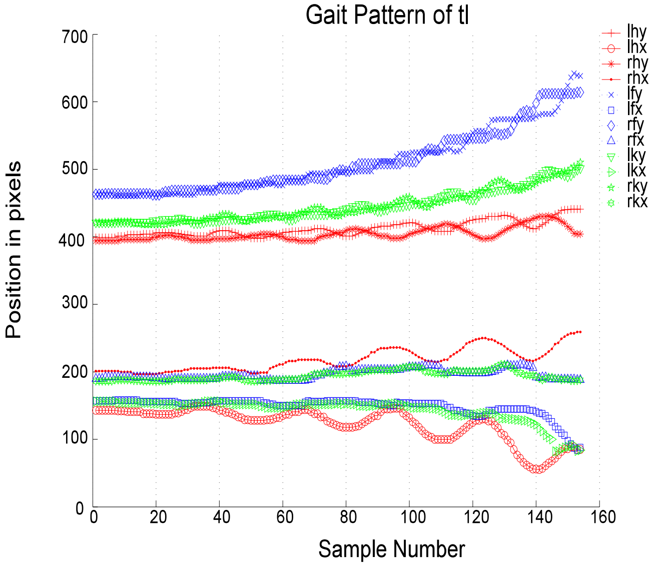


Figure 4. Plots of the markers for an unnormalized FN walk

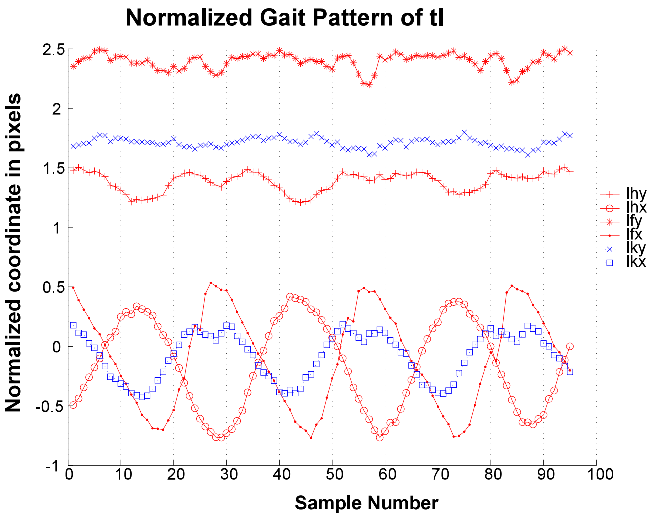


Figure 5. Plot of normalized FP walk

A. Linearity tests

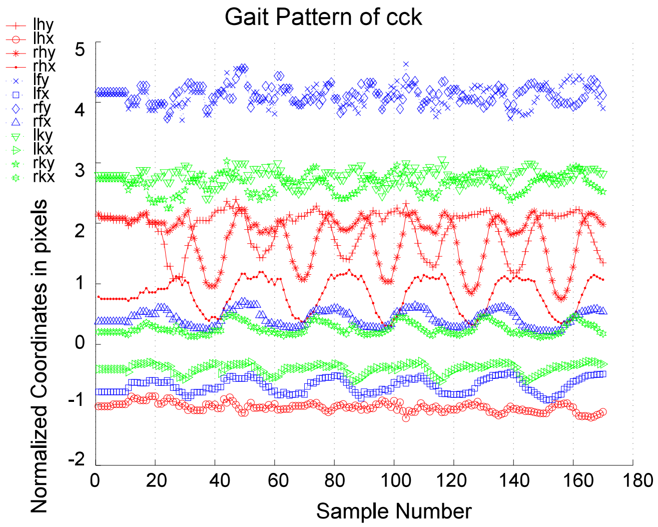


Figure 6. Plots of the markers for a normalized FN walk

In this section we show the results of the tests for signal linearity. The first is the autocorrelation plot for FP gait in Figure 9 which shows strong periodicity in movement, especially in the x-axis which due to its large amplitude swamps out - that is obscures - the "non-periodic" signal in the y-axis when considering the *total* movement of the hand. In contrast, the autocorrelation plot for the FN gait in Figure 7 does not show any periodicity in *any* of the twelve marker trajectories. This is an indicator of nonlinear dynamics or chaotic behaviour. However, it is interesting to note that the motion of a FN walk *silhouette* is periodic [32].

We now look at the results for nonparametric testing of nonlinear behaviour as discussed in Section III.A. The nonlinear prediction error is a discriminating statistic which gives a good test of linearity.

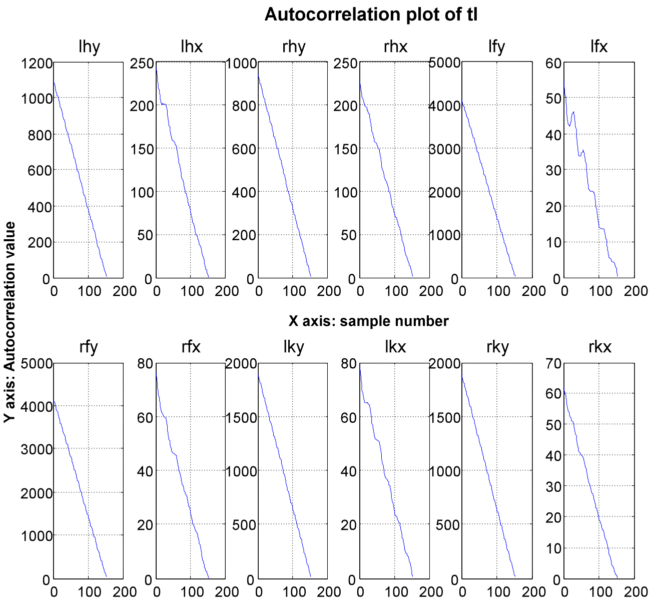


Figure 7. Autocorrelation plot - FN walk of 12 markers

It is generated from surrogate data and compared with that from the experimental data. An embedding dimension $m = 2$ was used, with a time delay $\tau = 5$. These values are determined experimentally in Section IV.B. A total of 19 surrogate data series were computed from the movement of one body part. For example in Table 1, *lhy* denotes the values of the movement of the left knee, y axis.

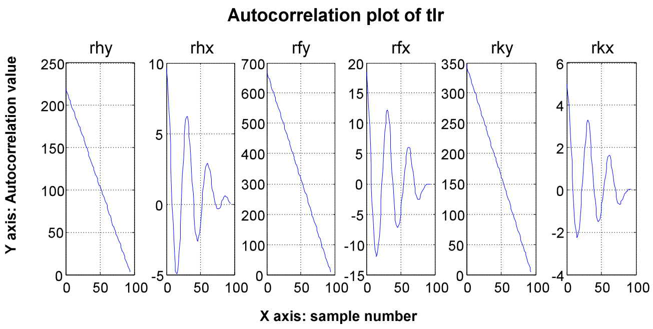


Figure 9. Autocorrelation plot of 6 body markers for FP walk

Here, sMean, sSTD are the mean and standard deviation of the values in all the 19 surrogate data series, *dMean* is the value for the actual data. We use the t-test to see if *dMean* lies within the variability of surrogate data described by *sMean* and sSTD. The probability column indicates the probability that *dMean* can be described by the null hypothesis H_0 being true. We see that for the first entry the data probably fits H_0, and more weakly for the third entry, but the rest reject H_0. Thus the null hypothesis can be rejected and the data can be considered nonlinear. In Figure 8 we show the plot of the *lhx* marker of a subject and two of its surrogates.

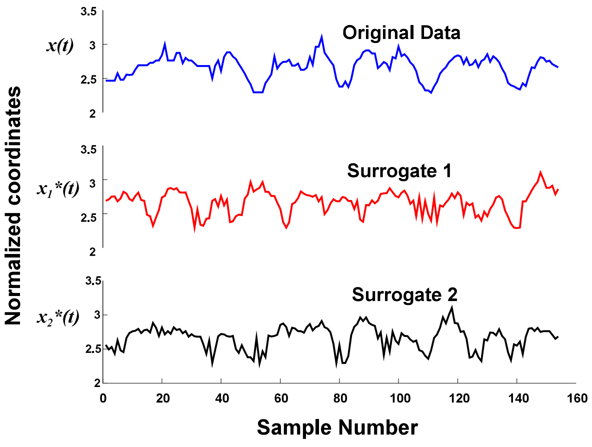


Figure 8. Segmenting the original data and two surrogates for computing statistics

TABLE 1
NONLINEAR PREDICTION ERROR FOR A TYPICAL FN WALK
T-TEST RESULT

	sMean	dMean	sSTD	probability	H_0
lhx	0.176	0.174	0.009	0.814	accept
lhy	0.182	0.119	0.016	0.00	reject
rhx	0.162	0.159	0.009	0.07	accept
rhy	0.145	0.122	0.008	0.00	reject
lfx	0.259	0.227	0.014	0.00	reject
lfy	0.134	0.090	0.011	0.00	reject
rfx	0.231	0.221	0.010	0.00	reject
rfy	0.146	0.109	0.008	0.00	reject
lkx	0.166	0.152	0.009	0.00	reject
lky	0.123	0.104	0.011	0.00	reject
rkx	0.172	0.158	0.010	0.00	reject
rky	0.118	0.090	0.009	0.00	reject

In Table 2, for the FP walk, we include the t-statistic instead of the standard deviation. This is because we see the *x-axis* values, those marked with an '*' in the last column, having high t-statistic values (indicating rejection of H_0) even though they seem strongly periodic. This phenomenon has been described by Stam et al. [33], and is actually an indication of the strongly periodic signals and thus an acceptance of H_0.

TABLE 2
NONLINEAR PREDICTION ERROR FOR FP WALK
T-TEST RESULT
* denotes special non rejection of H_0

	sMean	dMean	t-statistic	Probability	H_0
lhy	0.063	0.045	22.24	0.00	reject
lhx	0.184	0.047	19.97	0.00	accept*
lky	0.064	0.064	1.80	0.64	accept
lkx	0.268	0.037	29.91	0.00	accept*
lfy	0.059	0.064	-6.01	0.00	reject
lfx	0.158	0.087	12.84	0.00	accept*

Since we have evidence of the nonlinearity of FN gait, we have justification for using nonlinear measures on the data from FN gait.

B. *Measures of chaosity*

Recall in Section III.B, that in order to characterize chaos, a first step is to embed the data into vectors, which require the parameters m and τ, For τ and using the mutual information measure, we show a sample plot in Figure 10 for one person. The point at which the first minimum of the plot is taken to be the best value for τ which is 2 in this case, for all twelve marker trajectories. For m, we use the method of false nearest neighbours (FNN) as described in Section III.B. A typical plot is shown in Figure11. Taking the average of *all* the largest values where the FNN goes to zero, we find the nearest integer value to be six.

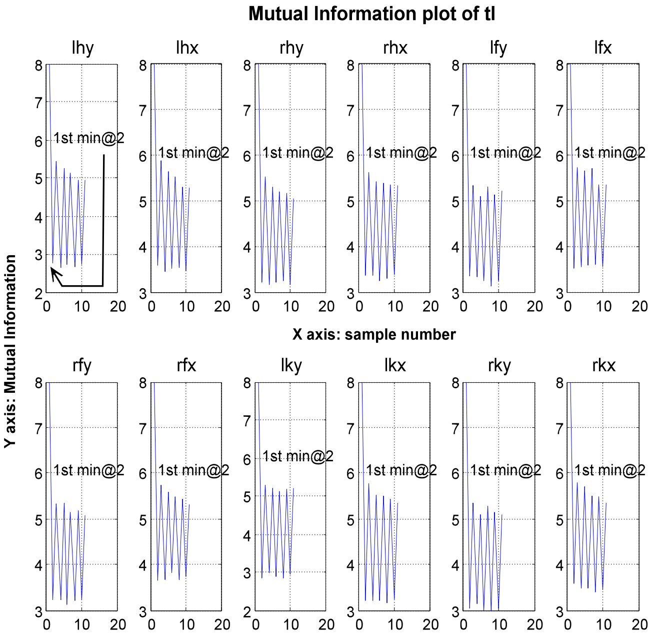


Figure 10. Mutual Information plots - markers of one person in FN walk. Position of first minimum shown in top left subplot.

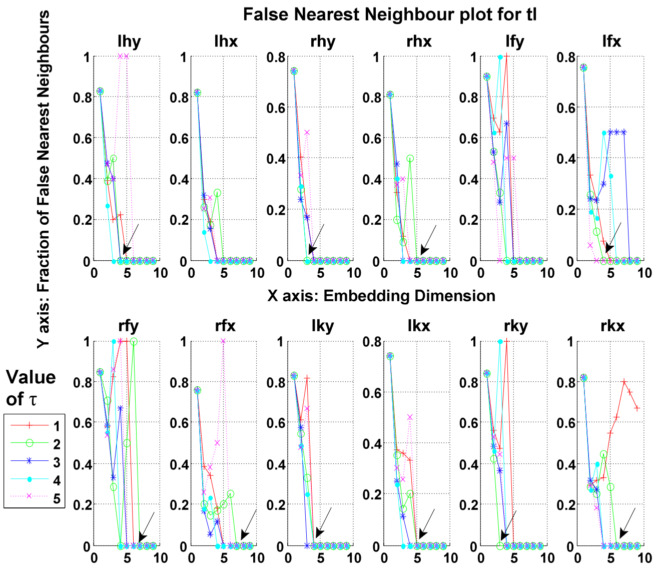


Figure 11. False Nearest Neighbour (FNN) plots for the markers of one person in a FN walk. Arrow marks point where fraction of FNN goes to zero for τ = 2.

V. RESULTS

In this section, we present the results of our experiments on characterizing gait using nonlinear measures of determinstic chaos and also quantities derived from the HHT.

A. *Characterizing gait using measures of deterministic chaos*

As we have also discussed in Section III.B, the slope of the line fitted to the trajectory will be λ_1. In Figure 12 we see a plot of λ_1 for the twelve marker trajectories of a person. We see that the data is mildly chaotic as λ_1 is positive. As a data reduction measure, we compute the average $\overline{\lambda}_1$ of all the λ_1 of the markers for a subject. An interesting observation in Table 3 is that subjects having similar $\overline{\lambda}_1$ are *s02*, *s03* and s10. We now employ a similar approach for the HHT.

TABLE 3 Values of λ_1 for 12 markers of 3 subjects for τ =2 and m = 5

τ2m5	s02	s02a	s03	s03a	s10	s10a
lhx	1.801	3.710	1.781	2.073	2.242	2.026
lhy	3.726	4.853	2.506	3.572	2.614	1.770
rhx	3.629	2.633	4.016	3.811	2.975	2.582
rhy	3.869	3.333	4.431	3.027	2.962	2.230
lfx	2.495	2.332	2.347	2.112	1.535	1.760
lfy	2.745	1.740	2.256	2.864	2.233	2.219
rfx	2.280	3.145	2.391	2.185	1.985	2.024
rfy	2.832	3.352	3.680	4.267	1.103	3.181
lkx	2.710	2.490	1.988	1.882	2.308	1.644
lxy	4.088	2.641	1.888	2.472	1.912	2.450
rkx	3.395	3.361	2.505	2.173	1.561	1.293
rky	2.877	3.361	3.168	2.538	1.605	2.453
avg	3.037	3.079	2.746	2.748	2.086	2.136
var	0.67	0.76	0.84	0.74	0.56	0.48

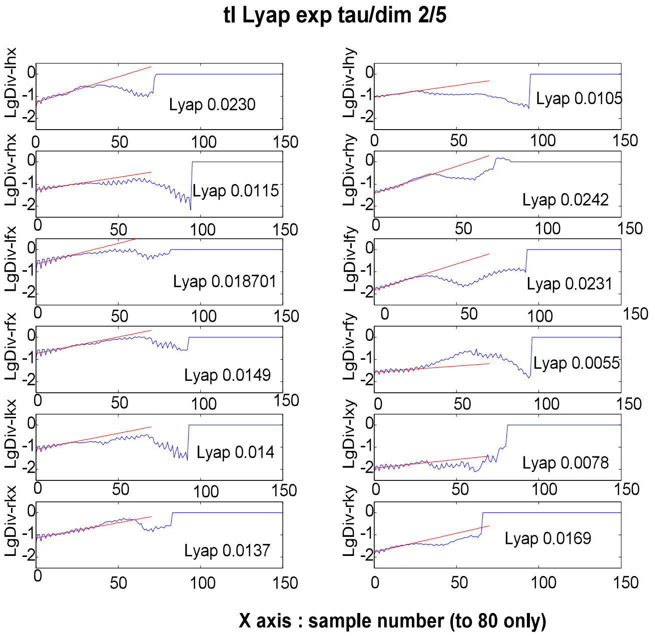


Figure 12. Computation of λ_1 of trajectories of a person's markers using Rosenstein's method as in (2). The y-axis are the log of the divergence and x-axis are the sample numbers. The slope of the average line gives λ_1.

B. Characterizing gait with HHT

The decomposition of the waveform of the *lhx* movement of a subject in Figure 8 (marked "original data") using EMD is shown in Figure 13. Here we show the plots for the IMF and *WMIF*. As we see from the top of the left column of Figure 13 the first few IMF's have a lot of high frequency content accounting for the fine movements of the marker. The frequency decreases with increasing IMFs. While the similarity to frequency decomposition methods like Fourier and wavelet analysis is there, note that the IMF waveforms do not have analytical expressions, hence the empirical nature of EMD. In the right column we see the normalized instantaneous frequency and amplitude at each sample point.

Through experimentation, we found that the third IMF gave the best results, leaving out the knee markers *lkx/y*, *rkx/y* and using *WMIF* instead of the *MIF*. In Table 4, we show the *WMIF* for only the 3 subjects with an extra video sequence. As in the case of chaosity we use as a feature the simple average of all the *WMIF*s of the markers of a subject, excluding the knee markers. This is indicated by the *avg wimf* row at the bottom row of Table 4.

C. Class separability

We apply statistical pattern recognition techniques to our data set even though it is small, to check the feasibility for when a suitable corpus of data is available. We assume that the subjects belong to a class and we examine the separability of the classes and if needed, to see if using other features can help. The values of a feature for a subject are assumed to be normally distributed, the prior probabilities of each class are the same, and we use the pooled variance as the variance of the data for all the subjects. Since there are only three subjects with a test video, we will use the variance from these groups. The Bhattacharyya distance B_{ij} between classes, defined as:

$$B_{ij}=(1/8)(\mu_i-\mu_j)^T((\Sigma_i+\Sigma_j)/2)^{-1}(\mu_i-\mu_j) \\ +(1/2)\ \ln(|(\Sigma_i+\Sigma_j)/2|/(|\Sigma_i|^{1/2}|\Sigma_j|^{1/2}) \quad (6)$$

where μ and Σ are the mean and variance of the classes is used extensively used for measurements of class separability.

TABLE 4 Weighted mean instantaneous frequency for IMF_3 of markers of 3 subjects

marker/subj	*s02*	*s02a*	*s03*	*s03a*	*s10*	*s10a*
lhx	4.16	3.42	3.36	3.5	5.45	3.44
lhy	3.32	4.06	3.52	3.5	2.78	2.55
rhx	3.56	3.91	3.19	3.45	3.26	5.33
rhy	3.86	4.05	4.00	3.35	3.39	3.79
lfx	5.72	4.94	3.78	4.07	6.95	5.19
lfy	3.62	3.21	2.98	1.72	3.2	3.46
rfx	3.19	5.36	3.97	4.48	4.08	6.42
rfy	3.36	3.11	3.36	3.04	3.32	2.26
avg wmif	3.85	4.01	3.52	3.39	4.05	4.06

Since we are using the pooled variance, the log term will be zero. This is more useful than showing the values of $\overline{\lambda}_1$ for *all* the subjects. The results of the calculation between pairs of classes are shown in Table 5 for the $\overline{\lambda}_1$ measure. This is reproduced from [6] for comparison and a more complete discussion. Since the table has are *symmetric* data, in the interests of clarity, we show only the upper diagonal values. We see that some classes are poorly separated with a B_{ij} value (rounded) that is less than or equal to 1. For example classes *s04*, *s06* and *s07* cannot be disambiguated, between *s08* and *s12, s02* and *s11* as well.

Similarly, in Table 6 for the *WIMF* classes that cannot be disambiguated are *s01, s02, s06, s08*, *s11* between *s10* and *s11,* and *s05* and *s12*. Making the assumption that the poorly separated classes are not separable, we fuse the $\overline{\lambda}_1$ measure mentioned earlier, so we can successfully classify members of our data set. Now we combine both features to see if we can achieve a better result. We combine both features using a logical AND. For example, if the *avg WMIF* (HHT-based feature) denotes the person being in the group *s01, s02, s06, s08*, *s11* and the $\overline{\lambda}_1$ (Largest Lyapunov Exponent based feature) being in the group *s04*, *s06* and *s07*, the person would be *s06*. To save space, we now show the *combined* confusion matrix in Table 7.

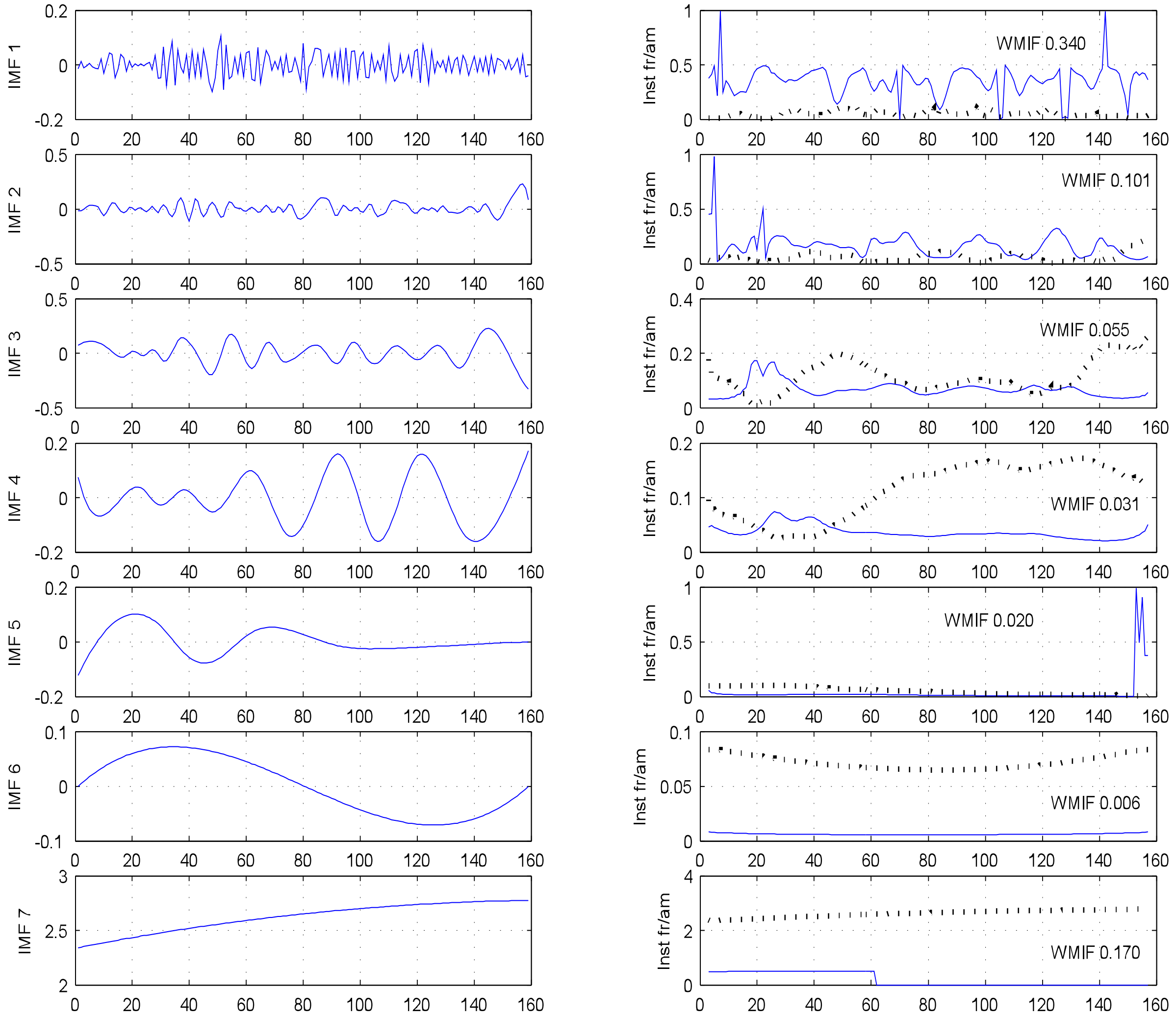


Figure 13. Plots of the (left) first seven IMFs of the *lhx* movement of a subject. On the right is the normalized Instantaneous Frequency (dotted line) and Instantaneous Amplitude (solid line) for the IMF. The Weighted Average IMF is shown as well. The x-axes all denote the sample number.

TABLE 5 Bhattacharyya distance betwen classes using $\overline{\lambda}_1$

	s01	*s02*	*s03*	*s04*	*s05*	*s06*	*s07*	*s08*	*s09*	*s10*	*s11*	*s12*
s01	0	85	11	25	11	22	27	3	31	78	75	5
s02		0	36	201	155	190	206	56	14	324	1	51
s03			0	68	43	62	71	3	6	146	29	2
s04				0	3	1	1	45	111	15	185	51
s05					0	2	4	2	2	2	141	29
s06						0	1	103	18	175	2	45
s07							0	48	115	14	190	53
s08								0	15	111	48	1
s09									0	206	10	12
s10										0	303	119
s11											0	43
s12												0

TABLE 6 Bhattacharyya distance betwen classes using HHT

	s01	*s02*	*s03*	*s04*	*s05*	*s06*	*s07*	*s08*	*s09*	*s10*	*s11*	*s12*
s01	0	0	4	56	24	1	6	0	7	2	0	20
s02	0	0	8	45	18	0	3	1	11	0	0	14
s03	0	0	0	92	50	10	21	3	0	12	6	44
s04	0	0	0	0	6	40	25	60	104	36	50	8
s05	0	0	0	0	0	15	6	28	59	12	21	0
s06	0	0	0	0	0	0	1	2	15	0	0	11
s07	0	0	0	0	0	0	0	7	26	1	4	4
s08	0	0	0	0	0	0	0	0	6	3	0	23
s09	0	0	0	0	0	0	0	0	0	17	9	52
s10	0	0	0	0	0	0	0	0	0	0	1	9
s11	0	0	0	0	0	0	0	0	0	0	0	17
s12	0	0	0	0	0	0	0	0	0	0	0	0

TABLE 7 Confusion matrix combining HHT and Largest Lyapunov Exponent as features.

PREDICTED/ACTUAL in %

	s01	*s02*	*s03*	*s04*	*s05*	*s06*	*s07*	*s08*	*s09*	*s10*	*s11*	*s12*
s01	100											
s02		100										
s03			100									
s04				100								
s05					100							
s06						100						
s07							100					
s08								100				
s09									100			
s10										100		
s11											100	
s12												100

A useful result is that we are able to separate all classes successfully using these two nonlinear measures.

VI. Conclusion and Future Work

We have used a camera as a sensor to derive the gait signals of a person in a multimedia video data stream. The objective is to see if the data can be used to identify a person. We have shown by using several types of analyses that signals derived from FN gait is nonlinear in nature. This is in contrast to current approaches which impose linear analyses of gait signals for convenience. This gives us a basis for using nonlinear measures of gait. In our experiments, the simple average of the nonlinear features by themselves are not able to discriminate completely between the classes of subjects. However, by combining them, we get a successful result. The novelty of our approach lies in the evaluation and use of both cyclostationarity and nonlinear measures of the gait signal. While our dataset is small, we note that the number of nonlinear features we can extract from the gait signals is potentially very large. A more exhaustive search for features and combinations of signals may yield useful results in terms of deriving new biometrics or improved recognition rates. So future work will need to test this out for current biometrics in a larger dataset and in a markerless environment. There is also much scope for investigating other types of signal analysis that is not based on linearity and stationarity assumptions. By doing so, we capitalize on the ubiquity of video cameras, from which we are able to obtain sensory data which can be used to augment security networks.

References

[1] M. W. Whittle, *Gait Analysis: an Introduction*, 4th ed., Philadelphia: Butterworth-Heinemann, 2007, pg 139.

[2] G. Verhoeven, “Did the digital (r)evolution change the concept of focal length?” *AARGNEWS*, vol. 34, pp. 30-35 March 2007.

[3] T. K. M Lee, M. Belkhatir, P.A. Lee and S. Sanei, “Fronto-normal gait incorporating accurate practical looming compensation,” *Proc. 19th Int’l Conf. Pattern Recognition (ICPR 2008)*, 2008.

[4] T. Lee, S. Ranganath and S. Sanei, “Patterns in paces: a survey of current approaches in gait recognition,” *Proc. Asian Biometric Workshop (ABW 2003)*, pp 421-425, Singapore, Nov. 2003.

[5] A. Veeraraghavan, A. K. Roy-Chowdhury, and Rama Chellapa, “Matching shape sequences in video with applications in human movement analysis,” *IEEE Transactions on PAMI*, vol.27, no.12, pp.1896-1909, Dec 2005.

[6] T. Lee, S. Ranganath, and S. Sanei, “Fusion of chaotic measure into a new hybrid face-gait system for human identification,” *Proc. ICPR*, Hong Kong, August, 2006.

[7] T. K. M. Lee, K. F. Loe, P. A. Lee and S. Sanei, “A comparison of the basic temporal features of fronto-normal and fronto-parallel gait,” *DSP 2007*, Cardiff, UK, July 1-4, 2007.

[8] T. K. M. Lee, S. Sanei and B. Clarke, “Fusion of nonlinear measures in fronto-normal gait recognition,” *International Multi-conference on Computing in the Global Information Technology*, Valencia, Spain, 2010.

[9] T. K. M Lee, M. Belkhatir, P. A. Lee and S. Sanei, “Nonlinear characterisation of fronto-normal gait for biometric use,” *Proc. Pacific-Rim Conf. On Multimedia (PCM 2008)*, 2008.

[10] H. Tong, “*Non-linear Time Series*” 1st ed, Oxford Science Publications, 1993.

[11] G. Johansson, “Visual perception of biological motion and a model for its analysis,” *Perception & Psychophysics*, vol.14, no.2, pp.201-211, 1973.

[12] J. E. Cutting and L. T. Kozlowski, “Recognizing friends by their walk: Gait perception without familiarity cues,” *Bull of the Psychonomic Society*, vol.9, no.5, pp.353-356, 1977.

[13] N. F. Troje, “Decomposing biological motion: A framework for analysis and synthesis of human gait patterns,” *Journal of Vision*, vol.2, pp.371-387, 2002.

[14] M.S. Nixon, T. Tan and R. Chellappa , “*Human Identification Based on Gait*”, 1st ed, Springer, 2005.

[15] R. K. Ibrahim, E. Ambikairajah, B.G. Celler and N.H. Lovell, “Gait pattern classification using compact features extracted from Intrinsic Mode Functions,” *30th Ann. Conf. IEEE Engineering in Medicine and Biology Society*, 2008.

[16] N. Wang, E. Ambikairajah, B. G. Celler and N. H. Lovell, “Accelerometry based classification of gait patterns using Empirical Mode Decomposition,” *Proc. 2008 IEEE Int’l Conf. Signals, Acoustics and Speech Processing (ICASSP)*, 2008.

[17] P. Kuchi and S. Sethuraman, “Gait recognition using Empirical Mode Decomposition,” *Int’l Conf. Advances In Pattern Recognition*, 2003.

[18] G. R. Bradski, “Computer video face tracking for use in a perceptual user interface,” *Intel Technology Journal*, vol.Q2, 1998.

[19] T. K. M. Lee, M. Belkhatir and S. Sanei, “Techniques, issues and perspectives for gait-based recognition” (in review) by Pattern Recognition / Elsevier.

[20] J. Theiler, B. Galdrikian, A. Longtin, S. Eubank and J.D. Farmer, “Using surrogate data to detect nonlinearity in time series,” *Proc. Nonlinear Modeling and Forecasting, Santa Fe Institute Studies in the Sciences of Complexity, Vol. XII* , Santa Fe, 1992.

[21] T. Schreiber and A. Schmitz, “Discrimination power of measures for nonlinearity in a time series,” *Phys. Rev. E*, vol.55, no.5, pp.5443 - 5447, May 1997.

[22] D. Kaplan, “Nonlinearity and Nonstationarity: The use of surrogate data in interpreting fluctuations in heart rate,” *Frontiers of Blood Pressure and Heart Rate Analysis*, M. Di Rienzo, et al., IOS Press, 1997.

[23] J. Theiler and D. Prichard, “Constrained-realization Monte-Carlo method for hypothesis testing,” *Physica D*, vol.94, pp.221-235, 1996.

[24] F. Takens, "Detecting strange attractors in turbulence," *Lecture notes in Mathematics*, vol. 898, pp. 361-381, Springer-Verlag, 1981.

[25] A. M. Fraser and H.L. Swinney, "Independent coordinates for strange attractors from mutual information," *Phys. Rev. A*, 33, pp. 1134-1140, 1986.

[26] K. B. Kennel, R. Brown and H. D. I. Abarbanel, "Determining embedding dimension for phase-space reconstruction using a geometrical construction," *Phys. Rev. A*, vol. 45, no. 3403, 1992.

[27] M.T. Rosenstein, J. J. Collins and C. J. De Luca, "A practical method for calculating largest Lyapunov exponents from small data sets," *Physica D*, vol. 65, pp. 117, 1993.

[28] H. Kantz, "A robust method to estimate the maximal Lyapunov exponent of a time series," *Physics. Letters A*, vol. 185, pp. 77, 1994.

[29] N. E. Huang, Z. Shen, S. R. Long, M. C. Wu, H. H. Shih, Q. Zheng, N. Yen, C. C. Tung and H. H. Liu, "The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis," *Proc. R. Soc. Lond. A*, 454, pp. 903-995, 1998.

[30] N. E. Huang, Z. Shen, and S. R. Long, "A new view of nonlinear water waves: the Hilbert spectrum" *Ann. Rev. Fluid Mechanics*, pp. 415-457, 1999.

[31] E. Hecht, "Optics" , 4th edition, Addison Wesley, 2001.

[32] L. Wang, T. Tan, H. Ning and W. Hu, "Silhouette analysis-based gait recognition for human identification," *IEEE Transactions on PAMI*, vol. 25, no. 12, pp. 1505-1518, Dec 2003.

[33] C. J. Stam, J. P. M. Pijn and W. S. Pritchard, Reliable detection of nonlinearity in experimental time series with strong periodic components, *Physica D*, vol.112, no.3-4, pp.361-380, February 1998.

www.iariajournals.org

International Journal On Advances in Intelligent Systems
ICAS, ACHI, ICCGI, UBICOMM, ADVCOMP, CENTRIC, GEOProcessing, SEMAPRO, BIOSYSCOM, BIOINFO, BIOTECHNO, FUTURE COMPUTING, SERVICE COMPUTATION, COGNITIVE, ADAPTIVE, CONTENT, PATTERNS, CLOUD COMPUTING, COMPUTATION TOOLS
issn: 1942-2679

International Journal On Advances in Internet Technology
ICDS, ICIW, CTRQ, UBICOMM, ICSNC, AFIN, INTERNET, AP2PS, EMERGING
issn: 1942-2652

International Journal On Advances in Life Sciences
eTELEMED, eKNOW, eL&mL, BIODIV, BIOENVIRONMENT, BIOGREEN, BIOSYSCOM, BIOINFO, BIOTECHNO
issn: 1942-2660

International Journal On Advances in Networks and Services
ICN, ICNS, ICIW, ICWMC, SENSORCOMM, MESH, CENTRIC, MMEDIA, SERVICE COMPUTATION
issn: 1942-2644

International Journal On Advances in Security
ICQNM, SECURWARE, MESH, DEPEND, INTERNET, CYBERLAWS
issn: 1942-2636

International Journal On Advances in Software
ICSEA, ICCGI, ADVCOMP, GEOProcessing, DBKDA, INTENSIVE, VALID, SIMUL, FUTURE COMPUTING, SERVICE COMPUTATION, COGNITIVE, ADAPTIVE, CONTENT, PATTERNS, CLOUD COMPUTING, COMPUTATION TOOLS
issn: 1942-2628

International Journal On Advances in Systems and Measurements
ICQNM, ICONS, ICIMP, SENSORCOMM, CENICS, VALID, SIMUL
issn: 1942-261x

International Journal On Advances in Telecommunications
AICT, ICDT, ICWMC, ICSNC, CTRQ, SPACOMM, MMEDIA
issn: 1942-2601