International Journal on Advances in Networks and Services

2011 vol. 4 nr. 3&4

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 3 & 4, 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

- Masashi Sugano, Osaka Prefecture University, Japan
- Athanasios Vasilakos, University of Western Macedonia, Greece
- You-Chiun Wang, National Chiao-Tung University, Taiwan
- Hongyi Wu, University of Louisiana at Lafayette, USA
- Dongfang Yang, National Research Council Canada – London, Canada

Underwater Technologies

- Miguel Ardid Ramirez, Polytechnic University of Valencia, Spain
- Fernando Boronat, Integrated Management Coastal Research Institute, Spain
- Mari Carmen Domingo, Technical University of Catalonia - Barcelona, Spain
- Jens Martin Hovem, Norwegian University of Science and Technology, Norway

Energy Optimization

- Huei-Wen Ferng, National Taiwan University of Science and Technology - Taipei, Taiwan
- Qilian Liang, University of Texas at Arlington, USA
- Weifa Liang, Australian National University-Canberra, Australia
- Min Song, Old Dominion University, USA

Mesh Networks

- Habib M. Ammari, Hofstra University, USA
- Stefano Avallone, University of Napoli, Italy
- Mathilde Benveniste, Wireless Systems Research/En-aerion, USA
- Andreas J Kassler, Karlstad University, Sweden
- Ilker Korkmaz, Izmir University of Economics, Turkey //editor assistant//

Centric Technologies

- Kong Cheng, Telcordia Research, USA
- Vitaly Klyuev, University of Aizu, Japan
- Arun Kumar, IBM, India
- Juong-Sik Lee, Nokia Research Center, USA
- Josef Noll, ConnectedLife@UNIK / UiO- Kjeller, Norway
- Willy Picard, The Poznan University of Economics, Poland
- Roman Y. Shtykh, Waseda University, Japan
- Weilian Su, Naval Postgraduate School - Monterey, USA

Multimedia

- Laszlo Boszormenyi, Klagenfurt University, Austria
- Dumitru Dan Burdescu, University of Craiova, Romania
- Noel Crespi, Institut TELECOM SudParis-Evry, France
- Mislav Grgic, University of Zagreb, Croatia
- Hermann Hellwagner, Klagenfurt University, Austria
- Polychronis Koutsakis, McMaster University, Canada

- 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 3 & 4, 2011

CONTENTS

Packet Scheduling Architecture with Service Specific Queue Sorting and Adaptive Time Domain Scheduling Algorithms for LTE-Advanced Networks

Rehana. Kausar, Yue. Chen, Kok. Keong. Chai, John Schormans
School of Electronic Engineering and Computer Science
Queen Mary University of London
London, UK
rehana.kausar,yue.chen,michael.chai@elec.qmul.ac.uk

Abstract— **In this paper, a cross layer design packet scheduling architecture is proposed for Long Term Evolution-Advanced downlink transmission, to guarantee the support of quality of service requirements in a mixed traffic environment. The proposed architecture uses service specific queue sorting algorithms for different traffic types and an adaptive time domain scheduling algorithm to adaptively allocate available resources to real time and non real time traffic. Multiuser diversity is exploited both in the time domain and frequency domain by jointly considering the channel state information and queue state information. The aim is to improve the support of QoS guarantees to real time voice and non real time streaming video traffic and to maintain a good trade-off between system throughput and user fairness by optimizing the use of available radio resources. Results show that proposed packet scheduling architecture reduces delay, delay viability and packet drop rate of real time traffic while satisfying minimum throughput requirements of non real time traffic and it maintains the system throughput and fairness among users at good level.**

Keywords-LTE-A; Packet Scheduling (PS); OFDMA; Quality of Service (QoS); mixed traffic.

I. INTRODUCTION

Long Term Evolution Advanced (LTE-A) is an all-IP (Internet Protocol) based future wireless communication network, which is aiming to support a wide variety of applications and services with different Quality of Service (QoS) requirements. It is targeting at superior performance in terms of spectral efficiency, fairness, QoS support and service satisfaction as compared to the existing Third Generation Partnership Project (3GPP) wireless networks.

To achieve the goal, Radio Resource Management (RRM) plays a vital role. Packet Scheduling (PS) being one of the core functionalities in RRM is very crucial to optimise the network performance and it has been under extensive research in recent years. Different PS algorithms have been deployed aiming at utilising the scarce radio resource efficiently. A QoS aware Packet Scheduling Architecture (PSA) is presented in [1], which takes into account different prioritizing stages such as QoS aware queue sorting and adaptive Time Domain (TD) scheduler with built-in congestion control to the existing conventional QoS aware PS algorithms. Delay dependent queue sorting algorithm for Real Time (RT) traffic reduces average delay of RT traffic and built-in congestion control policies reduce Packet Drop Rate (PDR) by adaptively allocating radio resources to RT and Non Real Time (NRT) traffic types based on QoS feedback of RT traffic. And by exploiting multiuser diversity in the TD and Frequency Domain (FD), system overall throughout is improved. By prioritising users with longer delays in RT and NRT streaming video traffic and using conventional Proportional Fairness (PF) algorithm to sort users in RT, NRT and Best Effort (BE) queues respectively, the proposed PSA in [1] maintains a good trade-off between system throughput and user-fairness. However, there is still need of further work on PSA [1] in order to meet the requirements of QoS for RT traffic and throughput requirements of NRT traffic. Service specific queue sorting algorithms are needed for each queue to guarantee the QoS support. In addition, the fix built-in congestion control policies used in [1] need to be replaced with an adaptive scheme to make the PSA capable to adapt to the network conditions, traffic patterns, system load and the QoS requirements of different traffic types.

Thus, the functionalities of queue sorting and adaptive TD scheduler are enhanced by extended research on these algorithms. New queue sorting algorithms for RT and NRT queues have been proposed to further improve the support of the provision of QoS guarantees for both RT and NRT traffics [2]. The results show that the queue sorting algorithms have reduced average delay, delay viability and PDR of RT traffic while satisfying the minimum throughput requirements of NRT streaming video traffic at the cost of minor delays in the BE traffic. It also shows that system overall throughput and user fairness are maintained at good level. To emphasise the significance of new queue sorting algorithms, the adaptive TD scheduler with built-in policies as in [1] was not used instead the users were picked from the queues one-by-one from each queue starting from the top most queue by simple fair scheduling method.

In [1], the λ denotes the proportion of available Physical Resource Blocks (PRBs) assigned to RT users and $(1-\lambda)C$ to NRT users where C is the total number of PRBs available. The initial value of λ is decided based on the trade-off between the average delay of RT and NRT traffic as shown in Fig. 1. Then the value of λ is adaptively adjusted according to the PDR of RT traffic using built-in congestion control policies. In [1] however, only the average delay of RT and NRT traffic is considered to set an initial value of λ which is not very realistic as other performance metrics such as throughput of NRT traffic, system throughput and fairness among users should also be taken into account while setting this value. An extensive research has been done to adjust the initial value of λ so that a stability region can be found which takes into account various performance measures such as average delay of RT and NRT traffic, minimum throughput of NRT traffic and overall system throughput and fairness among users instead of only considering trade-off between RT and NRT average delay. After setting the initio value of λ, a new adaptive TD scheduling algorithm is used to make adaptive TD scheduler capable of controlling PDR at all traffic patterns instead of using a fix traffic pattern with only a number of built-in policies as in [1]. The results in [1] only consider a traffic pattern in which RT and NRT users are equal which is should be analysed by considering variable number of RT and NRT users as the number of RT and NRT users may vary with time. That is why the behaviour of the proposed PSA is analysed under different traffic patterns with varying number of RT and NRT users.

In this paper, the proposed PSA with new queue sorting algorithms [2] and novel adaptive TD scheduler algorithm is presented to enhance PS performance both at service level and network level. At the service level, the QoS of RT and NRT streaming video traffic are significantly improved as compared to the existing PS algorithms. At the system level, overall system throughput performance and fairness among all users are improved in the new PSA. As described above, this work is based on [1] that was presented in UBICOMM 2010.

The remainder of this paper is organized as follows. In Section II, the related work on PS algorithms is discussed. System model is presented in Section III and the proposed PSA with new queue sorting and adaptive TD algorithm is described in Section IV. In Section V, the proposed packet scheduling algorithm and performance metrics to analyse the proposed packet scheduling algorithm, are presented. The results and discussion section (Section VI) presents analytical results from different perspectives to show the performance of the proposed PSA; the first part of Section 5 presents a set of results to compare the performance of PSA with existing QoS aware PS algorithm and the second part evaluates the performance of PSA under different traffic patterns. Finally, conclusion and future work are presented in Section VII.

II. RELATED WORK

The classic packet scheduling algorithms exploiting multiuser diversity are the MAX C/I and Proportional Fairness (PF) algorithms. MAX C/I algorithm allocates a physical resource block (PRB) to a user with the highest channel gain on that PRB, and can maximize the system throughput [1] [3-4]. PF algorithm takes fairness among users into consideration and allocates resources to users based on the ratio of their instantaneous throughput and its acquired time averaged throughput [1] [5]. However, these algorithms aim only at improving resource utilization based on channel conditions of users; QoS requirements, for example delay requirements of real time (RT) traffic or minimum throughput requirements of non-real time (NRT) traffic, are not considered at all. In the next generation of mobile communication networks, apart from system throughput and user fairness, the crucial point is to fulfil users' QoS requirements in a multi-service, multi-user mixed traffic environment. This is because different traffic types are competing for radio resources to fulfil their QoS requirements. To allocate radio resources efficiently and intelligently in such complex environments is challenging. Various methods have been proposed aiming to use radio resources efficiently to fulfil QoS requirement of different traffic types [6-8].

A low complexity QoS aware PF multicarrier algorithm is presented for OFDM system in [9]. The objective is to achieve proportional fairness in the system while improving QoS performance. A greedy method based multi carrier PF criterion is proposed with the consideration that traditional single carrier PF is not suitable for OFDM systems. A subcarrier reassignment procedure is used to further improve QoS performance. This paper proposes PS algorithm specifically for the multimedia traffic and improves QoS, throughput and fairness in the system. However, there is a need to analyze the behaviour of the proposed algorithm when the system has to deal with different traffic types such as interactive, background traffic, etc. In [6], a service classification scheme is used which classifies mixed traffic into different service specific queues and grants different scheduling priorities to them. QoS of RT traffic is improved at the cost of system spectral efficiency, when the RT queue is granted the highest priority. And fairness is significantly improved when fair scheduling is used in the TD to pick users from the queues instead of strictly prioritizing RT traffic queue. Fair scheduling picks users one-by-one from each queue and strict priority empties queues one after other giving priority to RT queue. Conventional PF and MAX C/I are used to prioritise users in the queues. The QoS of RT and NRT traffic can be improved by using service specific queue sorting algorithms to prioritise users. In [10], an urgency factor is used to boost the priority of a particular traffic type. When any packet from a queue is about to exceed its upper bound of delay requirement, its priority is increased by adding an urgency factor. Although most of the packets are sent when they are nearly ready to expire, a lower packet loss rate is achieved thus improving the performance of system by guaranteeing QoS requirements to different traffic types.

In mix traffic scenarios, queue state information (QSI) becomes very important in addition to channel state information (CSI) [11-12]. It can make scheduling decision even more efficient; especially in QoS aware scheduling algorithms it is very crucial. Typically this implies to minimize the amount of resources needed per user and thus allows for as many users as possible in the system, while still satisfying whatever quality of service requirements that may exist [13]. A time domain multiplexing (TDM) system based Modified Largest Waited Delay First (M-LWDF) is presented in [11] which takes into account both QSI and CSI. This algorithm serves a user with the maximum product of Head of Line (HOL) packet delay, channel condition and an arbitrary positive constant. This constant is used to control packet delay distribution for different users.

It updates the queue state after each TTI rather than updating after each sub carrier allocation. M-LWDF significantly improves the support of QoS guarantees to the RT and NRT traffic for TDM systems. In [14], an exponential (EXP) rule is proposed for scheduling multiple flows that share a time-varying channel. The EXP rule is applied in M-LWDF as one of the parameters that equalizes the delays of different RT packets to reduce the PDR of RT traffic due to time-out. M-LWDF algorithm is applied in a frequency domain multiplexing (FDM) system in [15] to optimize sub-carrier allocation in Orthogonal Frequency Division Multiple Access (OFDMA) based networks. It shows improved performance in terms of QoS but like M-LWDF updates the queues state each TTI rather than after each sub-carrier allocation. In [16], M-LWDF for OFDMA systems is modified by updating the queue status after every sub-carrier allocation. It takes into account RT and NRT traffic types and provides better QoS for both services. The results show that the support of provision of QoS guarantees in terms of delay and PDR for RT and minimum throughput for NRT traffic is improved. However this idea can be extended to more effective scheduling framework by adding more traffic types and making resource allocation more adaptive based on the QoS. In [17] an adaptive algorithm with connection admission control (CAC) design is proposed. Due to large number of users and limited PRBs, CAC restricts the ongoing connections to provide required QoS and makes decisions whether to reject or accept new connections. It improves the QoS of RT traffic by prioritizing RT users and delaying users of other traffic types. In [18], a prioritizing function is used for packet data scheduling in OFDMA systems to satisfy QoS requirements of RT and NRT traffics. Priority is associated to different traffic types by setting different values of the prioritizing function. This algorithm allocates resources in a static way by setting the value of priority function for different traffic types and cannot cope with the highly dynamic variation of wireless channel conditions. In [19], a server allocation scheme to parallel queues with randomly varying connectivity is presented. The allocation decision is based on the connectivity and on the lengths of the connected queues only. The main aim of the work presented in [19] is to stabilize different queues. However this allocation policy can minimize the delay and maximize throughput for the special case of symmetric queues i.e., queues with equal arrival, service, and connectivity. However the work proposed by the author aims at considering system level and service level PS performance jointly. That is why various parameters are considered instead of only taking into account the stability of user queues, as in [19]. It takes scheduling decisions based on channel conditions to increase system spectral efficiency, average PDR to reduce PDR and delay viability and queue length to reduce packet delay and make the user queues stable.

As described in [11-18] and certain of the references therein, the PS algorithms improve scheduling performance in different domains separately such as system throughput, user fairness, QoS of RT and NRT traffic types. The Combined consideration of service level (QoS) and system level performance (system throughput and user fairness) improvement has got very little or no attention despite the fact that it is very crucial. Scheduling performance in different domains needs to be united in an efficient PS architecture so that the system can be made cost effective and radio resources may be utilised at the best. PS performance in different areas can be improved jointly by an intelligent PSA which is capable to make scheduling decisions adaptive to the environment and to the achieved performance in terms of QoS of different traffic types. The detailed traffic types can be considered in PSA to make the PS algorithms more realistic.

III. SYSTEM MODEL

An OFDMA system is considered in which minimum allocation unit is one Physical Resource Block (PRB) containing 12 sub-carriers in each Transmission Time Interval (TTI) of 1ms duration. There are K mobile users and M PRBs. The downlink channel is a fading channel within each scheduling drop. The received symbol $Y_{k,m}(t)$ at the mobile user k on sub channel m is the sum of the additive white Gaussian noise (AWGN) and the product of actual data and channel gain, as given in (5) [10-11].

$$Y_{k,m}(t) = H_{k,m}(t)X_{k,m}(t) + Z_{k,m}(t) \tag{1}$$

where, $Y_{k,m}(t)$is data symbol from eNodeB to user k at sub channel m, $X_{k,m}(t)$ is the input, $\left[H_{k,m}(t)\right]^2$ is the complex channel gain of sub channel m for user k, and $Z_{k,m}(t)$ is the complex White Gaussian Noise [11]. It is assumed, as in [11] [14-17], that the power allocation is uniform, $P_m(t) = P/M$ on all sub channels where, P is the total transmit power of eNodeB, $P_m(t)$is the power allocated at channel m and M is total number of sub channels. At the start of each scheduling drop, the channel state information (CSI) $H_{k,m}(t)$ is known by the eNodeB.

The achievable throughput of a user k on sub channel m can be calculated by (6) as used in [11] and [12].

$$C_{k,m}(t) = Blog_2\left[1 + \frac{|H_{k,m}(t)|^2}{\sigma^2 \Gamma} P_m(t)\right] \tag{2}$$

where, $B$is the bandwidth of each PRB, σ^2 is the noise power density i.e., noise power per unit bandwidth and Γ is a constant signal-to-noise ratio (SNR) gap and has a simple relationship with the required Bit Error Rate (BER).

$$\Gamma = \frac{-ln(5BER)}{1.5} \tag{3}$$

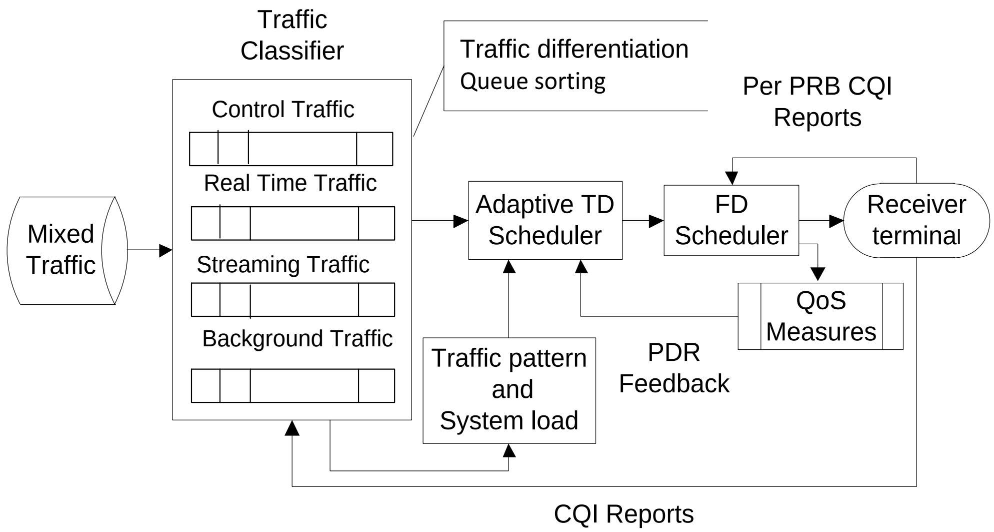


Figure 1. The cross layer packet scheduling architecture

IV. PACKET SCHEDULING ARCHITECTURE (PSA)

A schematic diagram of proposed PSA is shown in Fig. 1. It consists of a traffic classifier, adaptive TD scheduler and FD scheduler. Mixed traffic is classified into service specific queues at classifier stage. Users in these queues are prioritized according to QoS requirements. Adaptive TD scheduler adaptively allocates available radio resources to RT and NRT traffic types based on traffic pattern and system load information from traffic classifier and PDR information from QoS measure unit. QoS measure unit calculates PDR of RT traffic and minimum throughput of NRT traffic in each TTI to analyze the support of QoS provision to RT and NRT streaming video traffic. FD scheduler actually maps these resources to the selected users.

The detailed description of functionality, algorithms and policies of each proposed PSA stage are described as below.

A. *Classifier*

The need for traffic differentiation arises when there is a question to deal with mixed traffic demanding different QoS guarantees. In such an environment, it becomes very important to classify traffic into different service queues to enable queue specific prioritizing schemes effectively. Service differentiation is the first step towards optimising the utilization of available radio resources where the available radio resources are allocated according to the well-defined demands of traffic types [1].

In the proposed PSA mixed traffic is classified into four queues; Control, Real Time, Non Real Time and background traffic queue, as in [1]. These queues are represented by control, RT, NRT and BE queue hereafter. The queues at the traffic classifier stage are prioritized in the sequence as discussed above. These classes cover most of the common traffic types such as control information, low latency RT conversational, high throughput NRT streaming video and low priority background data. Control information is the signaling information exchanged between the User Equipment (UE) and eNodeB and it is separated from other data queues and is served before any other data queues. The control queue is always allocated enough radio resources to transmit signaling information to users. Background traffic represents the best effort (BE) class of traffic and does not have any QoS requirements. The service specific queue sorting algorithms used to prioritise users in these queues are as follows.

Control queue

In the proposed classifier, the control information is equally important information between users and therefore it is transmitted in Round Robin (RR) manner for all scheduled users.

RT queue

In RT queue, the delay requirement for each RT user is defined as $d_k < DB_k$ where d_k is the delay of user k, DB_{RT} the delay budget which is the upper delay bound of RT traffic. A delay dependent priority metric is used to sort users in the RT queue. The priority metric is shown in (4). It is the product of normalised waiting time of each RT user and its channel state information, and the product is added with the square of the user's queue length. In this priority metric, users with longer waiting time (normalised by DB), good channel conditions and longer queues, are prioritised in the front of the queue. By prioritising users with longer delays (normalised with DB), the priority metric reduces average delay of RT traffic significantly. In addition users are given equal opportunity to be scheduled thus improving fairness among users. By giving priority to users with longer queues, this priority metric reduces PDR of RT traffic due to time out. This is because in a user's queues, packet with the longest delay (provided it is not timed out) is transmitted first provided a PRB is allocated to this user. The overall system throughput is improved by exploiting multiuser diversity when users with good channel conditions are prioritised over the users with bad channel conditions.

The priority of an RT user k at time t is given by (4) below [2].

$$P_k^{RT}(t) = \left(\frac{T_k^{waiting}}{DB^{RT}}(t) \times [H_k^{RT}(t)]^2\right) + [Q_k(t)]^2 \qquad (4)$$

where, , $P_k^{RT}(t)$ is the priority of RT user k at time t, $T_k^{waiting}$ is the waiting time of RT user k, $DB^{RT}$is the delay budget of RT traffic, $H_{k \epsilon K}^{RT}$ is the channel state information of RT user k and $Q_k(t)$ is queue length of user k at time t.

NRT queue

The QoS requirement for NRT streaming video traffic is defined as $r_k(t) \geq T_k$, where $r_k(t)$ is the instantaneous throughput of user k at time t and T_k is throughput requirement of NRT user k. A QoS aware priority metric is used to sort users in NRT queue. The priority metric for NRT queue is shown in (5) It is the product of normalised waiting time, a ratio of minimum required throughput and average achieved throughput, and channel state information, of each NRT user. The priority metric reduces delay of NRT queue users by prioritising users with longer delays and improves fairness among users by allocating them fair share of time, to be scheduled. This is because when users with longer delays are put in the front of queue, then at the end users' total number of scheduling intervals become almost equal. the ratio of minimum required throughput and average achieved throughput increases the priority of users achieving low throughput and tries to allocate to each user equal or more than the minimum throughput required by NRT queue users. Multiplication of channel state information helps improving the overall system throughput by prioritising users with good channel conditions as in (4).

The priority of a NRT user k at time t is given in (5) below [2].

$$P_k^{NRT}(t) = \frac{T_k^{waiting}}{DB^{NRT}} \times \frac{T_k(t)}{R_k(t)} \times [H_k^{NRT}(t)]^2 \qquad (5)$$

where $P_k^{NRT}(t)$ is the priority of NRT user k at time t, $H_k^{NRT}(t)$is the channel state information of NRT user k at time t and DB^{NRT} is the delay budget of NRT streaming video traffic.

The time average throughput of user k, $R_k(t)$ is updated by the following formula as used in [1] [9],

$$R_k(t+1) = \left[1 - \frac{1}{t_c}\right] R_k(t) + \frac{1}{t_c} \sum_{m=1}^{M} r_{k,m}^{'}(t) \qquad (6)$$

where t_c is the length of time window to calculate the average data rate; $1/t_c$ is called attenuation co-efficient with classic value 0.001, $r_{k,m}^{'}(t)$ is the acquired data rate of user k at PRB m if m is allocated to k, else it is zero and r_k is instantaneous and $R_k$is average throughput of user k.

BE queue

BE traffic has no QoS requirements so priority is given to users based only on channel conditions. However to maintain some amount of fairness between users, classic PF algorithm is used as the queue sorting algorithm for BE. The priority metric for BE users is given below in (10),

$$P_k^{BE}(t) = \frac{r_k}{R_k} \qquad (10)$$

where $P_k^{BE}(t)$ is the priority of BE user k at time t.

B. *Adaptive TD scheduler*

After prioritising users in the queues, adaptive TD scheduler selects the most suitable users from the queues based on the priority of traffic types and the available PRB in the FD.

Packet scheduling algorithm is mainly focused on PRB allocation based on users' channel state information, traffic queue information and QoS requirements. However because of too many users and limited PRBs, it is infeasible to guarantee all ongoing users' QoS in each TTI. In this case, a TD scheduling algorithm is needed to make decisions adaptively whether to admit or reject scheduling request of a user. A novel Adaptive TD scheduling algorithm is proposed in this paper, where it chooses a pool of users from the queues of traffic classifier based on current network conditions and PDR feedback of RT traffic. This algorithm consists of two main steps. In the first step, it allocates the radio resources to RT and NRT traffics based on current traffic pattern, system load and service, and system level performance metrics. In the second step, it adjusts the RT resource allocation at the cost of minor delay in NRT traffic. This algorithm lowers the PDR of RT traffic and at the same time ensures that the minimum throughput requirement of NRT traffic is met. It is achieved by decreasing resource allocation to RT queue and allocating resources to NRT traffic queues when the PDR is lower than the threshold.

The adaptive TD scheduling algorithm works as follows.

Let the total number of available PRBs are denoted by C. Let λC be the proportion of PRBs assigned to RT traffic users and $(1-\lambda)C$ is assigned to NRT traffic users. At the first step the default value of λ is set from the built-in policies based on different parameters then the value of λ is adaptively adjusted according to PDR of RT traffic. A built-in policy defines the resources reserved for RT and NRT traffics e.g., policy (60%, 40%) means that 60% of the available PRBs are reserved for RT traffic and 40% are reserved for NRT traffic. At the start of transmission, TD adaptive scheduling algorithm uses a default policy to distribute PRBs between RT and NRT traffic types. A default policy is set at a point where the PS algorithm performs well in terms of all performance metrics thus improve the PS performance at service level and system level. This is defined as a stability region at which the PS algorithm produces balanced performance regarding all performance metrics. The default value of λ is adjusting with the current network condition. This is because the network conditions are changing rapidly in wireless environments. In this way the main challenanges in setting the default value of λ are; i) finding a stability region and ii) updating the value of λ based on changing wireless conditions. To find a stability region is subject the practical user distribution and total number of active users. It means that there may be different traffic patterns such as RT users are equal to NRT user or RT users may be lesser or more than NRT users. Similarly, the number of active users can vary with time. The default value of λ can have different values under different traffic patterns and varying system load. To set a stable default value of λ under different traffic patterns and with variable system load, a series of experiments have been done as described below.

Results Analysis of Built-in Policy

In this section an analysis is presented based on a series of simulation results which is done to make the PS algorithm work effectively under different traffic patterns and with variable system load. For this analysis, the QoS of RT traffic (delay, PDR), QoS of NRT traffic (minimum throughput), system throughput, user throughput fairness and a trade-off between system throughput and user fairness are analysed at the system load varying from 40 to 100 active users in a single cell scenario. These simulations have been conducted in the following traffic patterns.

- RT users $= NRT\ users$
- RT users $> NRT\ users$
- RT users $<$ NRT users

For each traffic pattern, simulations are run for network loads varying from 50 to 100. The reason of running these simulations is that PS performance behaves differently under different traffic patterns and different system load, and there is a need of finding out a stability region where PSA can produce balanced performance in terms of all PS performance metrics used in this paper. Fig.2 shows an example how these simulations are run. In Fig.2 the average delay of 80 users is calculated by using different built-in policies. This is to find a policy where the average delays of RT and NRT traffic are balanced. As shown, both RT and NRT traffic have a balanced delay at policy (70%, 30%). Trade-off between RT and NRT delay shows an insight how the simulations are run for analysis purpose.

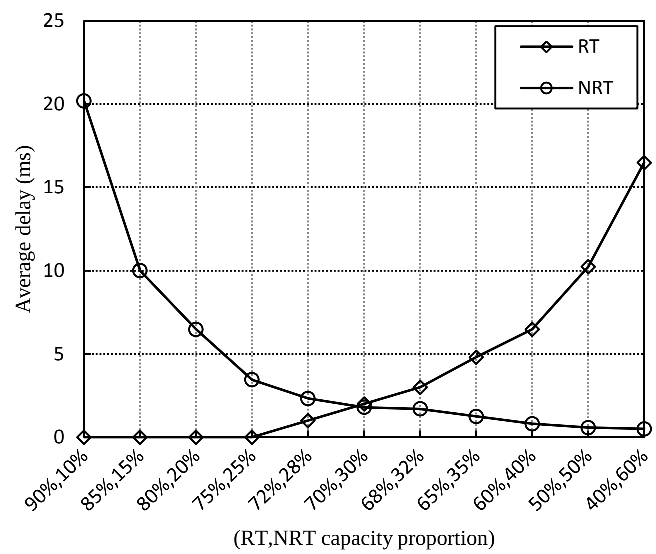


Figure 2. A trade-off between delay of RT and NRT traffic.

This trade-off value is different under different network loads for the same traffic pattern. And this trade-off value is different under different traffic patterns with the same system load. This difference appears for other performance measures such as in PDR, minimum throughput, and fairness etc. For all performance metrics, a balanced point is traced out by considering QoS requirements of RT and NRT traffic types. Based on all information, a stability region is analysed to set the default value of λ. The conclusion of the series of all experiments is as follows.

For the first traffic pattern (RT users > NRT users), if the total number of active users in the cell are more than 60, the built-in policy (70%, 30%) works well in terms of QoS of RT and NRT traffics and system throughput and user fairness. This is shown by analysis results that the system can work well at this policy for all traffic loads greater than 60. If the number of active users is lesser than 60, then policy (60%, 40%) works well and comes up with required performance guarantees. For second network condition (RT users = NRT users), policy (50%, 50%) works well for all network loads. For third network scenario (RT users < NRT users), if the number of users is greater than 60, policy (30%, 70%) works well and for a load lesser than 60, policy (20%, 80%) works well. In this way this algorithm reserves radio resources for RT and NRT traffic types based on a stability region where the proposed algorithm shows a balanced performance under variable system load and specific traffic pattern, in terms of all aimed performance metrics. The next step is to further improve QoS of RT and NRT streaming video traffic by making adaptive changed in the value of λ based on PDR of RT traffic. If the PDR of RT traffic is increased above a certain threshold, RT resource allocation is increased at the cost of minor delay in NRT traffic. And if PDR of RT traffic is lower than the threshold, RT resource allocation is decreased by diverting resources to NRT traffic types.

The adaptive change in the value of λ for the second step of adaptive TD scheduler follows the following rule (11).

$$\lambda(t+1) = \begin{cases} \lambda(t) & if\ PDR_{RT}(t) = \varphi \\ \lambda(t) + \eta & if\ PDR_{RT}(t) > \varphi \\ \lambda(t) - \eta & if\ PDR_{RT}(t) < \varphi \end{cases} \qquad (11)$$

where η is the increment/decrement of the resources reserved for RT traffic and φ is the PDR threshold set for RT traffic. Packets of RT users are dropped when they exceed upper bound of delay. PDR is calculated by QoS measure unit of the proposed PSA in each TTI and is fed back to the adaptive TD scheduler. And based on PDR value adaptive TD scheduler adaptively increment or decrement resources reserved for RT and NRT traffic.

If PDR of RT traffic is equal to φ, value of λ will not change. However if PDR is higher than φ, there will be an increment equal to η in the resource allocation to RT traffic, and if PDR is lower than φ, the resource allocation to RT traffic will be decremented by the same amount η.

This algorithm lowers the delay, delay viability and PDR of RT traffic and considers the minimum throughput requirements of NRT traffic to be satisfied at the same time. This is achieved by increasing NRT resource allocation when PDR is under the threshold.

C. FD scheduler

In the frequency domain, PRBs are mapped to the users. Multiuser diversity is exploited by using channel dependent frequency domain proportional fairness (PF-FD) algorithm. Per PRB CQI reports of each user are fed back to this stage and for each scheduling unit, the best PRB is selected and allocated to it.

V. The proposed Packet scheduling Algorithm and Permance Metrics

In this section the packet scheduling algorithm and the performance metrics for its performance evaluation are given.

A. Packet Scheduling Algorithm

A list of prioritized users is generated after applying queue sorting and adaptive TD scheduling algorithms at the classifier and adaptive TD scheduler stage of the PSA, respectively. The proposed PSA flow and the PRB allocation method is formalized in the following algorithm. At a given time *t*, PRBs are allocated to the prioritized users by this algorithm.

Algorithm: The packet scheduling algorithm.

1: Initialization: Let $k = 1,2,3,\ldots..K$, Q_{RT} = RT users, Q_{NRT} = NRT users, Q_{BE} = BE users, subject to $\{Q_{RT}, Q_{NRT}, Q_{BE}\} \subseteq K$,
Let $m = 1,2,3,\ldots..M$, and $Q_k(t)$ = *total packets of user k*;

2: Calculate $r_{k,m}(t)$ for all $k \in K$ according to (2);

3: For every user $k \in Q_{RT}$, for very user $k \in Q_{NRT}$ and for every user $k \in Q_{BE}$,
Calculate priority according to (4), (5) and (6) respectively;

4: Sort users in each queue in descending order of the priority;

5: Set the default value of λ from built-in policies based on the stability region; RT capacity = λ;

6: Allocate $(1-\lambda)$ capacity to Q_{NRT} and Q_{BE} queue;

7: Select a set of users from the queues based on capacity allocation;

8: Assign a user k with the highest priority with a PRB m,
Subject to $m = arg\ \max_m (r_{k,m})$;

9: Update $r_k = r_k + r_{k,m}(t)$;

10: Update $M = M - \{m\}$ and $K = K - \{k\}$;

11: Update $Q_k(t) = Q_k(t) - \{Transmitted + dropped\}\ packets$;

12: If $Q_k(t) \le 0$ then
Remove this user from user list K and allocate m to next user in the user set;

13: Go to step 8 if the PRB list is not empty else go to next TTI;

14: Update average achieved throughput $\overline{R_k(t+1)}$ for all users.

Figure 3. The proposed packet scheduling algorithm.

Resource allocation is completed when all PRBs are allocated.

B. Performance Metrics

We analyze the propose packet scheduling framework under performance metrics of system throughput, fairness among users and QoS of RT and NRT traffic types.

The system throughput is the sum of average throughput across all the users [20]. Individual user throughput helps calculating minimum throughput requirements of NRT users and system overall throughput is used to analyze network level PS performance in terms of system spectral efficiency.

To measure the fairness among users Raj Jain fairness index is adopted which is defined as below [20-21].

$$Fairness = \frac{\left[\sum_{k=1}^{K} \tilde{R}_k\right]^2}{K \sum_{k=1}^{K} \left(\tilde{R}_k\right)^2} \qquad (12)$$

The value of fairness index is 1 for the highest fairness when all users have same throughput such as at lower system loads. In (12), K represents total number of users and $\breve{R}_k$ is the time average throughput of user k.

User delay is equal to the number of TTIs in which the user is not scheduled and average delay of RT traffic is the total delay experienced by all RT users divided by the total number of users. The PDR is calculated by the ratio of number of packets dropped (due to time out) to the total number of RT packets as given in (14) as used in [2] [16].

$$p_k^{PDR} = \frac{n_k^{dropped}}{n_k^{total}}. \qquad (13)$$

Where p_k^{PDR} is the PDR, $n_k^{dropped}$ is total number of dropped packets by RT user k and n_k^{total} is total number of packets generated by RT user k. Overall PDR for RT traffic is calculated by taking the ratio of total packets dropped by all RT users to the total number of packets of RT users. And the delay violation probability is taken as the PDR of a user k with the maximum value of PDR out of all RT users as given by (15), as used in [2] [16].

$$Delay\ viability = \max_{k\in RT}(p_k^{PDR}) \qquad (14)$$

Where p_k^{PDR} is the PDR of RT user k. The long-term minimum throughput is taken as the minimal throughput among all NRT streaming video traffic users and is given by (15) [2] [16],

$$r_{min=}\min_{k\in NRT} r_k. \qquad (15)$$

Where r_{min} is the minimal throughput of all NRT streaming video traffic users and r_K is the throughput achieved by NRT streaming video traffic user k.

VI. SIMULATION RESULTS AND DISCUSSION

Simulation model used in all simulations is presented in this section. The results obtained are discussed in detail in this section.

A. Simulation model

The proposed PSA for LTE-A networks is simulated using a single cell OFDMA system with total system bandwidth of 10 MHz which is divided into 55 PRBs and PRB size is 180 kHz. The total system bandwidth is divided into 55 PRBs.

The wireless environment is typical Urban Non Line of Sight (NLOS) and the LTE-A system works with carrier frequency of 2GHz. The most suitable path loss model in this simulation is COST 231Walfisch-Ikegami (WI) [3] as used by many other literatures on LTE. In the simulation we assume all users are random distributed.

In the simulations, we take full buffer traffic model and packet is fixed to 180 bits/s. The first simulation is to compare the performance of the proposed PSA against the existing QoS aware PS algorithm. In these simulations the total number of RT users is equal to the total number of NRT users. The simulation results are shown in Figs. 4 to 9.

In the second set of simulation, the performances of the proposed PSA are analysed under different traffic patterns as mentioned in Section III. The second set of the simulation results compare the PS performances of the proposed PSA at different traffic patterns and variable system loads. The results are shown in Figs. 10 to 12.

The simulation parameters for the system level simulation are based on [22] and these values are used typically in most of the literatures. The simulation parameters and configurations are shown in Table 1.

TABLE 1 SIMULATION PARAMETERS

Parameter	**Value/comment**
Cell topology	Single cell
Cell Radius	1 km
UE distribution	Random
Smallest distance from UE to eNodeB/m	35 m
Path Loss model	COST 231 Walfisch-Ikegami (WI) model
Shadow fading standard deviation	8 dB
System bandwidth	10 MHz
PRB bandwidth	180 kHz
Carrier frequency	2 GHz
BS transmission power	46dBm(40w)
Traffic model	Full buffer

In each of the simulation, the delay upper bound for RT traffic is set to 40 ms [16] [23] which is equivalent to 40 time slots. The minimum throughput required by NRT streaming video traffic users is to 240 kbps as used in [2] and [16]. The total eNodeB transmission power and Bit error rate (BER) for all users are set to 46dBm (40w) and 10^{-4} respectively.

In [1], each user is assumed to have one service type and one scheduling unit (SU) carries the information about the user, service type and buffer status. However in this paper three separate traffic models are used for RT, NRT streaming video and BE traffic. For RT traffic, an “ON and OFF” traffic model is used with 35% “ON” time, and the packets are generated by using Poisson distribution. Poisson distribution is also employed for NRT streaming video and BE packet generation. The BER without buffering for RT traffic, NRT streaming and BE traffics are 0.1579, 0.8596 and 0.7448 respectively. For the BER with buffering of RT, NRT and BE, the values are 9.864e-007, 9.9219e-007 and 9.881e-007 respectively. Using the above simulation model, all simulations are run in Matlab R2009a on Windows 7 with 2.4 GHz CPU and 4-GB RAM.

B. Simulation results

The performance of the proposed PSA is evaluated against the standalone PF and QoS aware SWBS algorithm [15]. In the simulation result figures, PPSA represents the proposed PSA, PF represents the proportional fairness algorithm and SWBS represents the QoS aware packet scheduling algorithm.

First, we present results for the QoS support to RT and RT streaming video traffic. The conventional PF algorithm does not take into account QoS support to RT and NRT streaming video traffic and is not considered in these results. Fig.4 shows the average delay of RT traffic by the proposed PSA and SWBS algorithm. Both PPSA and SWBS show almost same average delay for a system load $K < 70$ as the total number of active users is small and users are frequently scheduled. At higher system loads, the average delay shown by both algorithms increases because of resource competition. However the performance of SWBS is poorer than the proposed PPSA. As can be seen for $K = 100$, average delay of PPSA is 0.56 ms which is significantly lower than the average delay by SWBS. This is because the proposed PSA is designed in such a way that it reduces user delay by giving high priority to the users with longer delays. At lower system load, both algorithms show almost the same performance because the available resources are sufficient enough to meet the requirements of all users.

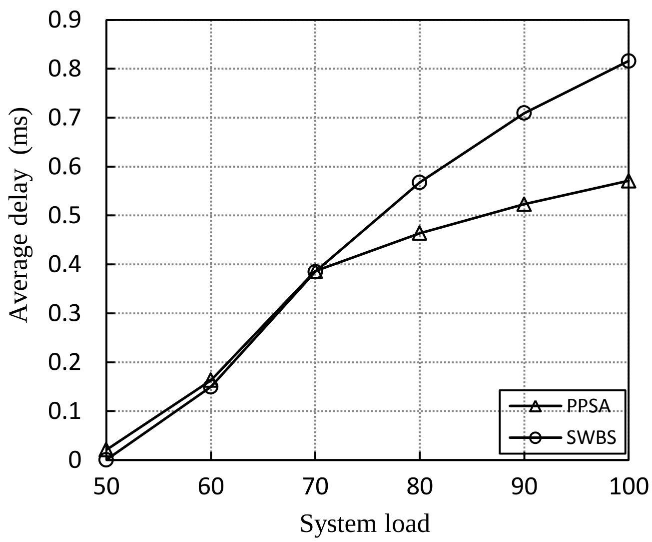


Figure 4. Average delay of RT traffic.

The PDR performance is analysed as the average PDR of RT traffic and the delay viability of RT users. The average PDR of RT traffic is calculated by (13) and is shown in Fig. 5. The performance shown by PPSA and SWBS is same for lower system loads when $K < 80$. When $K > 80$, the average PDR increases significantly with the user number. However PPSA can still maintains the best PDR performance. At $K = 100$, the PDR performance of PPSA is 30 % better than SWBS as shown. This is due to the particular design of adaptive TD scheduler in the proposed PSA as described in Section IV

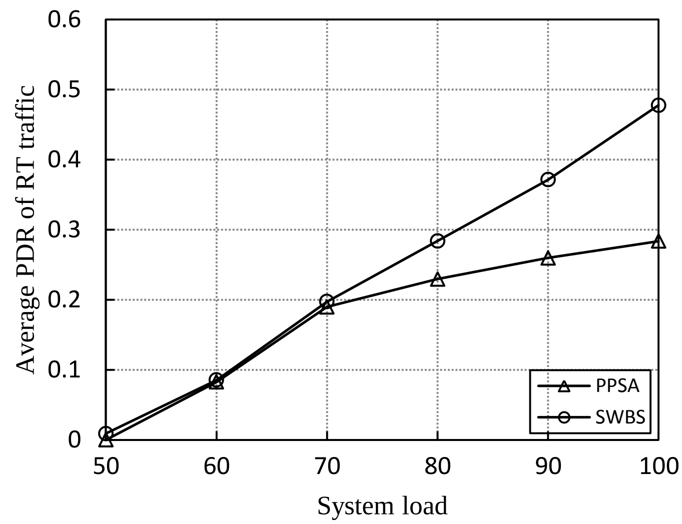


Figure 5. Average PDR of RT traffic.

The delay viability is a measure of difference of PDR among RT users and shows the highest PDR by an RT user. It is calculated by (14) and is shown in Fig.6. As can be seen, delay viability for both algorithms increases with the number of users. However PPSA shows higher performance as compared to SWBS particularly at higher system loads. The proposed PSA at its classifier stage, prioritises users with longer queues and transmits packets with the highest delay (provided it is not timed out), once PRB is allocated to the user. In this way it significantly reduces the number of dropped packets due to time out. Delay viability is further reduced by PSA when adaptive TD scheduling algorithm adaptively adjusts the radio resource allocation to RT traffic based on PDR threshold. That is why it has capability to keep the PDR of each user lower than the PDR shown by other algorithm.

The QoS support for NRT traffic is analysed by the minimum throughput of streaming video traffic as shown in Fig. 7. It is calculated by (5) for the proposed PSA, SWBS and conventional PF algorithm. The results for PF algorithm are included hereafter because it is designed to improve system throughput and fairness among users. While showing results on throughput of users and system throughput and fairness among users, PF shows comparable results.

The proposed PSA and SWBS can support minimum throughput guarantee of streaming video traffic and achieve more than required throughput$(R_k = 240kbps)$. However the conventional PF algorithm can only support minimum throughput guarantee at lower system load, $K = 50$. When $K > 50$, minimum throughput achieved by PF decreased and becomes 135 kbps at $K = 100$ as shown.

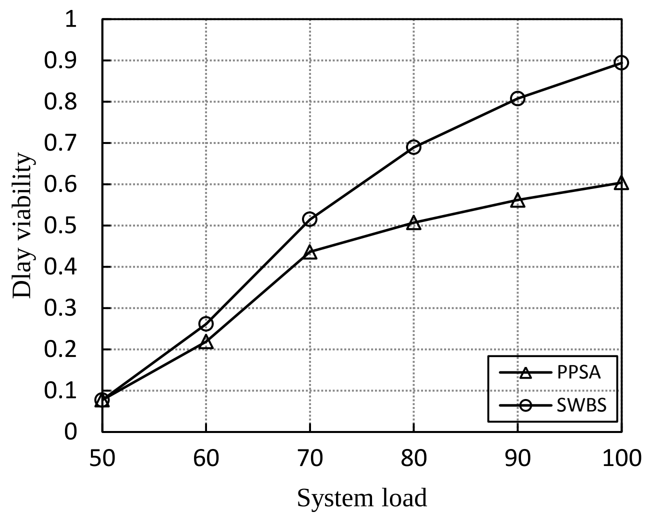


Figure 6. Delay delay viability of RT users.

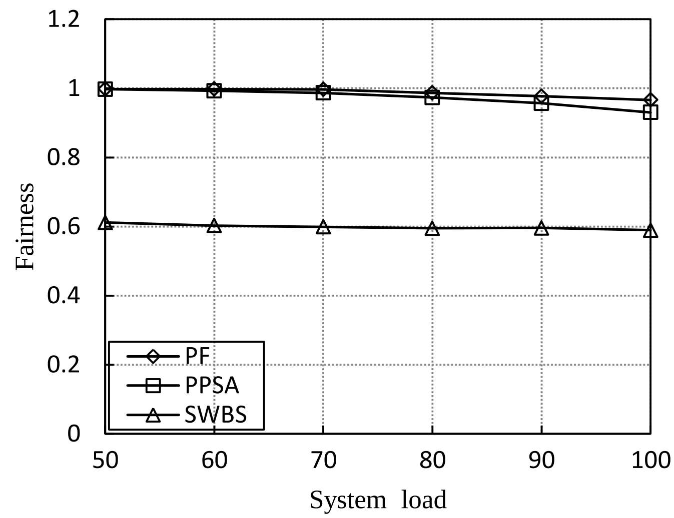


Figure 8. Fairness among users.

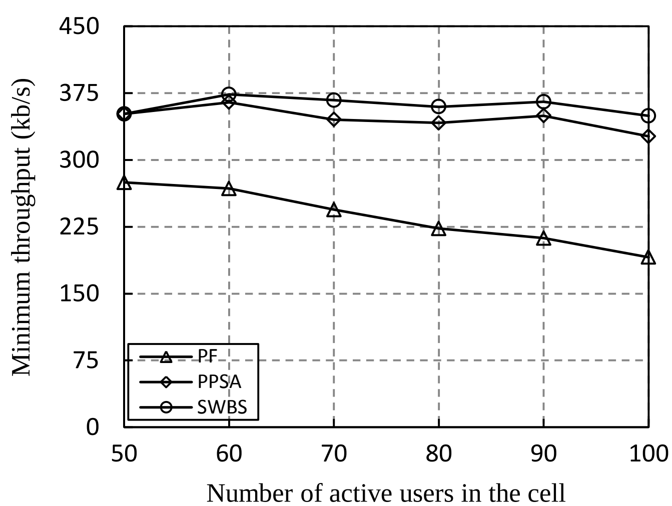


Figure 7. Minimum throughput of streaming video traffic.

The fairness performance is analysed by (12) for PF, proposed PPSA and SWBS algorithm and is shown in Fig.8. The proposed PSA significantly improves The PF algorithm shows the highest fairness among all algorithms at all system loads because it has a fairness control in its design. Fairness achieved by PPSA is almost similar to that achieved by PF up to a system load of 70 active users. However it is slightly lower than PF at system load higher than 70 active users. This is because at higher system loads, the resource competition among users increases significantly and PPSA is a QoS aware algorithm. It is designed to balance the PS performance in terms of all performance metrics. That is why at higher system load its fairness performance decreases slightly as shown. The SWBS algorithm however shows the lowest performance because in its design there it lacks fairness control.

We define the average system throughput as the average transmitted bits per second in the system [9]. Fig.9 shows system throughput achieved by the proposed PSA, SWBS and conventional PF algorithm. As can be seen, the proposed PSA achieves the highest throughput among all the algorithms. PF algorithm also achieves a high throughput because it is designed to make a good trade-off between system throughput and user fairness thus maintain system throughput at good level. However its performance results are poorer than that for PSA at all system loads. For example at a system load of 70 active users, system throughput achieved by PPSA is 24Mbps which is 6Mbps higher than PF algorithm and 9 Mbps higher than SWBS algorithm. This is because the proposed PSA exploits multiuser diversity both in the TD and FD and always gives priority to users with good channel condition. The SWBS algorithm achieves the lowest system throughput because it is only designed to improve QoS of RT and NRT traffic types and does not improve overall system throughput.

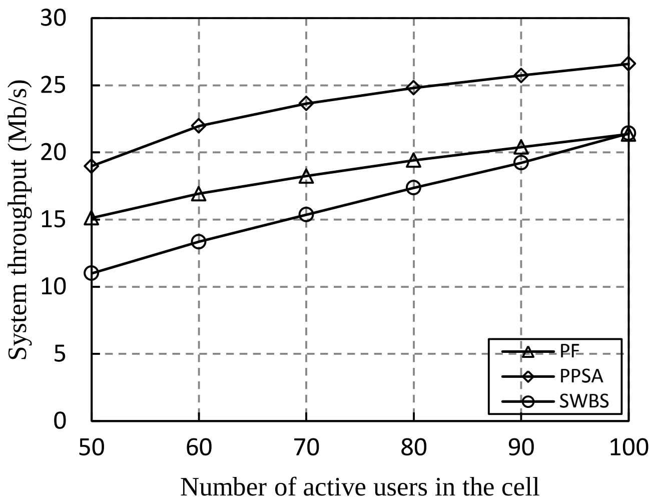


Figure 9. System throughput.

C. Performance results for different traffic patters

In this paper simulations have also been conducted to analyse the performance of the proposed PSA with three traffic patterns with number of users varying from 50 to 100. This is to prove the validity of the proposed PSA in varyig network condtions in terms of traffic patterns and system loads. In the first set of traffic pattern, the total number of active users are equally divided between RT and NRT traffic. The second set of traffic pattern consists of 70% RT active users and 30% of NRT active users. In the third set of traffic pattern, there are 30% of RT users and 70% of NRT users.

In these results the default value of λ is set according to the current traffic pattern and system load. This value is then adaptively adjusted based on PDR of RT traffic as discussed in Section IV.

In the previous set of results, the proposed PSAperformance is analysed on the service level by average delay, delay viability and PDR of RT traffic and minimum throughput of NRT traffic and on the system level by system throughput and fairness among users. In this set of simulation results, the performance results of the proposed PSA for varying network conditions are given. Thses results are shown in Figs. 10 to 12. As can be seen, there is not any huge different in the PS performance at the service and system level. And the proposed PSA is capable to maintain good performance under all varying network conditions . Average delay for RT traffic at $RT = NRT$ and $RT > NRT$ is almost equal, however its value at $RT < NRT$ is slightly lower. Delay viability at $RT = NRT$ and at $RT < NRT$ is almost same and its value for $RT > NRT$ shows very small difference at system load when $K < 60$. PDR with all traffic patterns is almost equal except at a system load when $K = 100$, where it shows verty slight difference. Thesupport for minimum throughput gaurantee to NRT streaming video traffic is well satisfied at all traffic patterns and all users achieve a throughput higher than the requirement $R_k > 240kbps$, as shown.System overall throughput value is almost same at all conditions and fairness at $RT < NRT$ is slightly lower than the other two network conditions

VII. Conclusion and future work

In this paper, we have presented a QoS aware packet scheduling architecture which is composed of three main units for the resource allocation in the downlink transmission of OFDMA-based LTE-A networks. The queue sorting algorithms at the classifier stage segregate mix traffic into service specific queues and prioritize users in these queues according to their QoS requirements. The novel adaptive TD scheduling algorithm sets a default value of radio resources for RT and NRT traffic based on traffic pattern and system load at first step. The default value is the changed adaptively based on PDR of RT traffic. In this way it helps maintaining good performance of the proposed PSA with variable conditions of traffic patterns, system load and PDR of RT traffic. In the FD the prioritized list of users is allocated PRBs in such a way that those users get the best PRB available. It helps improving the system spectral efficiency significantly. In this way the proposed PSA provides better QoS to different traffic types. It is able to improve system spectral efficiency by optimizing the use of given radio resources and maintains certain degree of fairness among users at the same time, by adaptively providing just enough resources to RT traffic and distributing extra resources efficiently to NRT services. The results show an improved QoS of RT traffic and a better trade-off between user fairness and system overall throughput. The performance comparison under different traffic patterns and with variable system loads also show that good performance of proposed PSA is maintained with variable conditions.

This work mainly focus on user-level PS performance by evaluating average delay and average PDR of RT traffic, delay viability of RT users and minimum throughput guarantees to NRT users. However packet-level PS performance may be evaluated by calculating jitter which is an importance performance metric at packet-level.

References

[1] Rehana. K., Yue. Chen, Kok. K. C., Laurie C., and John S., ”QoS aware mixed traffic packet scheduling in OFDMA based LTE-Advanced networks”, UBICOMM 2010, Copyright © IARIA 2010, pp. 53-58.

[2] Rehana. K., Yue. C., and Kok. K. C., ” Service Specific Queue Sorting and Scheduling Algorithm for OFDMA-Based LTE-Advanced Networks” Sixth International Conference on Broadband and Wireless Computing, Communication and Applications 2011, Barcelona, Spain [accepted].

[3] Harri H., and Antti T., “LTE for UMTS OFDMA and SC-FDMA Based Radio Access”, John Wiley and sons ltd 2009, pp 181-190.

[4] Martin S. (2008,April 23).WirelessMoves, 3GPP Moves on: LTE-Advaced. Last viewed 23 Jan. 2012 at 18:35 GMT. Website: hhttp://mobilesociety.typepad.com/mobile_life/2008/04/3gpp-moves-on-l.html

[5] Stefania S., Issam T., and Matthew B., "The UMTS Long Term Evolution Forum Theory to Practice", 2009 John Willey & Sons Ltd.ISBN: 978-0-470-69716-0.

[6] Jani P., Niko K.,. H., Martti M. and Mika R., "Mixed Traffic Packet Scheduling in UTRAN Long Term Evaluation Downlink" IEEE 2008, pp 978-982.

[7] Won-Hyoung P., Sunghyun C. and Saewoong B., "Scheduling design for multiple traffic classes in OFDMA networks", IEEE 2006, pp,790-795.

[8] Bilal S., Ritesh M., and Ashwin S., "Downlink Schedulinh for Multiclass Traffic in LTE", EURASIP Journal on Wireless Communications and Networking, Vol. 2009, Article ID 510617, 18 pages.

[9] Zhen K., Yu-Kwong, and Jianzhou W., "A low complexity QoS aware proportinal fair multicarrier scheduling algorithm for OFDM systems" vehiculer transaction on IEEE technology, June 2009, volume 58.

[10] Gutierrez I., Bader F., Pijoan J. L., "Prioritization function for packet scheduling in OFDMA systems", Wireless internet conference 08, Nov. 2008, Maui, USA.

[11] Andrews P., Kumaran K., Ramanan K., Stolyar A., Whiting P., Vijayakumar R., "Providing quality of service over a shared wireless link", Communication magazine, IEEE, vol.39, 2001, pp.150-154.

[12] Suleiman Y. Y., and Khalid A. B., "Dynamic buffer management for multimedia QoS beyond 3G wireless networks", IAENG International Journal of computer science, 36:4, IJCS_36_4_14, Nov. 2009.

[13] Toskala A., and Tiirola E., "UTRAN Long Term Evaluation in 3GPP," Proceedings of IEEE Personal Indoor and Mobile Radio Communications Conference (PIMRC'06), September 2006.

[14] Sanjay. S., and Alexander L. S, "Scheduling for Multiple Flows Sharing a Time-Varying Channel: The Exponential Rule" Bell Labs, Lucent Technologies, NJ 07974.

[15] Parimal P., Srikrshna B., and Aravind R., "A subcarrier allocation algorithm for OFDMA using buffer and channel state information", Vehicular Technology Conference, 2005. VTC-2005-Fall.2005 IEEE 62nd, pp.622-625.

[16] Jun S., Na Y., An L., and Haige X., "Opportunistic scheduling for heterogeneous services in downlink OFDMA system," School of EECS, Peking University, Beijing, P.R.China, IEEE computer Society 2009, pp.260-264.

[17] Haipeng L. E. I., Mingliang, A. Z., Yongyu C., and Dacheng Y., "Adaptive Connection Admission Control Algorithm for LTE Systems", IEEE 2008, pp. 2336-2340.

[18] Kian C.B., Simon A., Angela D., "Joint Time-Frequency Domain Proportional Fair Scheduler with HARQ for 3GPP LTE Systems", IEEE 2008.

[19] Leandros T., and Anthony E., "Dynamic Server Allocation to Parallel Queues with Randomly Varying Connectivity" IEEE Transaction on Information Theory, Vol. 39, N0. 2, March 1993, pp. 466-478.

[20] Lin X, and Laurie C. "Improving fairness in relay-based access networks," in ACM MSWIM, Nov.2008, pp. 18-22.

[21] Chisung B., and Dong-Ho C., "Fainess-Aware Adaptive Resource Allocation Scheme in Multihop OFDMA Systems," Communications Letters, IEEE, vol.11, pp. 134-136, Feb. 2007.

[22] 3GPP TSG-RAN, "TR25.814: Physical Layer Aspects for Evolved Utra". Version 7.0.0,June 2006.

[23] Ekstrom H., Furuskar A., Karlsson J., Meyer M., parkvall S., Torsner J., and Wahlqvist M., "Technical Solution for 3G LTE," IEEE Communications Magazine", vol. 44, March 2006, pp.38-45.

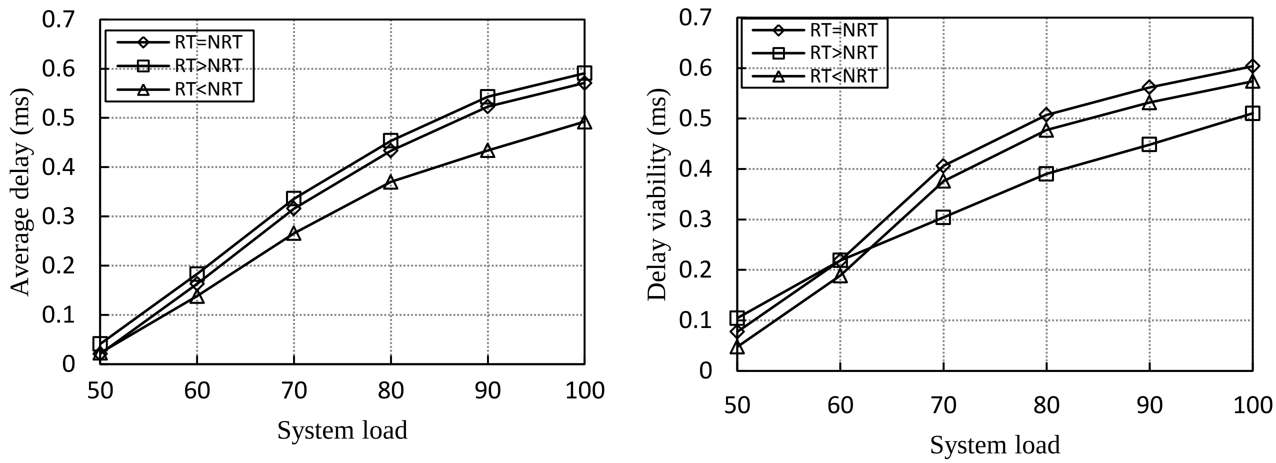


Figure 10. Average delay and delay viability of RT traffic.

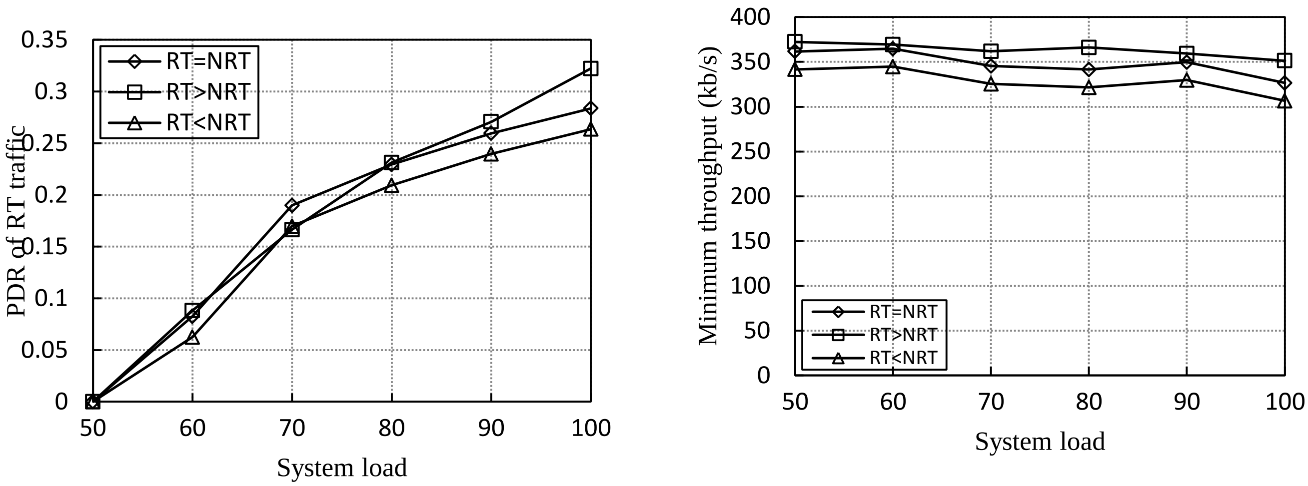


Figure 11. PDR of RT traffic and minimum throughput of NRT traffic.

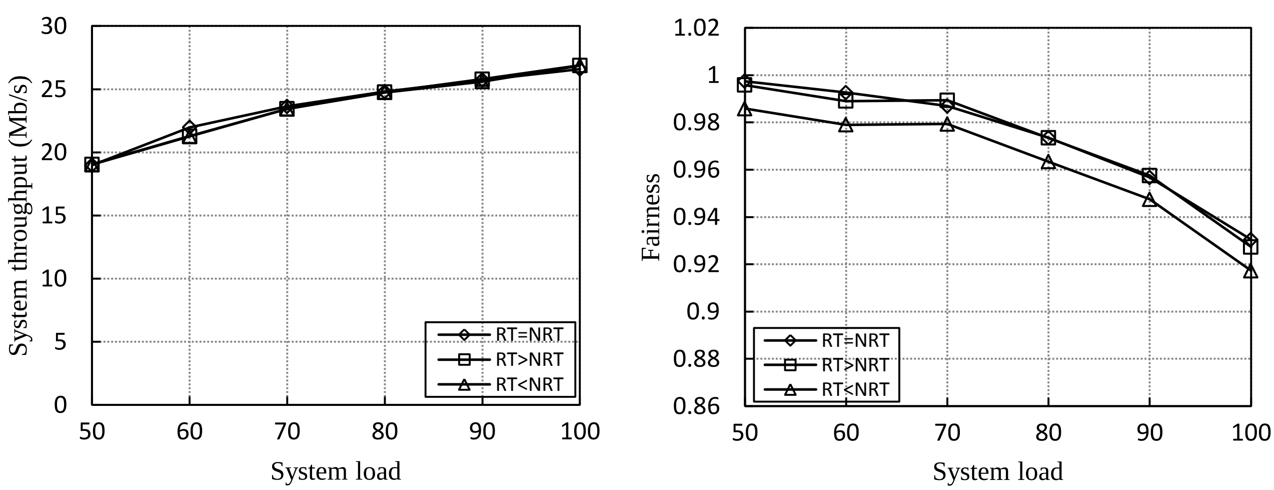


Figure 12. System throughput and fairness among users.

A Modular Platform for Wireless Body Area Network Research and Real-life Experiments

The ASE-BAN Testbed

Rune Hylsberg Jacobsen, Finn Overgaard Hansen, Jens Kargaard Madsen, Henrik Karstoft, Peter Høgh Mikkelsen, Tore Arne Skogberg, Esben Sune Rasmussen, Claus Andersen, Michael Alrøe, and Thomas Skjødeberg Toftegaard

Aarhus University School of Engineering, Aarhus University,
Aarhus, Denmark
e-mail: {rhj, foh, jkm, hka, phm, tas, esr, clan, ma, tst}@iha.dk

***Abstract* – The paper presents ASE-BAN, a wireless Body Area Network (BAN) developed at Aarhus University School of Engineering (ASE). ASE-BAN is a modular platform enabling research in the healthcare area and allowing real-life experiments with real users. The paper presents requirements, architecture and implementation of a hardware platform consisting of different modules, the current progress of research with development of sensor nodes for this ASE-BAN and the corresponding software. The concept of a body gateway is presented alongside with the preliminary results obtained with our current wireless BAN prototype.**

Keywords-low power; wireless sensor network; WBAN; Body area network; BAN; healthcare; Body gateway; testbed; IEEE 802.15.4; 6LoWPAN.

I. INTRODUCTION

There is an increasing need for personal home healthcare due to a growing population of elderly people [1]-[3]. To support the health problems of the elderly population wireless sensor technologies have enabled new types of applications for monitoring and controlling people's physiological parameters.

The first generation of e-healthcare solutions were more or less replacement of a wire with a wireless communication channel, i.e., another set of protocols on top of a new physical communication media. In the second generation, the devices communicated wirelessly with a local system host, which relayed alarms and possible also data to remote sites. In the third generation the healthcare sensors and actuators are wirelessly connected to a mobile body area network.

Miniaturization and cost reduction of modern electronics facilitate the assembly of tiny and affordable wearable devices for real-time monitoring systems of personal medical data. There is an increase in the demand for such devices, partly due to the demand for highly person-centric and prevention-based health-related services and partly because of the relative increase in number of elders in the developed countries.

A wireless network system can be set up, where network devices communicate accurate personal medical data to a host for storage or post-processing. The data may also be sent to medical practitioners such as caregivers and physicians for examination and diagnostic purposes. This enables greater mobility; reduces hospitalization, and results in better welfare at reduced costs for the society. The system provides ease in information-flow from the user to the central server or the doctors and caretakers, in a convenient and secure way.

A system for wireless real-time monitoring of physiological data from a body can be organized in a wireless BAN [4], as illustrated in Figure 1. The BAN consists of a number of different sensor and actuator nodes interconnected by using wireless communication with a body gateway. Sensors can be devices for picking up physiological signals from the body, e.g., electro-cardiogram (ECG) sensor used to monitor cardiovascular activity, an oximeter sensor used to monitor pulse and blood oxygen levels etc. Another example is an actuator node that can be used to stimulate muscle activity.

The body gateway communicates wirelessly with a local or a remote host application at a home base station or a

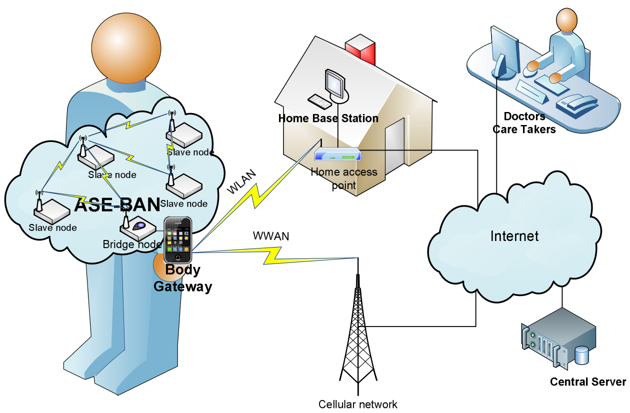


Figure 1. A wireless body area network system.

remote central server as shown in Figure 1. Some types of sensor nodes may acquire large quantities of medical information in real-time. Subsequently, data must be sent to the host for storage or post-processing from time to time.

Since wireless transmission is relatively energy costly, the gateway should only transmit context relevant data when needed, to minimize energy consumption. This is one of many requirements for a well-designed BAN.

As described above the concept of a BAN has over the last decade been researched intensely. This research involves investigating different themes raging from wireless propagation models locally around the body to how to design a feasible physical wearable low-power small scale network. The work described herein is an extension and an elaboration of our recent publications [1][5]. The target has from the beginning been to design a modular, wireless BAN testbed based on state-of-the-art wireless communication technology that can connect different wearable biomedical sensor nodes as integrated system components. To achieve this it was decided to make proprietary hardware that, over time can be re-designed and enhanced according to different design parameters like power consumption, physical size, price etc.

Figure 2 shows a picture of the hardware modules that can be mounted on top of each fitting in a small "box". The use of an open source software platform with a high degree of flexibility was subsequently chosen.

This paper is organized as follows. In Section II state-of-the-art for wireless BANs is described. Section III provides details about for requirements for the design of a BAN. Section IV describes the architecture and design of ASE-BAN. Section V goes into details about the current implementation status of ASE-BAN including the specific sensors integrated into the system. Especially, the fluid balance sensor and the ECG sensor node are described. Section VI presents the demonstrator and the preliminary results made so far. The paper concludes with a discussion of future research directions in Section VII.

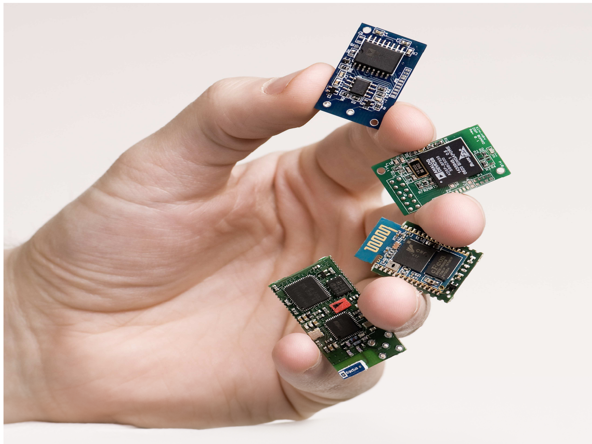


Figure 2. Four hardware modules of the ASE-BAN testbed.

II. STATE-OF-THE-ART FOR WIRELESS BANS

The wireless BAN has been a topic for research and development during the last ten years and several surveys on wireless BAN and their application in mobile health and telemedicine have been published in the literature [6]-[16]. The growing interest in building large-scale BANs across a public healthcare system such as a hospital have fueled a large number of research and development projects such as OpenCare [2], MobiHealth [6], MIMOSA [17], CodeBlue [18], SMART [19], AID-N [20], CareNet [21], ASNET [22], MITHril [23], WiMoCa [24] to mention a few. Whilst there are many similarities among the different approaches taken by research groups, the research domain suffers from a large fragmentation.

1) BAN sensor nodes

Essentially, wireless BANs are used to transmit physiological data such as vital signs by using radio wave communication. Most body sensors are utilized in an event-driven fashion, but BANs also need to support data streams for real-time monitoring [25]. Analysis of sensor data streams in BANs involves identifying and extracting the set of attributes or characteristics from each multi-dimensional time series that correspond to different performance goals of health monitoring applications.

To better monitor a human's vital signals, behavior, and the surrounding environment, a wide range of commercially available sensors can be deployed, such as accelerometer and gyroscope, as well as traditional medical sensors including electroencephalography (EEG), electromyography (EMG), electrocardiogram (ECG), blood pressure, pulse oximetry (SpO_2), respiratory inductive plethysmography (RIP), carbon dioxide (CO_2), and so on. Accelerometer sensors, along with visual and biosignal sensors, are utilized to characterize movement and to detect falls of the user [26]. Finally, ambient sensors measure environmental phenomena, such as humidity, light, sound pressure level, and temperature.

Recent technological developments have enabled sensor miniaturization, power-efficient design and improved biocompatibility. Issues related to systems integration, low-power sensor interface, and optimization of wireless communication channels are active research fields. With advances in MicroElectroMechanical systems (MEMS), sensor devices are getting even tinier in size. These are changing the traditional way of measuring human physiological parameters.

2) Radio communication

A wireless BAN is a radio frequency-based wireless networking technology that interconnects tiny nodes with sensor or actuator capabilities in, on, or around a human body. As such the topic has fueled research in the area of a body-centric wireless communication channel [27]. Antennas and propagation for telemedicine systems can be considered in two parts, those for systems outside the body and those that communicate with internal implanted sensors and devices. The increased interest in wireless channels on the body has led to a review of the types of propagation mode that may occur on the body. Use of the Medical Implant Communication System (MICS) at 402 MHz to

405 MHz, allows bands of 300 kHz to be achieved. However, due to the high availability of components for wireless body sensor networks both the industrial, scientific, and medical ISM bands between 400 MHz and 2.45 GHz, and the ultra-wideband (UWB) frequency allocation between 3.1 GHz and 10.6 GHz are frequently seen in actual implementations [27][28]. More recently there is an interest in investigating the performance of BANs operating at millimeter wavelengths and in particular at 60 GHz [29][30]. Looking at BANs, a more generally attractive alternative by using the radio channels for communication between sensors is to have bio-channels serving as a unique secured means of communication, where the human body is used to transmit either exogenous or endogenous information [31].

3) Networks and standards

Emerging and existing standards for wireless BANs and Wireless Personal Area Networks (WPANs) include Bluetooth Low-Energy, UWB, and ZigBee. However, proprietary and open technologies like Z-Wave [32], ANT [33], RuBee (IEEE 1902.1) [34] and RFID [35] have been utilized as well. Z-Wave is a proprietary mesh networking technologies for home automation. It works in the 900 MHz band. ANT is another proprietary sensor networking technology, featuring a simpler protocol stack and lower power consumption [36]. It implements a light-weighted protocol stack, ultra-low power consumption, and a data rate of 1 Mb/s. ANT has been embedded in some Nike shoes to collect workout data and it is able to talk to iPod products. RuBee and RFID are both used for logistics applications.

There have been many academic research projects utilizing IEEE 802.15.4 for transmitting health-related data [37][38]. These are based on IEEE 802.15.4 chips such as the CC2420 and the CC2430 from Texas Instruments. Implementations do seldomly use the higher-layer ZigBee protocol stack because either networking capability is not a must, or researchers are interested in devising more appropriate protocols.

Body area networking, i.e., networking among devices in, on, and around the body poses unique challenges for resource allocation, sensor fusion, hierarchical cooperation, quality of service (QoS), as well as security and privacy. Hierarchical aggregation, topology control, star and star-mesh hybrid topologies, coordination, multi-hopping multi-hop data forwarding [39]. In terms of research and development mesh networking and energy-efficient routing in BANs are still open issues. On the one hand, minimalistic networking schemes increase system run-time and reduce obtrusiveness. However, this could jeopardize QoS or privacy, which is unacceptable for life-critical or sensitive medical applications. Other topics related to the practical applications of body sensor networks such as multi-sensor data fusion, decision support, and technological scaling are also important.

Technologies for inter-BAN communication are mature, and include: WLAN, Bluetooth, Zigbee, cellular (GPRS), and 3G/UMTS etc. The more communication technologies that a personal server supports the easier it is for a BAN to be integrated with other applications.

On-going work within the IEEE 802.15.6 Task Group 6 aims at supporting applications with various data rates, where quality of service guarantees are crucial in case of life-threatening conditions [40]. An emerging BAN standard, IEEE 802.15.6 will likely employ UWB.

4) MAC layer

At the MAC layer, there is a tradeoff between reliability, latency and energy consumption that needs to be resolved. Normally, an asynchronous MAC mechanism, such as carrier sense multiple access with collision avoidance (CSMA/CA), is used with IEEE 802.15.4 to deal with collisions. To increase the lifespan of these sensors, energy-efficient MAC protocols will play an important role. Corroy and Baldus present in [41] a comparison between different low-power MAC layers. S-MAC [42], T-MAC [43], and TRAMA [17] use their transmission schedule and listening periods for synchronization and to maximize throughput, while reducing energy by turning off radios during much larger sleeping periods. On the other hand, low-power listening (LPL) approaches such as WiseMAC [44] and B-MAC [45] use channel polling to check if a node needs to wake up for data transmitting or receiving. Hereby the necessity of idle listening is reduced. Several other power-efficient MAC protocols have been developed and investigated. MAC protocols have been surveyed in [46]. It has been shown that many MAC protocols offer better performance in terms of the end-to-end packet delay and energy saving compared to the IEEE 802.15.4 MAC.

5) 6LoWPAN

The Internet Engineering Task Force (IETF) has led the specification of 6LoWPAN or Internet Protocol (IP) version 6 over low-power wireless personal area networks [47]. The approach has been to define modifications to IPv6 that allow it to be used over the IEEE 802.15.4 MAC/PHY layers. By using IP for the higher networking layers, the sensor network is interoperable with other IP networks including the Internet. This has the potential of making gateway devices simpler. The use of IEEE 802.15.4 allows the requirements of wireless BAN for low power and long lifetimes to be met. In addition, IPv6 has an addressing space adequate for all conceivable sensor networks. It also has the advantage that it is an established technology with an extensive set of support tools for development, design, control and reconfiguration. 6LoWPAN allows existing standards to be leveraged, rather than fostering a need to build standards from the beginning.

6) Software frameworks, middleware and OS

Many BAN projects use the open source operating system TinyOS [48] designed for small wireless devices. Another emerging operating system for wireless sensor networks in general is Contiki [49], designed for the Internet of Things.

Waluyo *et al.* presents in [50] a lightweight middleware for personal wireless body area networks designed to reside in personal mobile devices. A middleware taxonomy together with examples of current middleware projects grouped according to the taxonomy.

To support communication between the central located server computers and the body gateway, frameworks can be built to support the IEEE 11073 standard implementing a composite IEEE 11073 agent consisting of the body area network sensors communication with an IEEE 11073 manager on the central server [51]. On the sensor side, one project [52] has recently implemented the IEEE 1451 smart transducer standard in a BAN context [53].

III. General Requirements for Wireless BAN

Requirements in this section are mainly requirements which will have an influence on the system architecture for the BAN. It is thereby not an attempt to define a complete set of application-oriented functional requirements, which normally are defined by the use case technique. More technical requirements are currently being defined by the IEEE 802.15 WPAN Task Group 6 (TG6), which define the requirements for a WPAN [54]. An overview of these requirements, current challenges and wireless technologies for BANs are presented by Patel and Wang [3].

First, the user related requirements are described, followed by a set of more general system requirements. Most of these requirements have an impact on the hardware architecture and partly also on the software architecture. A subset of these requirements is also listed by Shnayder *et al.* [55].

A. User Related Requirements

1) Diverse User Group

Users of the BAN can for example be elderly persons living at home or in a nursing home. It can be physically disabled persons at all ages,; it can be persons suffering from dementia; it can be persons with chronicle diseases at all ages; and it can be athletes. Some of these users have several of these characteristics e.g., an elderly physically disabled person with a chronic disease.

In this way, a very diverse user group, spanning from young to very old, and in some cases people suffering from dementia, can be addressed. These different types of users have very different needs and different skill levels for handling new technology. The user group with dementia and disabled people raise the largest challenge for healthcare developers. This leads to the first challenge:

Challenge 1: Dealing with very diverse types of users, with different application needs and different skill levels.

Requirements: Adjustable technology, user friendly, easy installation and configuration of software and hardware**,** easy to add new functionality, sensors, and actuators.

Development of a BAN system for this diverse user group will benefit from using a user driven innovation and development process.

2) User Communication

The BAN should support different ways of communicating with the user. It could be by messages, LED lamps and sounds; it could be by speech syntheses or speech recognition, by activating normal buttons or soft buttons on a touch screen. Another possibility is communication with hearing-aids or headphones. Some of these devices can be used to give reminders to the user e.g., a reminder to take medicine or to exercise or to measure blood pressure.

Challenge 2: User interface design for a diverse user group.

Requirements: User friendly and easy to use interfaces.

This could be obtained by conducting usability studies with different user groups and different types of interfaces supported by incorporating industrial designers in the design team and process.

3) Calling for Help

The BAN should support a “call for help” device so a user can call for help at any time. This functionality could be supplemented with a voice-channel so the caretakers can communicate with the user.

Challenge 3: To offer safety and security to users.

Requirements: Physical design of a reliable call-device and a reliable system for transferring this event, as this could be an emergency call.

4) GPS Outdoor Positioning

The BAN should allow the connection of a GPS-device for locating people in case of an accident. It could for example be demented people who left the nursing home without supervision or a user getting a heart attack outside the home. As a GPS-receiver is a power demanding device, the receiver should be controlled by the BAN and the connected system so it only works on demand and therefore only use power in a short time frame.

Challenge 4: To locate a user in case of an accident.

Requirement: Outdoor navigation using GPS.

Ideally, indoor positioning is also relevant. However, this is currently much more challenging and not part of our current research scope.

5) Fall Detection

The BAN should support a fall detection device node with the purpose of sending an automatic call for help. It could be in situations where the user is unconscious after a fall or it could be a person with dementia, who could not operate a call button or a call device.

Challenge 5: Reliable detection of a fall.

Requirement: Physical design of a tiny and reliable fall detection node integrated on the person e.g., in the cloth or in a belt or as a decorative, personalized object.

6) Mobility

The user should be allowed to move freely around. For example a heart ECG monitoring should take place indoors in a private home or at work as well as outdoors and in public places.

Challenge 6: To be anytime and anywhere connected.

Requirement: Seamless connectivity over heterogeneous networks with automatic roaming supporting indoor as well as outdoor communication over Wireless Local Area Network (WLAN) and Wireless Wide Area Network (WWAN).

7) Physical Constraints for BAN Components

All the BAN components are connected with wireless technology and should be integrated in the person’s daily life. This raises specific requirements for the physical design, i.e., it should have a small form factor, be light-weighted, and have a smart design. Some of the devices requires skin contact and could be integrated in a plaster; some could be integrated in the cloth as an intelligent textile and some should be visible e.g., a device with user interaction for example integrated in the body gateway.

Challenge 7: Obtaining user acceptance of healthcare technology devices and wearing.

Requirements: Low form factor, low weight and easy installation, wearing and a nice-looking design.

8) Power Consumption

With the diverse user group in mind it is difficult for these users to handle battery exchange and charging of a number of sensor nodes. For general convenience the devices should be developed as low-powered devices with either long battery life or by utilizing some kind of energy harvesting technique. This leads to the architectural design with essentially only a single power demanding unit – the body gateway.

Challenge 8: Low-powered devices with energy-efficient communication.

Requirements: There is a demand for low powered devices (nodes) and communication protocols.

The body gateway requires more power and could be charged e.g., by induction or by a normal power charger with the inconvenience for the user and problems with being offline.

9) Economics for a BAN

The technology can help reducing the workload with caregiving, but with a cost of the new healthcare technology. With the high volume of users there are strict requirements to the solutions to be as cheap as possible both in buying, installation and operation.

Challenge 9: Obtaining low total system cost and operation cost.

Requirements: Low system costs and low cost of system operation, especially for the mobile communication part, which currently can be quite expensive.

B. General System Requirements

1) Security and Safety Issues

It is important that the BAN and the rest of the infrastructure are both safe and secure. Person-related information is normally regulated by national law and should be transferred in a safe and secure manner. Another problem could be external hackers which could threaten for example a close-looped application connected to a medicine injection pump. Person-related information is to be handled with confidentiality and a BAN sets strict requirements to the handling of this information.

Challenges 10: Obtaining a safe and secure system.

Requirements: Use of standard encryption techniques and authentication protocols.

2) Healthcare Application Flexibility

The BAN should support the possibility to place the application or business logic code on different components in the architecture. It could be on a sensor node, on the gateways, or on one of the connected servers. Implementing an application on a sensor node, doing pre-processing of the signal, can reduce the communication bandwidth and thereby save power, but at the cost of a more expensive sensor node.

Challenge 11: Obtaining a flexible software and hardware architecture with different processing capabilities.

Requirements: An adjustable software framework or structure for application code and flexible component-oriented hardware architecture.

An automatic configuration of the application and sensor node software is a clear goal.

3) Monitoring Data Types

Data types can be real-time, life-critical application data: ECG data as well as sporadic event data for example alarms and emergency calls for help.

Challenge 12: Very diverse requirements for signal monitoring.

Requirements: Support for continuous real-time monitoring as well as for events. See [57] and [2] for a list of technical requirements for different applications with bit rates from less than 1 kb/s for drug dosage and up to 10 Mb/s for video imaging.

4) User Identification

The BAN should support an identification mean so the user can be unambiguously identified by supporting systems and the identification can be send with the collected data to remote servers.

Challenge 13: To obtain an unambiguous and secure identification.

Requirement: A secure identification of the user is required for the BAN system.

5) Node and Person Matching

The BAN should support a mean for unambiguous identification of sensor and actuator nodes on a given person and connect these devices with the user's identification code. In this way the sensor data can be linked to a given person. A problem occurs when a sensor node connects to nodes on other persons BAN in near vicinity of the person.

Challenge 14: Matching nodes with the person wearing the wireless node**.**

Requirement: For a secure and easy identification method.

This could for example be obtained by using Body-coupled communication (BCC) where the BCC is used to discover an identify sensor nodes on the same body as presented in [57].

6) Open Standards and Open Source

The BAN should be based on open international standards for supporting as many BAN devices from different vendors as possible and with different types of functionality. The Continua Health Alliance [58], a non-profit coalition of more than 200 member companies, has defined interoperability goals for wireless systems and the IEEE group is working on a standard for wireless personal area networks [57]. The Continua Alliance material and software are mainly openly available for members of the alliance.

Challenge 15: Development of open standards for the BAN.

Requirements: Base BAN on open standards and optionally also open source software solutions for BAN components.

7) Network Topology and Communication

The BAN should work with any kind of network topology from a star network with bidirectional communication between gateway, sensors, and actuators, to a meshed network that allows communication between all nodes. It is critical to have a network infrastructure and related communication protocols that minimize the power consumption of this part of the BAN as well.

Challenge 16: Design a network with ultra-low power, secure and reliable communication.

Requirements: Support for star and mesh topology.

IV. System Architecture

The system architecture is a conceptual model that allows components to be added, removed, and modified. It allows data to be collected based on information requests. It provides a framework to abstract the underlying hardware resources from the applications and may be implemented as a middleware [59]. The system architecture of ASE-BAN is defined in order to describe the structure, behavior, and the different views of the ASE-BAN system. The architecture can be deployed in both indoor and outdoor environments. It can extend existing healthcare infrastructures such as e.g., OpenCare [2] and can be generally integrated into it infrastructures by use of web services [60].

A. *System Context for ASE-BAN*

The overall design guideline for the ASE-BAN is to have a body gateway node acting as the link from the body network to external systems, a central server or a home base station as shown in Figure 3.

This body gateway should be the only power demanding component with a longer communication range supporting both wireless Local Area Network (LAN) and Wireless Wide Area Network (WWAN) communication and with a seamless handover between the two network types.

The other BAN nodes should be ultra-low power sensor or actuator nodes with a limited communication range, i.e., less than one meter, where the communication power level can be adjusted to the minimum required for getting a reliable on-body communication.

Figure 3 shows a domain model for a complete healthcare system including the ASE-BAN system which is mounted on the indicated user.

When the user is at home the communication will be over WLAN from the body gateway and it can typically send both alarms and monitored data from the BAN to the home base station component e.g., a touch screen based computer. If the user leaves his or her home the BAN will automatically stop sending real-time monitoring data and store them locally on the BAN gateway component and only communicate alarms and keep-alive signals over the WWAN (e.g., GSM or UMTS).

This solution is previously proposed by Saadaoui and Wolf [61]. It saves communication cost, i.e., both power and money. The principle of having a central server and a home base station is implemented in the OpenCare project described in [2], where the BAN is described as a Mobile Tier component for communicating a single physical value from a user and not as being a part of a body area network.

The idea of having a powerful gateway for the body area network is also described in the work by Jovanov *et al.* [62] and Otto *et al.* [63] where they describe a three tier system consisting of tier 1: wireless BAN nodes, tier 2: personal server and tier 3: central systems. On their wireless BAN each node communicates in a star network topology with the personal server, i.e., the gateway.

For ASE-BAN both a star and a mesh network topology have been used as possible network solutions. The mesh configuration enables ultra-low power communication and communication in difficult setup's e.g., from a person's back to his/her front KKK. Another important difference, in relation to the work described in [61], is the introduction of the home base station component, which gives another level of service to the users living in a private home; for elderly people normally one or two persons. The home base station collects monitoring data from the BANs on the people living in the house and it also supports shared and non-personal related healthcare devices in the home, which assist the residents with staying healthy. This could be medicine dispenser automation, a blood pressure meter or a smart weight scale, which can have one or more users. Using a home base station enables the development of healthcare applications which take decisions based on inputs from several different sources, i.e., BAN sensors or from the shared devices.

Another advantage with the WWAN enabled ASE-BAN is the extra security obtained by having a backup channel for alarms in case of malfunctions in the normal data flow from BAN to home base station and to the central server.

B. *ASE-BAN System architecture*

The ASE-BAN system architecture, shown in Figure 4, enables continuous transmission of medical data for

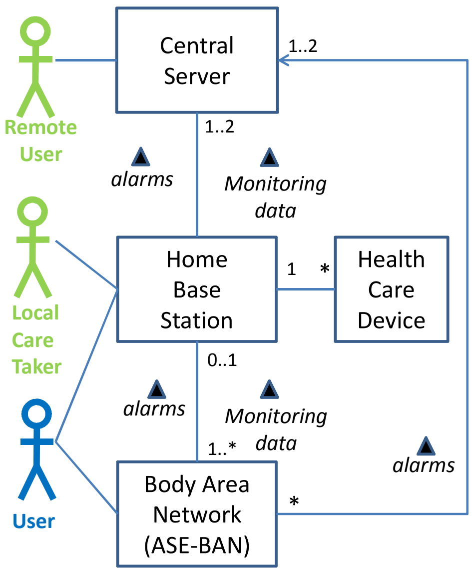


Figure 3. System domain model including ASE-BAN.

the reporting of vital signs as well as transmission of alarms. The architecture scales well in terms of the number of users and the number of sensors and actuators on each user. It supports the normal behavior of an elderly person. The ASE-BAN system is composed of a number of sensor nodes and a body gateway. Each BAN is private for one user.

The body gateway is a powerful, wireless node providing connectivity between the BAN and external systems. It contributes in the collection and processing of physiologic data from the BAN. An ASE-BAN installation can be extended gradually with sensors nodes to fit the intended purpose.

The body gateway is equipped with at least two radio components – one for the communication within the BAN and one for interconnecting with external systems. The BAN forms a Wireless Personal Area Network (WPAN) for its internal communication. When the BAN user is at home the communication will be over WLAN to the Home Base Station whereas the ASE-BAN will rely on public telecommunication networks and use WWAN when the user moves outside his/her premises.

In addition to storage and a CPU, the body gateway may be composed of one or more input-output (IO) units to support human intervention such as keypad, microphone, loudspeaker and a display. Moreover the body gateway may support the interaction with systems in proximity by using Near-Field Communication (NFC) units and can be equipped with sensors that complement the sensor nodes of the BAN such as a GPS receiver, camera etc. The storage module can be used to store physiological data signs in outdoor environments to reduce the high communication costs of WWAN networks.

The sensor node combines one or more sensors with a low power processing unit (CPU) and the WPAN communication unit, i.e., the Radio. Its main function is the sampling and pre-processing of physiological data and to participate in the communication within the BAN. A special sensor node configuration, called the relay node, acts as a relay or a router for the wireless mesh network communication. It may be used to enhance the robustness of the communication by means of multi-hop communication. One possible usage is to ensure that a sensor node located on a person’s back can communicate with a body gateway placed on a person’s front.

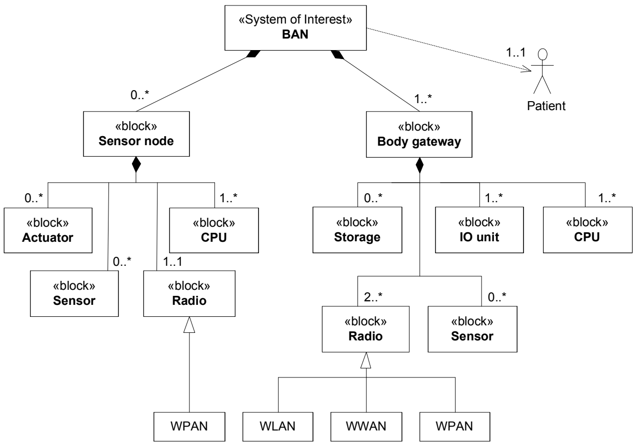


Figure 4. Block definition diagram describing the ASE-BAN architecture.

For more computational demanding sensor nodes an additional CPU such as a Digital Signal Processor (DSP) can be added to the sensor node. This allows for a distributed data processing in the ASE-BAN.

The candidate wireless technologies are based on the standards IEEE 802.11b, IEEE 802.15.1 and IEEE 802.15.4. In the testbed the latter IEEE 802.15.4 standard is used for communication within the BAN.

Table I lists the sensor nodes supported in ASE-BAN. Sensor nodes have been classified according to reaction types: Continuous or Event. Sensor nodes of the Continuous type is used for the continuous monitoring of physiological data whereas the Event type is used for the issuing of alarms when a pre-determined threshold is met e.g., low fluid balance or in case of a fall.

C. Software platform

1) Software processing capabilities

The architecture supports running software on different places. The CPU at the sensor node can be of different types from a simple microcontroller to an advanced digital signal processor that allows advanced preprocessing of the sensor signals and the execution of application algorithms. An example of an advanced preprocessing is the Heart Rate Variability (HRV) detection or ECG signal supervision for heart artifacts. This gives the possibility only to send alarms in case of malfunction and in this way limits the power demanding wireless communication. The next application level is on the body gateway, which normally has a powerful processing capability. The software running on this platform can correlate signals from several sensor nodes and in this way take decisions based on multiple sensor inputs.

The next level of processing is performed either on the home base station or on the remote central server communicating with the ASE-BAN via the body gateway component, when the user is away from home.

TABLE I. ASE-BAN SUPPORTED SENSOR NODES.

Sensor node	Description		
	Key function	***Reaction type***	***Data rate***
Temperature	Ambient temperature monitoring	Continuous or Event	Low
Fluid balance	Monitoring of the fluid balancing	Continuous or Event	Low
Electrocardiogram (ECG)	Monitoring of heart rate and heart rate variations	Continuous or Event	High /Variable
Fall detection	Detection and reporting of a fall	Event	Low
Relay node	Multi-hop communication	None	Variable

2) Sensor node software: TinyOS

Many of the existing implementations of BANs have been based on tailor-made software that fits the purpose of the application. Most of these software implementations are proprietary, lacking flexibility and openness, and customers are often faced with vendor lock-in and high total cost of ownership. With the emergence of small operating systems such as TinyOS and Contiki a gap between the application and the underlying hardware, available in the sensor world, have been closed. Hence application developers or no longer forced to account for all the lower level details when developing applications.

TinyOS is an open source operating system designed for low-power wireless devices, such as those used in sensor networks and personal area networks [65][66]. The operating system and its associated tools are supported by a worldwide community that counts people from academia as well as the industry.

ASE-BAN sensor devices were based on TinyOS. This gives the following immediate benefits:

- Hardware abstraction layer
- Hardware platform support (MSP430 and CC2420)
- Core operating system functions e.g., memory management, interrupts handling, timer etc.
- An event-driven concurrency model for program execution
- Driver support e.g., for the radio hardware unit
- Networking protocols
- Software development tools
- No license cost (BSD-licensed)

D. Wireless node hardware platform

The hardware architecture reflects the different stakeholders of the platform: sensor, communication, power and embedded processing specialists. The basic elements of the architecture are shown in Figure 5.

Energy, as energy sourcing and power conditioning; *Communication* that implements secondary communication technologies; *Physical IO* containing sensors, actuators and pre- and post-data processing and finally *Processing Element*, for managing the system and optional data processing. The processing element component can also include the primary radio frequency (RF) transceiver that is often integrated with the main CPU. Each wireless sensor node is a mix of these building blocks.

1) Processing Element.

Many of today's wireless sensor node platforms are based on TinyOS and AVR/MSP430 processors. Several ARM Cortex-M0/M3 devices are emerging [67], allowing the nodes to benefit from the 32-bit architecture, thus enabling more processing power in the nodes, which again enables new methods in data aggregation and compression [68].

A comparison of some processors with build-in RF is shown in Table II. Energy consumption is given for the wake-up, the active, the sleep and the transmission and reception phase (*Tx/Rx*) of the node operation. Energy consumption estimations are based on current consumption and wake up time values from the respective data sheets.

The ARM and the AVR processors use 32-/8-bit RISC architectures respectively, whereas the MSP430 uses a 16-bit Von Neumann architecture. The AVR and MSP430 processors are assumed on average to take three clock cycles per normalized instruction compared to an ARM Cortex-M3.

Table II shows how the different CPUs have different strength and weaknesses. For computation intensive applications, the ARM Cortex-M3 CPU is preferred, whereas communication intensive applications would benefit more from the efficient RF front-end of the AVR processor. Applications that wake up regularly, but transmit less data, will benefit from the MSP430's short wake-up time. Hence, the choice of processor depends on the application used.

2) Communication

The primary wireless connection used for the wireless BAN is application dependent as well. Wireless sensor node research extends well beyond near-distance, inter-body wireless communication. Future applications may well extend to agricultural, environmental, and energy surveillance applications. Table III shows several wireless standards which the platform should be prepared to accommodate for.

3) Energy

Being open to different applications also effects how energy should be sourced and conditioned. Rechargeable batteries require a charging circuit, and even during shut

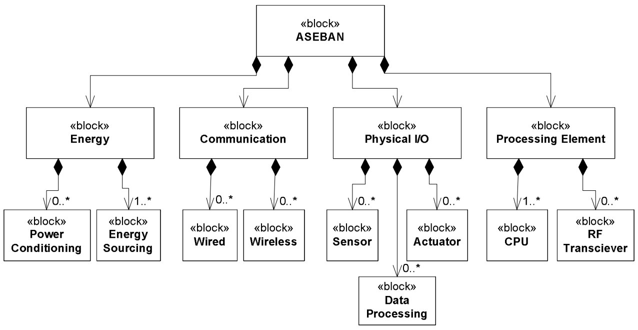


Figure 5. ASE-BAN node hardware block diagram.

TABLE II. ENERGY CONSUMPTION COMPARISON.

Processor	**Energy Consumption** (1.8 volt)			
	Wake-up [nJ]	***Active*** [nJ/instr.]	***Sleep*** [nJ/s]	***Tx/Rx*** [nJ/bit]
ATmega128RF1 (AVR) [69]	147	0.620	1800 (PDX)	100
CC430F6126 (MSP430) [70]	14	0.860	3600 (LPM3)	150
EM357 (ARM, Cortex-M3) [71]	1188	0.460	1440 (DS1)	200

TABLE III. COMPARISON OF WIRELESS TECHNOLOGIES.

	Z-Wave	IEEE	Dash-7	ANT	Bluetooth Low E
Standard	Proprietary	IEEE 802.15.4	ISO 18000-7	Proprietary	IEEE 802.15.1
Target	Home automation	Health-care	Military, Industry	Health-care	Health-care
Frequency	900 MHz	2.4 GHz	433 MHz	2.4 GHz	2.4 GHz
Topology	Mesh	Mesh, Star	Mesh	Mesh, Star, P2P	Star, P2P
Range	30M	30M	1000M	30M	1M
Data Rate	40 kb/s	250 kb/s	200 kb/s	20 kb/s	200 kb/s

down its internal transistors (FETs) will drain a small quiescent current. A coin-cell battery provides a higher capacity and a lower self-discharge than rechargeable battery. For applications that require medium power and have short operational life, a rechargeable battery is preferred. This is illustrated in the following example: A sensor node based on the EM357 samples an analogue value, preprocesses the sample, receives and transmits an IEEE 802.15.4 packet and goes back to deep sleep at a fixed interval. Figure 6 illustrates the achievable battery life with different wake-up periods.

For applications running up till a month, the rechargeable battery will be the right choice. For applications that have low activity, but a long lifespan, the coin cell battery (e.g., CR2032) is the preferred choice. The ASE-BAN platform is prepared for both.

Secondary communication forms include Universal Serial Bus (USB) for debugging and the base station applications as well as different wireless interfaces such as Bluetooth for body gateway applications.

4) Physical I/O

The Physical I/O should be able to interface to most sensors and actuators. This interface should be kept open as sensor technology is a key design requirement. A list of the required interfaces for current sensors is given in Table IV. As the table indicates, GPIO, SPI, I2C and analogue interfaces will be sufficient. Interfaces such as USB host or LVDS does not match the requirements of a low-power wireless node. A DSP is included for computation intense pre- and post-data processing.

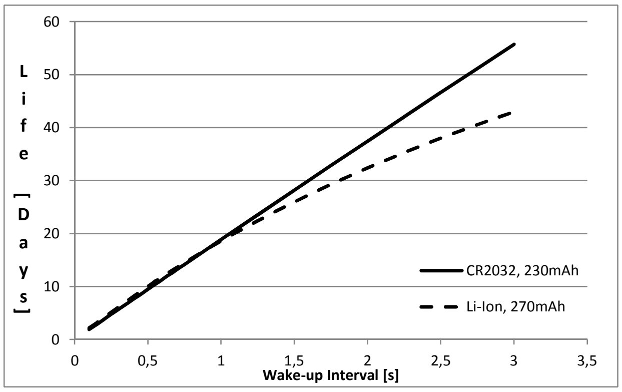


Figure 6. Example of battery life vs. wake-up interval.

TABLE IV. SENSOR CHARACTERISTICS AND INTERFACES.

Application	Digital I/F	Analog I/F	Inter-rupt	Supply	Example
Acceleration	SPI	x3	Yes	1.8-3.3V	ADXL345
Temperature	I2C	x1	Yes/No	2.7-5.5V	AD7414
Microphone	No	x1	Yes		
Codec	I2C/SPI	No	No	5V	AD1877
Impedance	I2C	No	No	2.7-5.5V	AD5933
Gyro	SPI	x1	No	5V	ADXR520
Digital Potentiometer	I2C	No	No	2.7-5.5V	AD5175
Strain Gauge	No	x2	No	2.7-5.5V	
DSP	SPI/I2C/TDM	No	Yes	1.2+3.3V	BF533
I/O Extender	I2C/SPI	No	Yes	3.3V	
H-Bridge	4xGPIO	No	No	3.3V	

The mechanical properties of the nodes are important as well. In order to make the nodes usable in real applications, they should be compact and have a self-supporting ruggedized structure. These properties are not found in many research nodes created so far.

V. IMPLEMENTATION OF ASE-BAN

This section describes the implementation of the ASE-BAN platform covering hardware, software, protocols, body gateway and the different sensors.

A. *Hardware platform*

The ASE-BAN hardware platform shown in Figure 7 is built from hardware modules attached to a common base Printed Circuit Board (PCB) as illustrated in Figure 8 and pictured in Figure 9.

The modules are soldered together with the base PCB by

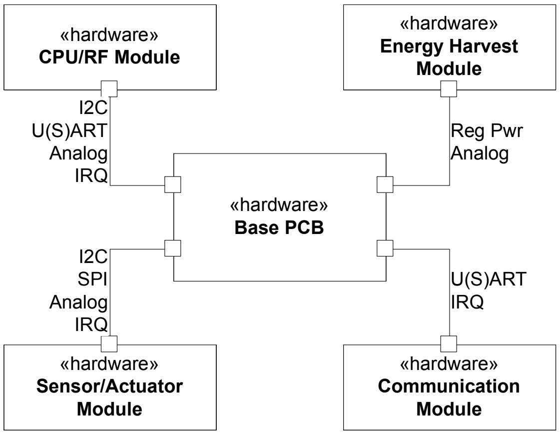


Figure 7. ASE-BAN's modular node design

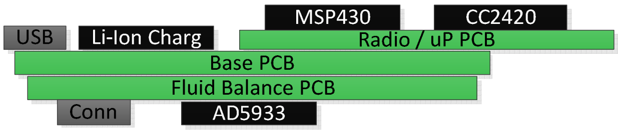


Figure 8. ASE-BANs physical layering with the fluid balance sensor stacked to the base PCB.

using edge-plating technique on the modules. This technique is well-known from Bluetooth modules. Using this approach is quite beneficial, since the design becomes very modular and the structure becomes quite robust and compact as no connectors are involved. The edge plating technique allows easy assembly and the construction of new modules. This is in contrast to use of fine pitch connectors or BGA techniques.

The modules support the architecture described earlier. The base PCB acts as a passive backplane, processing is placed in the CPU/RF module, sensors on the sensor module and so on. This allows us to select among a variety of communication forms, sensors and energy sources, while maintaining the basic functions of the system.

The base PCB includes sensors such as, temperature, 3D acceleration and proximity. These sensors are placed on the PCB backside and may be replaced by a sensor module with other sensors attached. It also has a build-in Li-Ion charger, power conditioning, debugging LEDs, buttons and a micro USB connector for charging and possible wired communication.

Similar design techniques exist for the energy sourcing and conditioning circuits. They may e.g., also be replaced by an energy harvesting module or a CR2032 battery. The solder terminals are placed along the PCB edge allowing the sensor modules only to use a fraction of the full base PCB length. Connections such as power and serial IO are duplicated to support this.

An example of a sensor node is the fluid balance node shown in Figure 10. Through impedance measurements in skin tissue the node estimates the current fluid balance. The battery chosen is a 270 mAh Li-Polymer battery. The fluid balance sensor consumes a lot of energy, and the sensor is not intended to be fitted for long periods of time without

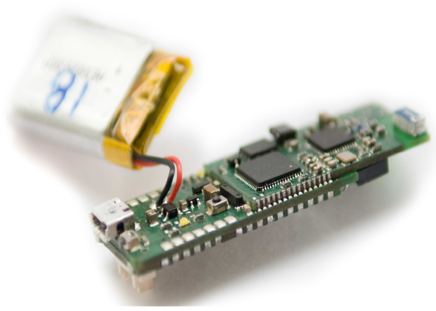


Figure 9. ASE-BAN node (processing and energy side).

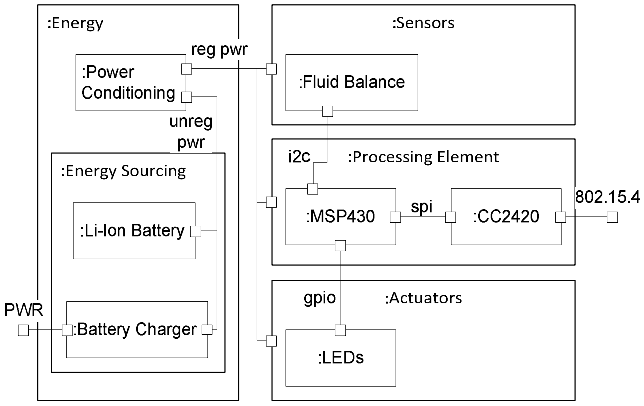


Figure 10. Fluid balance sensor node internal block diagram.

service. The application allows us to recharge the node when the user is in bed at night. The charging circuit takes power from the micro-USB connector and a matching power adapter should be a commodity in today's smartphone households.

Power conditioning is done by means of LDO regulators, as the quiescent current of switched-mode regulators becomes dominant over the improved power conversion efficiency.

As the processing element, the MSP430 processor [72] from Texas Instruments was chosen in conjunction with the CC2420 radio transceiver [73] also from Texas Instruments. This is a rather common set-up seen in sensor nodes such as the TelosB [74]. This setup allows easier integration with TinyOS, as this is the target operating system. As mentioned in the architecture section, the MSP430 provides overall good performance for medium processing applications, so it fits this application quite well. The CC2420 is a well-established IEEE 802.15.4 radio transceiver used in related research [75][76]. Actuators are LEDs for this application, but buzzers and vibrators are being considered for stand-alone fluid balance applications.

The fluid balance sensor itself, shown in Figure 11, is based on an impedance measurement integrated circuit from Analog Devices. The primary interface to the processor is I2C used for sending commands and receiving measurements. The sensor module has a local LDO regulator that can be shut down by the processor to minimize power consumption during node sleeping periods. The fluid balance sensor is described in more detail in a later section.

The fluid balance sensor node has a small connector for attaching external electrodes. It is 8x20x55 mm in size and weights 9 grams, excluding battery. This allows the node to be placed in a wristband for realistic application evaluation.

The current setup enables the creation of sensor nodes for a wide range of applications: Temperature sensors that wake up every minute but has years of service time, thus requiring a coin cell battery and a very basic CPU/RF module. HRV sensors are very processing intensive using an additional DSP for HRV estimation, have only 2-3 days operating time and require rechargeable Li-Ion batteries.

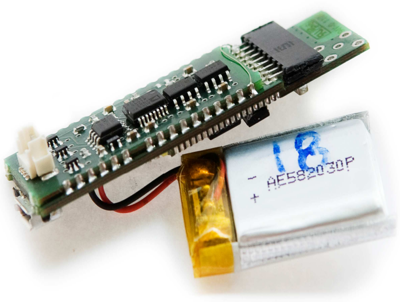


Figure 11. ASE-BAN node (fluid balance sensor side).

The hardware construction has proven to be convenient and robust obtained by the close attachment of the PCBs. The hardware design accomplishes the goals for the ASE-BAN platform, namely to provide a flexible platform for interdisciplinary research and development of sensors for a wide range of healthcare applications.

B. *Software and protocols*

The network interface abstraction that comes with TinyOS provides a generic way to use the network regardless of the underlying hardware instance.

The core TinyOS communication abstraction is based on Active Messages. Active Messages provides an unreliable, single-hop datagram protocol, and provides a unified communication interface to both the radio and the built-in serial port.

More recently TinyOS has been extended with a IPv6 protocol stack, called the Berkeley Low power IP (BLIP) stack [66]. The BLIP protocol stack is adapted to the IEEE 802.15.4 radios and has been optimized to run on sensor nodes with limited resource characteristics [77]. This implementation of IPv6 over IEEE 802.15.4 communication, commonly known as the 6LoWPAN protocol stack, has been standardized by the Internet Engineering Task Force (IETF). This provides an Application Program Interface (API) for communication that is then implemented for the particular interface(s) of the sensor device. Furthermore, one can seamlessly connect sensor devices to the Internet or to networks based on Internet technology such as e.g., the OpenCare project [2].

1) *Protocols*

In ASE-BAN experiments with the usage of Active Messages as well as BLIP have been made. Figure 12 shows an example of the protocol stacks for an end-to-end system based on 6LoWPAN. Looking at communication end-to-end one finds IPv6 as the common denominator that connects the home base station (or a remote central server accessible over the Internet).

The use of IPv6 at the network layer allows us to connect sensor devices to the Internet and to support multi-hop communication based on standardized, light-weighted routing protocols [78].

To bridge the gap between the healthcare sensor application and the communication interface, it was decided

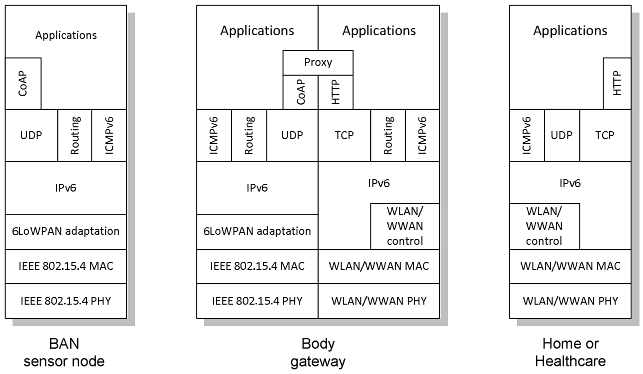


Figure 12. ASE-BAN protocol stack.

to rely on the UDP socket interface provided by the BLIP protocol stack or to rely on embedded web services [79]. In the latter case a middleware layer, which is able to map applications' requirements to the sensor network resources, was used.

These server controlled resources are accessed by clients in a synchronous request/response fashion using methods such as GET, PUT, POST, and DELETE of HTTP/CoAP, as shown in Figure 13.

IETF is proposing the Constrained Application Protocol (CoAP) to support RESTful web services in constrained environments such as wireless sensor networks. In ASE-BAN we have chosen to follow this path and have implemented a TinyOS implementation of CoAP similar to [80] .

Hence, healthcare services can be provided end-to-end with open, flexible and scalable software which eventually leads to an attractive total cost of ownership for the healthcare provider, i.e., often the society in general.

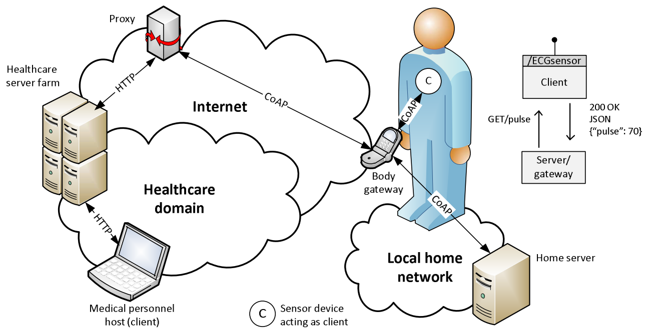


Figure 13. Example of web service architecture for interoperable healthcare services for ASE-BAN.

C. Body gateway and node software components

1) Smartphone as a body gateway

The body gateway was developed using a smartphone (a Google Nexus One), but was also imagined as an embedded solution. The gateway is responsible for relaying information from the internal wireless BAN to an external IP network. A smartphone based on the Android operating system was chosen as the target platform [81][82], since it allowed for easy prototyping using high-level code.

The first prototype of the body gateway used a Bluetooth Serial Port Profile (SPP) to communicate with the bridge node of the wireless BAN. This is not an optional solution as the bridge node requires daily recharging caused by the power demand of the Bluetooth protocol.

The second prototype still needed the bridge node, but used a directly wired serial port. This was possible due to the openness of the Android platform, which allowed compiling and replacing the driver module for the USB connector on the phone. This reduced the power consumption and minimized the footprint. A third option was considered, where the master node is connected to the same battery as the phone and directly inserted into the phone via the microSD card interface.

2) System Software Components and communication

The actual ASE-BAN consists of 4 different types of nodes with software: a central server, a body gateway (the smartphone), a bridge node and the sensor nodes.

The central server and the body gateway software components shown in Figure 14 use essentially the same code to handle incoming events and differ only in the way they receive these events.

The central server receives incoming events via the HTTP protocol from the body gateway, while the body gateway receives these using a serial port connected to the bridge node.

The bridge node software component shown in Figure 15 is designed to bridge data from the wireless BAN to the serial port. The wireless BAN software uses the Active Message protocol, where data is delivered as datagrams, while the serial port uses a simple frame stuffing algorithm to isolate each datagram.

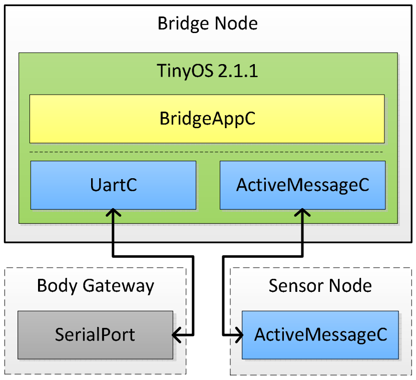


Figure 15. Bridge node software components

Figure 16 shows the sensor node software components for a sensor node equipped with a fluid balance sensor and an on-board accelerometer. The SoftI2cC component implements the I2C communication for two general purpose I/O pins. The accelerometer component is the software driver for the physical accelerometer which communicates through the I2C interface. The FallDetectionC component implements a fall detection algorithm. The SensorAppC component binds the software components together and delegates the received events e.g., a fall event and a fluid balance alarm event to the ActiveMessageC component, which implement the Active Message protocol.

D. Sensors

1) Fluid Balance sensor

The fluid balance sensor enables wireless BAN to issue alarms when the body fluid level of the user becomes critically low. The fluid balance sensor is designed for measurement of the electrical-biological impedance (EBI) of adult humans. The focus is on the detection of the dehydration, typical for elderly or demented people, who may benefit from some kind of feedback when water intake is needed. However, the sensor may also be implemented

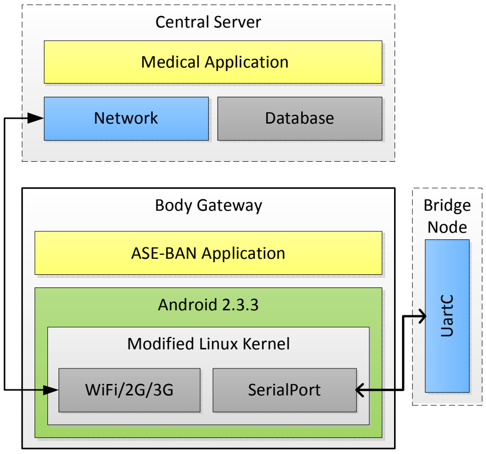


Figure 14. Central server and body gateway software components.

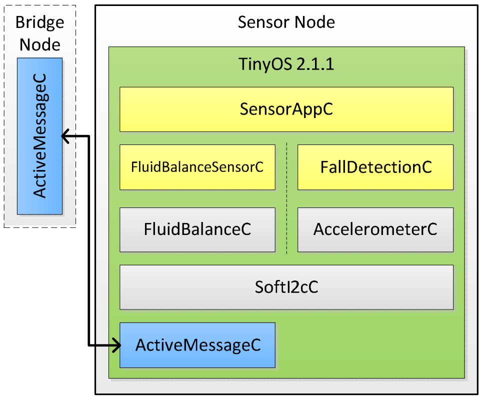


Figure 16. Sensor node software components.

within areas such as total body composition, lungs composition and respiration rate if the software is adjusted accordingly [83].

Biological tissues are composed of groups of cells with each cell consisting of a cell membrane separating the internal fluid from the external fluid. The conductivity of the fluid may be represented through a resistance between any two points and the impedance of the tissue becomes modeled as the resistance of the external fluid in parallel with the resistance of the internal fluid and separated by the capacitance of the cell membrane, thus leading to equivalent models by Fricke (1924) and Cole (1928). They are commonly called “2R-1C” networks since they consist of two resistance values and one capacitance shown in Figure 17 [83]. Typical resistance levels are from 10 Ω to 200 Ω and the average capacitance value is within the 100 nF range depending on the measurement principle being used, such as from arm to arm or from arm to leg. The impedance is within the ASE-BAN circuit monitored from 5 kHz to 100 kHz.

The present design uses the AD5933 impedance measurement device from Analog Devices, which comprises a programmable sine wave oscillator and discrete single-frequency Fourier transform circuitry converting the measured quantity into real and imaginary values representing the complex impedance. External circuitry is used to convert the AD5933 oscillator output voltage into current, which is routed through the skin to the tissue to be analyzed using two electrodes. Two additional electrodes are used to monitor the voltage created across the tissue and this voltage is correlated with the excitation signal thus determining the relative magnitude and phase of the impedance. Calibration uses a fixed resistor for scaling of the numerical value output from the AD5933 back into resistance thus reducing errors related to component tolerances.

Fluid balance measurements must be determined with approximately 1 Ω of accuracy for determination of the change in impedance due to dehydration. The contact resistance from electrode to skin may exceed 1 kΩ at the frequency range of interest. Four electrodes are needed to unload the measurement electrodes from the excitation current. The present design uses an AC current level of 40 μA, which is far below the detectable range [84]. A plot of the impedance in the complex plane approximates a half circle in the fourth quadrant, usually called a Cole plot, is shown in Figure 18. Impedance measurement is conducted by fitting the measured values with the periphery of a half circle, and then extrapolating the half circle to zero and infinite frequency for determination of the resistance of the external and internal fluids as well as the bulk capacitance of the cell membrane. At infinite frequency the impedance reduces to the series resistance of the Cole model and at zero frequency the impedance becomes the sum of the resistance components. The capacitance value is determined from the frequency at the extreme imaginary value. The degree of dehydration is detected as a change within the resistance pattern, and examination is currently being conducted to determine this correlation, the effect of electrode contamination and the required monitoring precision.

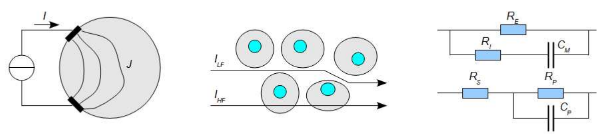


Figure 17. The electrical-biological impedance measurement principle is shown with the current path at low and high frequency (LF and HF respectively). To the right the Fricke model (top) and Cole model (bottom).

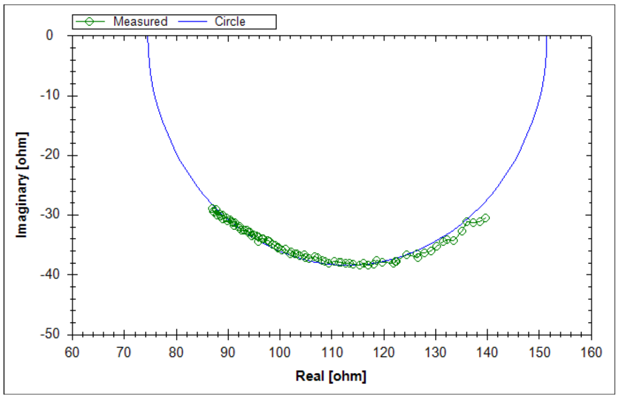


Figure 18. Cole plot of a network with two series-connected resistors of 75 Ω each and one capacitor of 220 nF in parallel to one of the resistors with the approximating circle. The frequency range was from 5 kHz (rightmost data point) up to 25 kHz using 250 Hz of step size.

A result is shown within the picture using a 2R-1C model network. The analysis is to be carried out autonomously by the fluid balance sensor and the result is either uplinked using the radio transmitter or alternatively output directly to the user as an audible or visible indication of the need to drink water.

2) ECG sensor

Monitoring of heart activity uses an analogue interface with an instrumentation amplifier. The signal from the heart is less than 5 mV of peak amplitude so the interface amplifies the signal to fit the input range of the A/D converter. The input is differential to reduce mains hum and a third electrode is used for suppression of common-mode disturbance. The required bandwidth is 40 Hz for heart rate determination but sampling at 500 times per second resolves frequencies up to 150 Hz for diagnostic use. Digital filtering may implement a filter for additional hum suppression, such as notch filtering at the frequency of the disturbance without serious impact upon bandwidth or low pass filtering with high order slope and cut off around 40 Hz.

The interface is general-purpose and may be used for any low-voltage analogue interfacing with bandwidth set by the sampling rate at the A/D converter.

A more advanced interface is offered through the DSP based ECG sensor, which is interfaced to the ASE-BAN PCB base module and is intended to unload the microcomputer and radio link through the use of a dedicated signal processor device.

From the electrical signal, the HRV can be derived [85]. The complete electric signal: the electrocardiogram (ECG) and the HRV is measured by the sensor. The ECG and the HRV are widely used for medical surveillance and diagnostic purposes.

The ECG sensor module measures two lead ECG signals on the chest of the user. The sensor is intended to be worn for several days without intervention; hence appropriate electrodes are used, to provide for high signal quality and user comfort. The prototype uses insulated bioelectrodes which provides good signal quality and reduced risk for skin irritation [86]. The module amplifies the weak ECG signal (peak to peak amplitude of approximately 2 mV), before the signal is AD-converted. The sampling rate is 500 Hz in 16 bits to provide for sufficient signal quality for diagnostic purposes [87]. Figure 19 shows a recorded ECG on the home base station.

The processor module shown in Figure 20 for the ECG sensor is a small foot-print DSP platform equipped with a Blackfin BF533 signal processor from Analog Devices [88]. The Blackfin BF533 is a high-performance fix-point processor with two 16-bits Multiply-And-Accumulate units, capable of parallel processing. The processor is capable of handle clock-speeds up to 600 MHz. The on-chip Real-Time-Clock is connected to a 32 kHz crystal.

The DSP processor module is used for sensor local analysis of the ECG signal. This is done to save node energy, since processing data is more energy efficient that transmission of data. Commercial radios typically dissipate *Te* ~150 nJ/bit [89]-[91], versus the processor referenced in Table II dissipate on the order of *Ce* ~1 nJ/sample [92]. This indicates a break-even between transmission and processing of data at ~2500 instructions per sample (16 times 150 inst./sample) [93].

Equations for the energy budget can be setup as follows:

$$K \cdot Ce + n \cdot Te = N \cdot Te, \qquad (1)$$

K is the number of instructions needed to reduce *N* samples to *n* samples. *Ce* is the energy-use per instruction and *Te* is the energy-use per transmit of one sample. Rearranging Equation (1), one arrives at Equation (2):

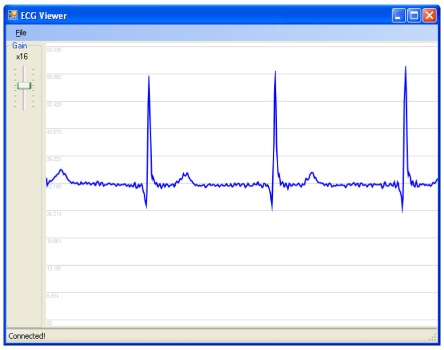


Figure 19. ASE-BAN measurement of an ECG signal.

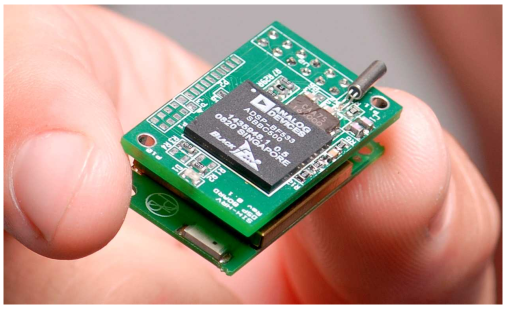


Figure 20. The ASE-BAN ECG sensor module with DSP. The size is 13 mm x 18 mm x 30 mm. The weight is 6 g.

$$\frac{K}{N} = \left(1 - \frac{n}{N}\right) \cdot \left(\frac{Te}{Ce}\right), \qquad (2)$$

Our application aims at reducing bandwidth by a factor n/N ~1000 times (HRV transmission at 0.5 Hz, and ECG sampling rate at 500 Hz). According to the analysis above *Te/Ce* is set to 2500 inst./sample and hence K/N attain a value of ~2500 inst./samples.

ASE-BAN software runs standard adaptive noise removing techniques to remove hum in the ECG. The R-peak in the ECG signal is calculated using the Pan Thomkins algorithm [56] and the ECG signals is finally analysed, calculating standard Heart Rate Variability pNN50 [85]. This algorithmic data reduction can be run on the processor module. An efficient implementation is estimated to be using 500 inst./sample or less. This is significantly below the 2500 inst./sample break-even limit calculated above.

3) HRV sensor

A dedicated module for high-speed processing of the data is offered using the BF533 digital signal processing device of the BlackFin-series from Analog Devices. The module was developed for evaluation of algorithms and interfaces to the ASE-BAN module.

4) Accelerometer sensor

Acceleration detection is a versatile instrument with applications for user-orientation (standing or lying), fall detection, and alarm generation for users that do not move at all following a fall, in addition to motion and tap detection. The project interfaces to LIS331 or MMA8452Q, which are tree-axis seismic accelerometers interfacing through I2C. A mass is part of a capacitive half bridge for each of the axis within the device, so the sensor is capable of detecting static gravitation such as monitoring the physical orientation along earth's gravity as well as detection of shock and vibration. The sensors may generate an interrupt request to the micro controller if the acceleration crosses through programmed limits.

5) Fall detector

For use in the demonstrators a fall detector application was designed. The fall detection algorithm is based on the movement pattern during a fall. When a person falls either from a standing or sitting position or when walking the sensor will experience the same basic pattern. First, the sensor will sense close to 1 g of downwards force due to gravity. Then, during the first phase of the fall, a free or near free fall condition will be experienced. The free fall condition will be followed by the impact phase in which the person will hit the ground. The sensor will see this as a series of large spikes in the sensed acceleration. This detection state machine is shown in Figure 21.

The algorithm could be improved to track the fall after the impact to determine if the person is unconscious or is trying to get back up. The fall detector is currently able to detect the laboratory reproductions of a fall but optimizations such as lowering the 50 Hz sample rate to reduce energy consumption or adjusting acceleration and timing thresholds to increase detection reliability has not been performed.

6) Proximity detector

Applications include the substitution of mechanical switches and proximity detection, such as the presence of a user of the ASE-BAN circuitry. The interface uses the AT42QT1010 chip, which detects the change within capacitance due to the presence or absence of the human body or one of the fingers within the proximity of a conductive plate. The interface is one bit so the output is detected or not detected, and the device includes options for reducing sensitivity to electrical noise or transients from quick brushes with an object, such as during cleaning.

7) Temperature sensor

The project includes several temperature monitoring points due to the build-in sensors at the micro controller and the fluid balance sensor, but they are not externally accessible and the measurement precision is limited.

VI. DEMONSTARTOR AND RESULTS

This section describes the ASE-BAN demonstrator and the results. Since this is work in progress only preliminary results will be presented together with a prototype design of a casing for an ASE-BAN sensor node.

A. Demonstrator

A fully integrated ASE-BAN demonstrator, to be used by a test person, is under preparation. The objective of this demonstrator is to show the integration of the developed ASE-BAN components as well as the feasibility of integration into a complete healthcare system.

From a sensor point of view the demonstrator consists of fluid balance sensor nodes, an ECG sensor node, an ambient temperature sensor node and a fall detection sensor node. All sensor nodes except the fall detection sensor are installed on the front of the test person. The fall detection sensor device is installed on the back at the person's waist. To provide connectivity to sensors on the back an ASE-BAN relay node is installed. Figure 22 shows the positioning of sensor nodes and how they are connected in the network.

To demonstrate the feasibility of the integration with a healthcare system it is shown how sensor nodes can be monitored remotely via the Internet as well as from the OpenCare project infrastructure installed in the user's home.

The two fluid balance sensor nodes – one attached to the upper arm and the other attached to the person's thigh are used in order to test the consistency of sensor readings.

In the demonstrator an Android smartphone is used to act as ASE-BAN body gateway. The smartphone connects to the sensor nodes by using one of the sensor nodes e.g., the ambient temperature sensor node as a bridge node. A serial connection between the smartphone and the bridge node is configured. Both a wired and a wireless option are viable. In the former case the bridge node is piggy-backed onto the smartphone whereas in the latter case a serial Bluetooth connection is used between the bridge node and the body gateway.

The demonstrator is able to push data to the home base station and/or an external web server residing in the healthcare domain, i.e., a central server. In the latter case a global IPv6 network infrastructure with the 6LoWPAN networking capabilities of the BAN is established. For this part of the demonstrator a Linux PC is used as a 6LoWPAN due to the lack of support for IPv6 in Android. Effort is ongoing to include IPv6 support and to port the 6LoWPAN edge router software to Android. The results with 6LoWPAN

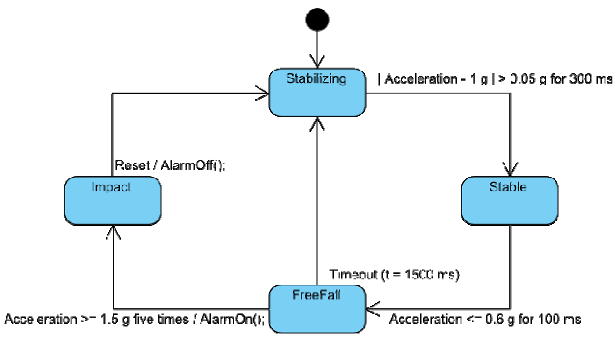


Figure 21. The four states of the fall detection algorithm.

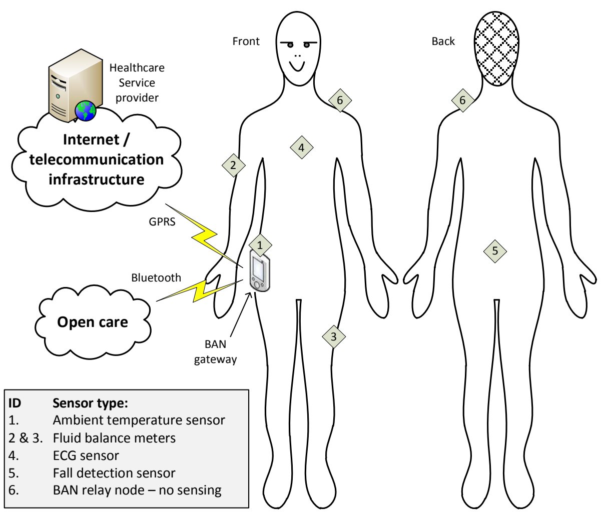


Figure 22. BAN with four sensor devices connected to a global network infrastructure.

are described as Result 2. Another experiment with the demonstrator used the smartphone with the Active Messages protocol as described in Results 3. In both cases data can be displayed on the smartphone as well as on a remote central server.

B. Results

1) Result 1– Radio communication channel

The human body has a significant impact on the radio communication in a wireless BAN. To gain a better understanding of this effect a dedicated test environment has been created for this. This consists of a boiler suit fitted with an array of IEEE 802.15.4 compatible radio modules. These radio modules are connected to a PC through a cable to form a wired network for test purpose. This setup is illustrated in Figure 23 which also shows the actual boiler suit.

The radio modules continuously broadcast packets containing their network id. These packets are received by some or all of the other radio modules depending on the radio conditions. As all modules transmit at the same power level each receiving module is able to calculate the loss in signal strength by subtracting the Received Signal Strength Indicator (RSSI) value recorded when the packet was received from the known transmission power level. The cost of all links is transferred to the PC through the wired network and stored in a log file. Simultaneously a camera connected to the PC records the actions of the wearer of the boiler suit.

These images are stored in the log file together with the link quality data from the same instance in time that the image was taken.

By creating log files of standard everyday scenarios (sitting down on a chair and getting back up, walking etc.) the expected signal conditions in an actual wireless BAN were identified. It was evident from the tests that the communication in a wireless BAN often relies on reflections from the surroundings for the radio waves to reach their destination as there is often no line of sight path. Areas that should receive special attentions were also identified. The effect of a swinging arm during a walk was for instance very significant between certain nodes but barely visible on other links. Calculations indicate that packet loss can be reduced significantly and energy consumption lowered due to improved link quality if automatic power control is employed. The information obtained with these tests where used to design the network for the demonstrator.

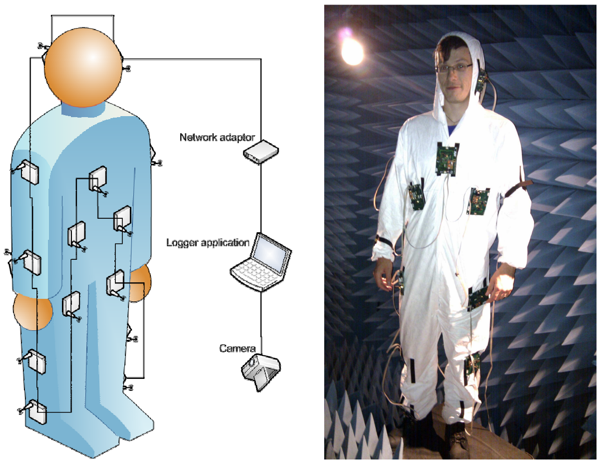


Figure 23. The test environment (left) and the actual boiler suit (right).

For a detailed explanation of the full test environment and a deeper analysis of the results please refer to [94].

2) Result 2 – 6LoWPAN network

This part of the demonstration focus on body area networking aspects and global IPv6 Internet connectivity. Our experimental setup consists of up to 6 sensor nodes and a base station. In order to demonstrate interoperability a mix of ASE-BAN nodes and the TelosB nodes, that offer similar hardware architecture, have been used [74]. For the base station a Linux PC with a TelosB node for the IEEE 802.15.4 connectivity has been used. The Linux PC is configured as a 6LoWPAN edge router.

The ASE-BAN nodes implements the 6LoWPAN protocol standards based on the open source TinyOS stack called BLIP. Global IPv6 connectivity will be provided by using an IPv6 deployment and tunnel broker service provider such as SixXS [95]. Over this global IPv6 infrastructure sensor data is delivered by using web services.

The ASE-BAN testbed successfully demonstrates many networking aspects that are of importance and relevance for body area networking. The following list provides a few highlights of this demonstrator:

- Sensor device network interoperability (TelosB and ASE-BAN hardware)
- Multi-hop communication in the BAN (mesh-networking)
- 6LoWPAN networking
- Dynamic routing by using Hydro routing protocol
- Global IPv6 connectivity using a tunnel broker
- Web services for BAN

3) Result 3 – ASE-BAN applications

The demonstrator contains software applications that were developed to handle issues from different parts of the ASE-BAN. In this section a fall detection application based on accelerometer readings will be described.

A central issue was how to connect the smartphone to the actual wireless BAN, originally the solutions was to use a Bluetooth connection, since this seemed to be the easiest and most compatible solution. To gain battery life and a smaller footprint solutions that allowed using the USB connector of Android as a serial port, as shown in Figure 24, were implemented.

For the gateway a Google Nexus One (HTC) Android smartphone with CyanogenMod 7.0.2.1 was used [96]. CyanogenMod is a customized, aftermarket firmware based on Android 2.3.3. The platform allowed using the USB connector as a serial port.

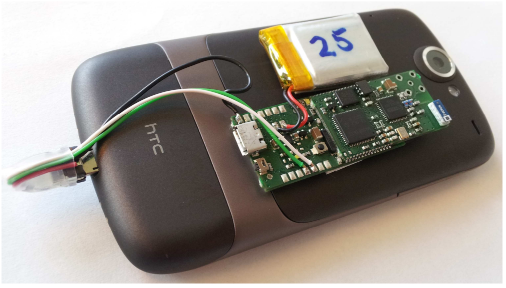


Figure 24. Body gateway with bridge node.

Figure 26. Body gateway accelerometer curve.

ASE-BAN needed to be modular, and support sensor types which are not defined yet. Therefore a highly flexible way to handle data transport and data presentation was designed. For data transport Java Script Object Notation (JSON) was chosen [97]. JSON is a lightweight alternative to XML, and is native to the JavaScript language.

First data is collected on the fall detector node shown in Figure 25, #1 and is transferred to the body gateway. The actual data is formatted as JSON.

Data example from the fall detector node:

```
{"d":0.156,"f":"fd"}.
```

This is a simple JSON byte array with two fields: “`d`” is data as a JSON Object, and “`f`” is a unique data format identifier. In this case the format identifier “`fd`” is used, which tells us that the data is a fall detection format and that the data is in G (gravity).

Next data is received at the body gateway, where data will be relayed to the Central Server, Figure 25, #2:

```
{"d":0.156,"f":"fb","t":69585742574,
 "u":"Foo Bar","s":"Accelerometer"}.
```

Here additional fields are added, “`t`” is a timestamp in Unix time. “`u`” is the unique name for the BAN user, “`s`” is the unique sensor the data originated from.

The last step is when data enters the database and is stored for later usage, Figure 25,# 3.

For data presentation it was decided to use a combination of JavaScript and HTML5. This allowed us to reuse UI code for both the Android smartphone and the central server.

An example is shown in Figure 26 with accelerometer data displayed on the Android smartphone while Figure 27 shows the same data presented in a browser that accesses the central server.

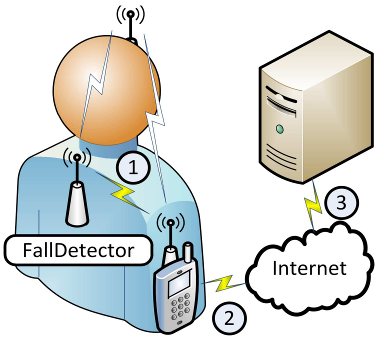


Figure 25. Communication flow.

C. *Prototype casing for the ASE-BAN module*

To bring the ASE-BAN module out of the laboratory and into the hands of the user group a casing has been designed by an industrial designer. The design is based on the fall detection application but is open enough to be adapted to any application supported by the ASE-BAN platform. Figure 28 shows the casing being used by a potential user.

The black circle in the middle is a button for user while the green circle is a multi-color LED used to inform the user of system events. The back side features a secondary button used to test remaining battery capacity. When this button is pressed the LED will light up in green, orange or red depending on battery status. The bottom of the casing exposes two metal pads which mates with two matching pads when inserted into the charger as illustrated in Figure 29.

The driving force behind the design has been to create something that the user would naturally embrace and think of as a decorative object. Both the center piece of the casing and the two “wings” come in different colors to allow users to make their individual unit unique. Furthermore, the design

Figure 27. Central server accelerometer curve.

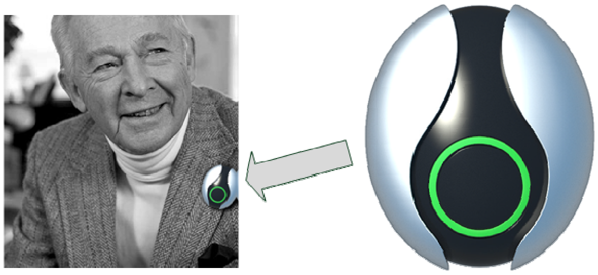


Figure 28. The ASE-BAN casing being worn by a user.

of the backside of the casing makes it possible to wear it either as a broche or as a necklace.

VII. Discussion and Future Work

With additional personalized healthcare assistance a more comprehensive and affordable healthcare solution is provided to the BAN user. It is absolute vital that the solution is highly reliable and easy to configure and operate. Furthermore, it must be energy-efficient to ensure a long battery life-time.

The next generation of wireless technologies is being driven by the rapid convergence of three key technologies: MicroElectroMechanical systems (MEMS), digital circuitry, and the explosive growth of wireless communications. Common to all three are reductions in size, weight, power consumption, and cost associated with the large number of units produced, as well as reductions in complexity and functionality.

In sensor networks, energy consumption is of highest priority and the RF communication design blocks consume the most energy. Wireless sensor network designers strive to reduce the power consumption of the blocks in general. The improvement for wireless sensor network is likely to be used to reduce size and power consumptions instead of increasing capacity and speed. The use of energy harvesting is an important aspect of sensor devices. With a smart combination of energy efficient protocols and energy harvesting methods, the optimal solution for achieving autonomous and long-lasting BANs can be reached.

Efficient protocol support is also needed for the IP based wireless sensor networks and the ongoing work of the IETF 6LoWPAN working group is heading in this direction. This includes protocol optimization for small devices such as neighbor discovery, compression mechanisms for TCP, light-weighted key management protocols as well as energy efficient routing protocols.

Figure 29. The ASE-BAN casing and its charger.

In essence, the implemented, modular wireless BAN testbed is flexible and can be customized to the individual needs of users. Our hardware platform is energy-efficient and has a low footprint. Whilst the basic capabilities of the testbed have been demonstrated there is still effort to be done in particular with respect to the software and protocol parts. As an example a plan exists to adapt the ASE-BAN testbed to the Medical / health device communication standards, i.e., the IEEE 11073 standards [51]. The IEEE 11073 standards specify the communication between medical/healthcare devices and external computer systems in a client-server architecture. Features such as automatic and detailed electronic data capture of client-related and vital signs information as well as device operational data can be communicated, from the BAN node application to servers residing at e.g., a hospital, over an IP network. Standards like IEEE 11073 are critically important to ensure multi-vendor interoperability thus enabling the personal healthcare to converge from today's defragmented market where isolated solutions exist.

The long term plan is to design a low-cost plug-and-play biomedical wearable computing network that can be integrated as part of a future ambient assisted living network, to be used e.g., for local personal real-time monitoring of an elderly person at home.

To succeed there is a need to combine a number of competences like integrated electronics, communication technology, embedded real-time systems, software and digital signal processing. For the integrated electronics part, work with issues like power optimization is ongoing. With respect to communication technology the key issue is to design a wireless wearable and human centric network based on the communication channel in a near human body environment. For the software part plans are made for developing application frameworks for the different system components. Finally, with respect to signal processing, some of the data processing will be performed locally on the testbed. For certain applications the system must run in real-time and have a very high reliability as e.g., with continuous heart-rate-variability monitoring.

In future generation of ASE-BAN much more emphasis will be put into the security and privacy aspects. A security framework adapted to wireless body area network has to be sufficiently light-weighted to meet the constraints of the sensor devices. On the other hand it also needs to be capable of providing the in-depth security and privacy required for the wireless sensor applications. Flexible security mechanisms must be developed and new generation of system on chips must offer basic security features as an embedded part of the chip.

Besides a more complete integration and intentions to apply the platform with trial users plans for a third generation platform are under preparation.

ACKNOWLEDGMENT

The authors would like to thank designer Lena Monrad Gade from the company *Designers by Choice* for her novel and insightful design of a sensor node case.

REFERENCES

[1] J. K. Madsen, H. Karstoft, F.O. Hansen, and T.S. Toftegaard, “ASE-BAN – a Wireless Body Area Network Testbed,” Proceedings EMERGING 2010, Florence, Italy, 2010, pp. 1-4.

[2] S. Wagner and C. Nielsen, "OpenCare project: An open, flexible and easily extendible infrastructure for pervasive healthcare assisted living solutions," 3rd International Conference on Pervasive Computing Technologies for Healthcare, April 2009, pp. 1-10.

[3] P. Patel and J. Wang, “Applications, Challenges, and Prospective in Emerging Body Area Networking Technologies,” IEEE Wireless Communications, Februar 2010, pp. 80-88.

[4] K. Van Dam, S. Pitchers, and M. Barnard, “Body area networks: Towards a wearable future," in Proceedings of WWRF kick off meeting, Munich, Germany, 6-7 March 2001.

[5] F.O. Hansen and T.S. Toftegaard “Requirements and System Architecture for a Healthcare Wireless Body Area Network”, International Conference on Health Informatics, HEALTHINF, January 2011, pp. 193-199.

[6] G. Z. Yang, “Body Sensor Networks,” Springer-Verlag, 2006. London, U.K.

[7] B. Latré, B. Braem, I. Moerman, C. Blondia, and P. Demeester, "A survey on wireless body area networks," Wirel.Netw., vol. 17, no. 1, pp. 1-18.

[8] M. Chen, S. Gonzalez, A. Vasilakos, H. Cao, and V.C. Leung, "Body Area Networks: A Survey," Mob.Netw.Appl., vol. 16, no. 2, pp. 171-193.

[9] C. Liolios, C. Doukas, G. Fourlas, and I. Maglogiannis, "An overview of body sensor networks in enabling pervasive healthcare and assistive environments," Proceedings of the 3rd International Conference on Pervasive Technologies Related to Assistive Environments, ACM, New York, NY, USA, pp. 43:1-43:10.

[10] K. JeongGil, L. Chenyang, M.B. Srivastava, J.A. Stankovic, A. Terzis, and M. Welsh, "Wireless Sensor Networks for Healthcare,", Proceedings of the IEEE, vol. 98, no. 11, pp. 1947-1960.

[11] F. Tufail and M.H. Islam, "Wearable Wireless Body Area Networks," International Conference on Information Management and Engineering, ICIME '09, April 2009, pp. 656-660.

[12] M.A. Hanson, H.C. Powell, A.T. Barth, K. Ringgenberg, B.H. Calhoun, J.H. Aylor, and J. Lach, "Body Area Sensor Networks: Challenges and Opportunities," Computer, vol. 42, no. 1, pp. 58-65.

[13] S. Ullah, H. Higgins, B. Braem, B. Latré, C. Blondia, I. Moerman, S. Saleem, Z. Rahman, and K. Kwak, "A Comprehensive Survey of Wireless Body Area Networks," Journal of medical systems, pp. 1-30.

[14] J. Xing and Y. Zhu, "A survey on body area network," Proceedings of the 5th International Conference on Wireless communications, networking and mobile computing, IEEE Press, Piscataway, NJ, USA, pp. 404-407.

[15] C. Gomez and J. Paradells, “Wireless home automation networks: A survey of architectures and technologies”, IEEE Communications Magazine, vol. 48, issue 6, 2010, pp. 92-101.

[16] Y. Hao and R. Foster, "Wireless body sensor networks for health-monitoring applications," Physiological Measurement, vol. 29, no. 11, pp. R27-R56.

[17] J.M. Quero, C.L. Tarrida, J.J Santana, V. Ermolov, I. Jantunen, H. Laine, and J. Eichholz, "Health Care Applications Based on Mobile Phone Centric Smart Sensor Network," 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS, August 2007, pp. 6298-6301.

[18] V. Shnayder, B. Chen, K. Lorincz, T.R.F. Fulford-Jones, and M. Welsh “Sensor nets for medical care,” Harvard University Technical Report TR-08-05, April 2005.

[19] D. Curtis, E. Shih, J. Waterman, J. Guttag, J. Bailey, T. Stair, R.A. Greenes, and L. Ohno-Machado, “Physiological signal monitoring in the waiting area of an emergency room,” Proceedings of the ICST 3rd international conference on Body area networks, BodyNets 2008, pp. 5:1-5:8.

[20] T. Gao, T. Massey, M. Sarrafzadeh, L. Selavo, and M. Welsh, “Participatory user centered design techniques for a large scale ad-hoc health information system,” Proceedings of the 1st ACM SIGMOBILE international workshop on Systems and networking support for healthcare and assisted living environments, 2007, pp. 43-48.

[21] S. Jiang, Y. Cao, S. Lyengar, P. Kuryloski, R. Jafari, Y. Xue, R. Bajcsy, and S. Wicker, “CareNet: an integratied wireless sensor networking environemnt for remote healthcare,” Proceedings of the ICST 3rd international conference on Body area networks, BodyNets 2008, pp. 9:1-9:3.

[22] A.S. Mahmoud, T.R. Sheltami, and M.H. Abu-Amara, “Wireless sensor network implementation for mobile patient,” IEEE GCC Conference, GCC, 2006, pp. 1-5.

[23] R. DeVaul, M. Sung, J. Gips, and A. Pentland, “MIThril 2003: applications and architecture,” Proceedings. Seventh IEEE International Symposium on Wearable Computers, October 2003, pp. 4-11.

[24] E. Farella, A. Pieracci, and A. Acquaviva, “Design and implementation of WiMoCA node for a body area wireless sensor network,” Systems Communications, 2005. Proceedings, 2005, pp. 342-347.

[25] M.K. Garg, D. Kim, D.S. Turaga, and B. Prabhakaran, "Multimodal analysis of body sensor network data streams for real-time healthcare," ACM Proceedings of the international conference on Multimedia information retrieval, New York, NY, USA, pp. 469-478.

[26] M.R. Narayanan, M.E. Scalzi, S.J. Redmond, S.R. Lord, B.G. Celler, and N.H. Lovell, "A wearable triaxial accelerometry system for longitudinal assessment of falls risk," Engineering in Medicine and Biology Society, 2008. EMBS 2008. 30th Annual International Conference of the IEEE, aug., pp. 2840-2843.

[27] P.S. Hall, Y. Hao, and S.L. Cotton, "Advances in antennas and propagation for body centric wireless communications," Antennas and Propagation (EuCAP), 2010 Proceedings of the Fourth European Conference on, april, pp. 1-7.

[28] T. Zasowski, F. Althaus, M. Stager, A. Wittneben, and G. Troster, "UWB for noninvasive wireless body area networks: channel measurements and results,", IEEE Conference on Ultra Wideband Systems and Technologies, 2003, pp. 285-289.

[29] R. Fisher, "60 GHz WPAN Standardization within IEEE 802.15.3c", International Symposium on Signals, Systems and Electronics, ISSSE '07, August 2007, pp. 103-105.

[30] S. Alipour, F. Parvaresh, H. Ghajari, and F.K. Donald, "Propagation characteristics for a 60 GHz Wireless body area network (WBAN)," Military Communications Conference, MILCOM, November 2010, pp. 719-723.

[31] C.C.Y. Poon, Y. Zhang, and S. Bao, "A novel biometrics method to secure wireless body area sensor networks for telemedicine and m-health," IEEE Communications Magazine, vol. 44, no. 4, pp. 73-81.

[32] M. Pulli, "System for Monitoring People's Home Activities", Dissertation/Thesis, Unpublished, Helsinki university of technology, Espoo, Finland, 2010, url http://www.diem.fi/news/diem-thesis-a-system-for-monitoring-peoples-home-activities, accessed January 2012.

[33] A. Johansson, S. Wei, and X. Youzhi, "An ANT Based Wireless Body Sensor Biofeedback Network for Medical E-Health Care". 7th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM), 2011 pp.1-5

[34] X. Yu, X. Xia, and X. Chen, "Design and Application of RuBee-Based Telemedicine Data Acquisition System", IEEE/ACIS 10th International Conference on Computer and Information Science (ICIS), pp. 365-370, 2011.

[35] L. Ho, M. Moh, Z. Walker, T. Hamada,and C-F. Su, "A prototype on RFID and sensor networks for elder healthcare: progress report", Proceedings of the 2005 ACM SIGCOMM workshop on Experimental approaches to wireless network design and analysis, 2005, pp. 70-75.

[36] ANT web site. URL: http://www.thisisant.com/. Accessed January 2012.

[37] N.F. Timmons, and W.G. Scanlon, "Analysis of the performance of IEEE 802.15.4 for medical sensor body area networking," First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks, SECON 2004. November 2004, pp. 16-24.

[38] D. Yazar and A. Dunkels, "Efficient application integration in IP-based sensor networks," Proceedings of the First ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings, 2009, pp. 43-48.

[39] A. Natarajan, B. de Silva, K. Yap, and M. Motani, "To hop or not to hop: network architecture for body sensor networks," Proceedings of the 6th Annual IEEE communications society conference on Sensor, Mesh and Ad Hoc Communications and Networks. IEEE Press, Piscataway, NJ, USA, pp. 682-690.

[40] IEEE 802.15.4 Task Group 6 (TG6). URL: http://www.ieee802.org/15/pub/TG6.html. Accessed January 2012.

[41] S. Corroy and H. Baldus, "Low power medium access control for body-coupled communication networks,", 6th International Symposium on Wireless Communication Systems, ISWCS, September 2009, pp. 398-402.

[42] Y. Wei, J. Heidemann, and D. Estrin, "An energy-efficient mac protocol for wireless sensor networks," in Proceedings of IEEE Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM, vol. 3, June 2002, pp. 1567–1576.

[43] T. van Dam and K. Langendoen, "An adaptive energy-efficient mac protocol for wireless sensor networks," in Proceedings of the 1st international conference on Embedded networked sensor systems, SenSys03, November 2003, pp. 171-180.

[44] A. El-Hoiydi and J. Decotignie, "Low power downlink mac protocols for infrastructure wireless sensor networks," Mobile Networks and Applications, vol. 10, no. 5, October 2005, pp. 675-690.

[45] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in Proceedings of the 2nd international conference on Embedded networked sensor systems, SenSys04, November 2004, p. 95-107.

[46] S.A. Gopalan, D. Kim, J. Nah, and J. Park, "A survey on power-efficient MAC protocols for wireless body area networks," 3rd IEEE International Conference on Broadband Network and Multimedia Technology, IC-BNMT, October 2010, pp. 1230-1234.

[47] IETF 6loWPAN working group. URL: http://datatracker.ietf.org/wg/6lowpan/. Accessed January 2012.

[48] TinyOS web site,URL: http://www.tinyos.net. Accessed January 2012.

[49] Contiki web site, URL: http://www.sics.se/contiki/. Accessed January 2012

[50] A. Waluyo, I. Pek, X. Chen, W. Yeoh, "Design and evaluation of lightweight middelware for personal wireless body area network,". Personal and Ubiquitous Computing, vol. 13, issue 9, 2009, pp. 509-525.

[51] ISO/IEEE 11073 Personal Health Data (PHD) Standards, IEEE std. 11073-20601 – Application profile – optimized exchange protocol.

[52] E. Kim, S. Lim, J. Ahn, J. Nah, and N. Kim, "Integration of IEEE 1451 and HL7 exchaning information for Patients sensor data," Journal Medical Systems, vol. 34, issue 6, 2010, pp. 1033-1041.

[53] IEEE std 1451.0-2007, "IEEE standard for smart transducer interface for sensors and acturators – common functions, communication protocols, and transducer electronic data sheet (TEDS) formats".

[54] B. Zhen, M. Pastel, S. Lee. E- Won. And A. Astrin, IEEE P802.15 Wireless Personal Area Network, "TG6 Technical Requirements Document", IEEE P802.15 15-08-0644-09-0006. URL: https://mentor.ieee.org/802.15/dcn/08/15-08-0644-02-0006-tg6-technical-requirements-document.doc. Accessed January 2012.

[55] V. Shnayder, B. Chen, K. Lorincz, T. Fulford-Jones, and M. Welsh, "Sensor Networks for Medical Care," Proceedings of the 3rd international conference on Embedded networked sensor system, 2005, pp. 314-327.

[56] J. Pan and W. J. Tompkins, "A real-time QRS detection algorithm". IEEE Transaction on Biomedical Engineering, vol. BME-32, no. 3, 1985, pp. 230-236.

[57] T. Falck, H. Baldus, J. Espina, and K. Klabunde, "Plug´n play simplicity for wireless medical body sensors," Mobile Network Applications, vol. 12, no.2-3, 2007, pp. 143-153.

[58] ContinuaAlliance, http://www.continuaalliance.org. Accessed January 2012.

[59] P. Brandão and J. Bacon, "Body sensor networks: can we use them?" M-PAC'09 Proceedings of the International Workshop on Middleware for Pervasive Mobile and Embedded Computing, 2009, pp. 3:1-3:6.

[60] H. Mayumi and O. Masakazu, "Applying XML Web Services into Health Care Management," hicss, vol. 6, pp.155a, HISSC'05 Proceedings of the Proceedings of the 38th Annual Hawaii International Conference on System Sciences, 2005, p. 155.1.

[61] S. Saadaoui and L. Wolf, "Architecture Concept of a Wireless Body Area Sensor Network for Health Monitoring of Elderly People," 4th IEEE Consumer Communications and Networking Conference, CCNC 2007, January 2007, pp. 722-726.

[62] E. Jovanov, A. Milenkovic, C. Otto, and P.C. de Groen, "A wireless body area network of intelligent motion sensors for computer assisted rehabilitation," Journal of NeuroEngineering and Rehabilitation, Vol. 2, no. 6, 2005, pp. 6-16.

[63] C. Otto, A. Milenkovic, C. Sanders, and E. Jovanov, "System Architecture of a wireless body area sensor network for ubiquitous health monitoring," Journal of Mobile Multimedia. Vol. 1. No. 4, 2006, pp. 307-326.

[64] K. Kortermand, R.H. Jacobsen, and T.S. Toftegaard, "Routing analysis of wireless body area networks using path availability measurements," 4th International Conference on Biomedical Engineering and Informatics (BMEI), vol.3, October 2011, pp. 1380-1385.

[65] P. Levis, S. Madden, J. Polastre, R. Szewczyk, K. Whitehouse, A. Woo, D. Gay, J. Hill, M. Welsh, E. Brewer, and D. Culler. “TinyOS: An operating system for wireless sensor networks,” In Ambient Intelligence, Springer-Verlag, 2004, pp. 115-148.

[66] TinyOS web site, BLIP tutorial, URL: http://docs.tinyos.net/index.php/BLIP_Tutorial. Accessed January 2012.

[67] Energy Micro, News Archieve: “Details of the energy friendly EFR4D radio product family”. URL: http://www.energymicro.com/news-archive/energy-micro-announces-details-of-energy-friendly-radio-product-family, Accessed January 2012.

[68] L. Krishnamachari, D. Estrin, and S. Wicker, "The impact of data aggregation in wireless sensor networks," ICDCSW '02 Proceedings of the 22nd International Conference on Distributed Computing Systems, 2002, pp. 575-578.

[69] Atmel ATmega128RF1 Datasheet URL: http://www.atmel.com/dyn/resources/prod_documents/doc8266.pdf, 2010. Accessed January 2012.

[70] Texas Instruments CC430F6126 Datasheet. Doc. No. SLAS554E –MAY 2009–REVISED NOVEMBER 2010. Texas Instruments. URL: http://focus.ti.com/lit/ds/symlink/cc430f6126.pdf. Accessed January 2012.

[71] Ember EM351/EM357 High-preformance, Integrated ZigBee/802.15.4 System-on-Chip. Datasheet: Ember, 2011. URL: http://www.ember.com/pdf/120-035X-000_EM35x_Datasheet.pdf . Accessed January 2012.

[72] Texas Instruments MSP430F1611 Datasheet, Texas Instruments, 2011, URL: http://focus.ti.com/lit/ds/symlink/msp430f1611.pdf. Accessed January 2012.

[73] Chipcon CC2420 Datasheet. Texas Instruments, 2007. URL: http://focus.ti.com/lit/ds/symlink/cc2420.pdf. Accessed January 2012 .

[74] TelosB Datasheet, Memsic, 2010, URL: http://www.memsic.com/support/documentation/wireless-sensor-networks/category/7-datasheets.html?download=152%3Atelosb. Accessed January 2012.

[75] C. Gomez, A. Boix, and J. Paradells, “Impact of LQI-based routing metrics on the performance of a one-to-one routing protocol for IEEE 802.15.4 multihop networks,” EURASIP Journal on Wireless Communications and Networking, 2010, February 2010, pp. 6:1–6:20.

[76] B. Chen, K.-K. Muniswamy-Reddy, and M. Welsh. “Ad-hoc multicast routing on resource-limited sensor nodes,” in Proceedings of the 2nd International Workshop on Multi-hop Ad, 2006, pp. 87–94.

[77] J.W. Hui and D.E. Culler, “IPv6 in Low-Power Wireless Networks”, Proceedings of the IEEE, vol. 98, no. 11, pp. 1865-1878.

[78] K. Jeonggil, A. Terzis, S. Dawson-Haggerty, D.E. Culler, J.W. Hui, and P. Levis, "Connecting low-power and lossy networks to the internet," IEEE Communications Magazine, vol. 49, no. 4, pp. 96-101.

[79] Z. Shelby, “Embedded web services,” IEEE Wireless Communications, vol. 17, no. 6, pp. 52-57.

[80] K. Kuladinithi, O. Bergmann, T. Pötsch, M. Becker, and C. Görg. “Implementation of CoAP and its Application in Transport Logistics,” Presented at IP+SN 2011, April 11th 2011, Chicago. URL: http://hinrg.cs.jhu.edu/joomla/images/stories/coap-ipsn.pdf. Accessed January 2011

[81] P. Hii and W. Chung, "A Comprehensive Ubiquitous Healthcare Solution on an Android™ Mobile Device," Sensors, vol. 11, no. 7, pp. 6799-6815.

[82] Android OS: URL: http://code.google.com/intl/da-DK/android/, Accessed January 2012.

[83] A.A. Pena, “A feasibility Study of the Suitability of an AD5933-based Spectrometer for EBI Applications,” Final degree thesis, 9/2009, University of Borås.

[84] DS/EN 60601-2-27. “Medical electrical equipment – Part 2-27”, 2nd revision, 2006.

[85] Guidelines. “Heart rate variability, Standards of measurement, physiological interpretation, and clinical use,” European Heart Journal vol. 17, 1996, pp. 354-381.

[86] C. Park, P.H. Chou, Y. Bai, R. Matthews, and A. Hibbs, "An ultra-wearable, wireless, low power ECG monitoring system", Biomedical Circuits and Systems Conference, November 2006, pp. 241-244.

[87] T. Bragge, M.P. Tarvainen, P.O. Ranta-Aho, and P.A. Karjalainen, “High-resolution QRS fiducial point corrections in sparsely sampled ECG recordings,” Physiological Measurement. 2005, vol. 26, no. 5, pp. 743-753.

[88] Analog Devices, “High Performance General Purpose Blackfin Processor,” 2008, URL: http://www.analog.com/en/embedded-processing-dsp/blackfin/adsp-bf533/processors/product.html. Accessed January 2012.

[89] L. Nord and J. Haartsen, “The Bluetooth Radio Specification and the Bluetooth Baseband Specification”, Bluetooth, 1999-2000.

[90] H. Darabi, S. Khorram, H.M Chien, M.A. Pan, S. Wu, S. Moloudi, J.C., Leete, J.J. Rael, M. Syed, B. Ibrahim, M. Rofougaran, and A. Rofougaran, “A 2.4 GHz CMOS Transceiver for Bluetooth,” IEEE Journal of Solid-State Circuits, vol. 36, issue 12, 2001, pp. 2016-2024.

[91] F.O. Eynde, J.J. Schmit, V. Charlier, R. Alexandre, C. Sturman, K. Coffin, B. Mollekens, J. Craninckx, S. Terrijn, A. Monterastelli, S. Beerens, P.Goetschalckx, M. Ingels, D. Joos, S. Guncer, and A. Pontioglu, “A fully integrated single chip SOC for Bluetooth,” 2001 IEEE Solid-State Circuits Conference, ISSCC, Digest of Technical Papers, 2001, pp. 196-197.

[92] R. Amirtharajah, S. Meninger, J.O. Mur-Miranda, A. Chandrakasan, and J. Lang., “A micropwer programmable DSP powered using a MEMS based vibratio-to-electric energy converter,” 2000 IEEE Solid-State Circuits Conference, ISSCC, Digest of Technical Papers, 2000, pp. 362-363.

[93] K. Holger and A. Willig, “Protocols and Architectures for Wireless Sensor Networks,” Wiley 2005, pp. 40-45.

[94] C. Andersen and E. Rasmussen,”Development of an energy optimized Wireless Body Area Network (WBAN),”. Master thesis at Aarhus University, December 2010.

[95] SixXS IPv6 Deployment & Tunnel Broker, URL: http://www.sixxs.net. Accessed January 2012.

[96] CyanogenMod: A customized aftermarket firmware distribution, URL: http://www.cyanogenmod.com. Accessed January 2012.

[97] JSON: JavaScript Object Notation, URL: http://www.json.org. Accessed January 2012.

A One-Shot Dynamic Optimization Methodology and Application Metrics Estimation Model for Wireless Sensor Networks

Arslan Munir and Ann Gordon-Ross
Department of Electrical and Computer Engineering
University of Florida, Gainesville, Florida 32611
Email: amunir@ufl.edu, ann@ece.ufl.edu

Susan Lysecky and Roman Lysecky
Department of Electrical and Computer Engineering
University of Arizona, Tucson, Arizona 85721
Email: {slysecky, rlysecky}@ece.arizona.edu

***Abstract*—Wireless sensor networks (WSNs), consisting of autonomous sensor nodes, have emerged as ubiquitous networks that span diverse application domains (e.g., health care, logistics, defense) each with varying application requirements (e.g., lifetime, throughput, reliability). Typically, sensor-based platforms possess tunable parameters (e.g., processor voltage, processor frequency, sensing frequency), which enable platform specialization for particular application requirements. WSN application design can be daunting for application developers, which are oftentimes not trained engineers (e.g., biologists, agriculturists) who wish to utilize the sensor-based systems within their given domain. Dynamic optimizations enable sensor-based platforms to tune parameters in-situ to automatically determine an optimized operating state. However, rapidly changing application behavior and environmental stimuli necessitate a lightweight and highly responsive dynamic optimization methodology. In this paper, we propose a very lightweight dynamic optimization methodology that determines initial tunable parameter settings to give a high-quality operating state in *one-shot* for time-critical and highly constrained applications. We compare our one-shot dynamic optimization methodology with other lightweight dynamic optimization methodologies (i.e., greedy- and simulated annealing-based) to provide insights into the solution quality and resource requirements of our methodology. Results reveal that the one-shot solution is within 8% of the optimal solution on average. To assist dynamic optimizations in determining an operating state, we propose an application metric estimation model to establish a relationship between application metrics (e.g., lifetime, throughput) and sensor-based platform parameters.**

***Keywords*-Wireless sensor networks, dynamic optimization, application metrics estimation**

I. Introduction and Motivation

Wireless sensor networks (WSNs) consist of spatially distributed autonomous sensor nodes that observe a phenomenon (environment, target, etc.). WSNs are becoming ubiquitous because of their proliferation in diverse application domains (e.g., defense, health care, logistics) each with varying application requirements (e.g., lifetime, throughput, reliability) [1]. For example, a security/defense system may have a higher throughput requirement whereas an ambient conditions monitoring application may be more sensitive to lifetime. This diversity makes WSN design challenging with commercial-off-the-shelf (COTS) sensor nodes.

COTS sensor nodes are mass-produced to optimize cost and are not specialized for any particular application. Furthermore, WSN application developers oftentimes are not trained engineers, but rather biologists, teachers, or agriculturists who wish to utilize the sensor-based systems within their given domain. Fortunately, many COTS sensor nodes possess tunable parameters (e.g., processor voltage and frequency, sensing frequency) whose values can be *tuned* for a specific application. Faced with an overwhelming number of tunable parameter choices, WSN design can be a daunting task for non-experts and necessitates an automated parameter tuning process for assistance.

Parameter optimization is the process of assigning appropriate (optimal or near-optimal) values to tunable parameters either statically or dynamically to meet application requirements. *Static optimizations* assign parameter values at deployment and these values remain fixed during the sensor node's lifetime. Accurate prediction/simulation of environmental stimuli is challenging and applications with changing environmental stimuli do not benefit from static optimizations. Alternatively, *dynamic optimizations* assign parameter values during runtime and reassign/change these values in accordance with changing environmental stimuli, thus enabling close adherence to application requirements.

There exists much research in the area of dynamic optimizations [2][3][4][5][6], but most previous work targets the memory (cache) or processor in computer systems. Little work exists on WSN dynamic optimization, which presents additional challenges because of a unique design space, energy constraints, and operating environment. The dynamic profiling and optimization (DPOP) project aspires to alleviate the complexities associated with sensor-based system design using dynamic profiling methods capable of observing application-level behavior and dynamic optimization to tune the underlying platform accordingly [7]. The DPOP project has evaluated dynamic profiling methods for observing application-level behavior by gathering profiling statistics, but dynamic optimization methods still need exploration. In this paper, we explore a fine-grained design space for sensor-based platforms with many tunable parameters to more closely meet application requirements (Gordon-Ross et al. [8] showed that finer-grained design spaces provide interesting design alternatives and result in increased benefits in the cache subsystem). The exploration of a fine-grained design space coupled with limited battery reserves and rapidly changing application requirements and environmental stimuli necessitates a lightweight and highly responsive dynamic optimization methodology.

Our main contributions in this paper are:

- We propose a lightweight dynamic optimization methodology that determines appropriate initial tunable parameter values to give a good quality operating state (tunable parameter value settings) in *one-shot* with minimal design exploration for highly constrained applications. Results reveal that this one-shot operating state is within 8% of the optimal solution (obtained from exhaustive search) averaged over several different application domains and design spaces.
- We evaluate alternative initial parameter settings to provide a comparison with our one-shot initial parameter settings. Results reveal that the average percentage improvement attained by the one-shot initial parameter settings over alternative initial parameter settings for different application domains and design spaces is 33% on average.
- We analyze memory and execution time requirements of our one-shot dynamic optimization methodology and compare these with other lightweight dynamic optimization methodologies (greedy- and simulated-annealing (SA)-based). Results indicate that our one-shot dynamic optimization methodology requires 204% and 458% less memory on average as compared to the greedy- and SA-based methodologies, respectively. The one-shot solution requires 18% less execution time on average as compared to the greedy- and SA-based methodologies even if these methodologies are restricted to explore only 0.03% of the design space on average.
- To assist dynamic optimizations in determining an operating state, we for the first time, to the best of our knowledge, propose an *application metric estimation model*, which estimates high-level application metrics (lifetime, throughput, and reliability) from sensor-based platform parameters (e.g., processor voltage and frequency, sensing frequency, transceiver transmission power, etc.). Our one-shot dynamic optimization methodology leverages this estimation model when comparing different operating states for optimization purposes. We emphasize that this application metric estimation model can be leveraged by any dynamic optimization methodology and facilitates the WSN design process.

The remainder of this paper is organized as follows. Section II surveys previous work in the area of dynamic optimizations. Section III presents our one-shot dynamic optimization methodology and Section IV describes our application metrics estimation model leveraged by our one-shot dynamic optimization methodology. Section V presents experimental results and Section VI presents conclusions and future research work directions.

II. RELATED WORK

There exists much research in the area of dynamic optimizations [2][3][4][5][6][9][10], however, most previous work focuses on the processor or memory (cache) in computer systems. Whereas previous work can provide valuable insights into WSN dynamic optimizations, these works are not directly applicable due to a WSN's unique design space, energy constraints, and operating environment.

In the area of WSN dynamic profiling and optimizations, Sridharan et al. [11] obtained accurate environmental stimuli by dynamically profiling the WSN's operating environment, but did not propose any methodology to leverage these profiling statistics for optimizations. Shenoy et al. [12] presented profiling methods for dynamically monitoring sensor-based platforms and analyzed the associated network traffic and energy, but did not explore dynamic optimizations. In prior work, Munir et al. [13] proposed a Markov Decision Process (MDP)-based methodology as a first step towards WSN dynamic optimizations, but this method required prohibitively large computational resources for larger design spaces. Ideally, this method required a base station node with more computing resources to carry out the optimal operating state determination process, and these operating states could be communicated to other sensor nodes. The large computational requirements inhibited the methodology's implementation on resource constrained sensor nodes to enable autonomous operating state decisions. Kogekar et al. [14] proposed an approach for dynamic software reconfiguration in WSNs using adaptive software, which used tasks to detect environmental changes (event occurrences) and then adapted the software to the new conditions. Though their work considered software reconfiguration, they did not consider senor node tunable parameters.

In the area of WSN optimizations, Wang et al. [15] proposed a distributed energy optimization method for target tracking applications. The energy management mechanism consisted of an optimal sensing scheme that leveraged dynamic awakening of sensor nodes. The dynamic awakening scheme awoke the group of sensor nodes located in the target's vicinity for reporting the sensed data. The results verified that dynamic awakening combined with optimal sensor node selection enhanced the WSN energy efficiency. Liu et al. [16] proposed a dynamic node collaboration scheme for mobile target tracking in wireless camera sensor networks (wireless camera sensor networks can provide much more accurate information in target tracking applications as compared to traditional sensor networks). The proposed scheme comprised of two components: a cluster head election scheme during the tracking process and an optimization algorithm to select an optimal subset of camera sensors as the cluster members for cooperative estimation of the target's location. Khanna et al. [17] proposed a reduced-complexity genetic algorithm for secure and dynamic deployment of resource constrained multi-hop WSNs. The genetic algorithm adaptively configured optimal position and security attributes by dynamically monitoring network traffic, packet integrity, and battery usage.

Several papers explored dynamic voltage and frequency scaling (DVFS) for reduced energy consumption in WSNs. Min et al. [18] demonstrated that dynamic processor voltage scaling reduced energy consumption by 60%. Similarly, Yuan et al. [19] studied a DVFS system that used additional transmitted data packet information to select appropriate processor voltage and frequency values. Although DVFS provides a mechanism for dynamic optimizations, considering additional sensor node tunable parameters increases the design space and the sensor node's ability to better meet application requirements. To the best of our knowledge, our work is the first to explore an extensive sensor node design space.

In prior work, Lysecky et al. [20] proposed SA-based automated application specific tuning of parameterized sensor-based embedded systems and found that automated tuning can better meet application requirements by 40% on average as compared to a static configuration of tunable parameters. Verma [21] studied SA-based and particle swarm optimization (PSO) methods for automated application specific tuning and observed that an SA-based method performed better than PSO because PSO often quickly converged to local minima. Exhaustive search algorithms have been used in literature for performance analysis and comparison with heuristic algorithms. Mannion et al. [22] proposed a PareDown decomposition algorithm for partitioning pre-defined behavioral blocks onto a minimum number of programmable sensor blocks and compared the partitioning algorithm's performance with an exhaustive search algorithm. Meier et al. [23] proposed an exhaustive search based scheme called NoSE (Neighbor Search and link Estimation) for neighbor search, link assessment, and energy consumption minimization.

Even though there exists some work on optimizations in WSNs [15][18][19][24][25][26][27], dynamic optimizations require further research and more in depth considerations. Specifically, a sensor node's constrained energy and storage resources necessitate lightweight dynamic optimization methodologies for sensor node parameter tuning.

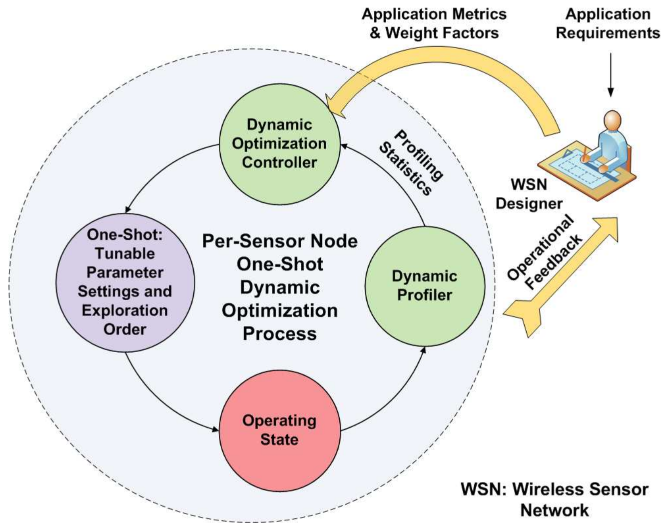


Figure 1. One-shot dynamic optimization methodology for wireless sensor networks.

III. Dynamic Optimization Methodology

In this section, we give an overview of our one-shot dynamic optimization methodology and the associated algorithm. We also formulate the state space and objective function for our methodology.

A. Overview

Fig. 1 depicts our one-shot dynamic optimization methodology for WSNs. WSN designers evaluate application requirements and capture these requirements as high-level *application metrics* (e.g., lifetime, throughput, reliability) and associated *weight factors*. The weight factors signify the relative weightage/importance of application metrics with respect to each other. The dynamic optimization methodology leverages an application metric estimation model to determine application metric values offered by an operating state (we describe this application metric estimation model in Section IV).

Fig. 1 shows the per-node one-shot dynamic optimization process (encompassed by the dashed circle), which is orchestrated by the *dynamic optimization controller*. The dynamic optimization controller invokes the *one-shot step* wherein the sensor node operating state is directly determined by intelligent tunable parameter value settings, and hence the methodology is termed as *one-shot*. The one-shot step also determines an exploration order (ascending or descending) for tunable parameters. This exploration order can be leveraged by an *online optimization algorithm* to provide improvements over the one-shot solution by further design space exploration and is the focus of our future work. This exploration order is critical in reducing the number of states explored by the online optimization algorithm. The sensor node moves directly to the operating state specified by the one-shot step. A *dynamic profiler* records profiling statistics (e.g., battery energy, wireless channel condition) given the current operating state and environmental stimuli and passes these profiling statistics to the dynamic optimization controller.

The dynamic optimization controller processes the profiling statistics to determine if the current operating state meets the application requirements. If the application requirements are not met, the dynamic optimization controller reinvokes the one-shot dynamic optimization process to determine the new operating state. This feedback process continues to ensure the selection of a good operating state to better meet application requirements in the presence of changing environmental stimuli.

B. State Space

The state space S for our one-shot dynamic optimization methodology given N tunable parameters is defined as:

$$S = P_1 \times P_2 \times \cdots \times P_N \tag{1}$$

where P_i denotes the state space for tunable parameter i, $\forall\, i \in \{1, 2, \ldots, N\}$ and $\times$ denotes the Cartesian product. Each tunable parameter P_i consists of n values:

$$P_i = \{p_{i_1}, p_{i_2}, p_{i_3}, \ldots, p_{i_n}\} \;\;:\;\; |P_i| = n \tag{2}$$

where $|P_i|$ denotes the tunable parameter P_i's state space cardinality (the number of tunable values in P_i). S is a set of n-tuples (each n-tuple s represents a sensor node state) formed by taking one tunable parameter value from each tunable parameter. A single n-tuple $s \in S$ is given as:

$$\begin{aligned} s \;\; &= \;\; (p_{1_y}, p_{2_y}, \ldots, p_{N_y}) : p_{i_y} \in P_i, \\ & \quad\;\; \forall\, i \in \{1, 2, \ldots, N\},\, y \in \{1, 2, \ldots, n\} \end{aligned} \tag{3}$$

We point out that some n-tuples in S may not be feasible (such as invalid combinations of processor voltage and frequency) and can be treated as *do not care* tuples.

C. Optimization Objection Function

The sensor node dynamic optimization problem can be formulated as an unconstrained optimization problem:

$$\begin{aligned} \max f(s) \;\; &= \;\; \sum_{k=1}^{m} \omega_k f_k(s) \\ s.t. \;\;\; & s \in S \\ & \omega_k \geq 0, \quad k = 1, 2, \ldots, m \\ & \omega_k \leq 1, \quad k = 1, 2, \ldots, m \\ & \sum_{k=1}^{m} \omega_k = 1, \end{aligned} \tag{4}$$

where $f(s)$ denotes the objective function characterizing application metrics and weight factors. $f_k(s)$ and ω_k in (4) denote the objective function and weight factor for the

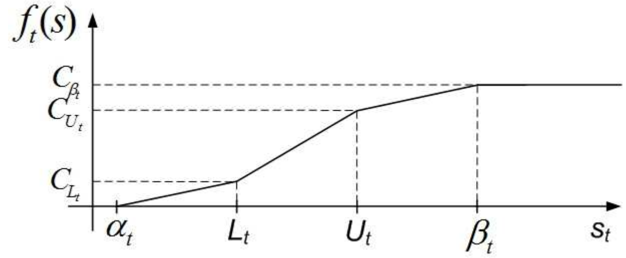


Figure 2. Throughput objective function $f_t(s)$.

k^{th} application metric, respectively, given that there are m application metrics. Each state $s \in S$ has an associated objective function value and the optimization goal is to determine a state that gives the maximum (optimal) objective function value $f^{opt}(s)$ ($f^{opt}(s)$ indicates the best possible adherence to the specified application requirements given the design space S). The solution quality for any $s \in S$ can be determined by normalizing the objective function value corresponding to state s with respect to $f^{opt}(s)$. The normalized objective function value corresponding to a state can vary from 0 to 1 where 1 indicates the optimal solution.

For our dynamic optimization methodology, we consider three application metrics (m = 3), lifetime, throughput, and reliability, whose objective functions are denoted by $f_l(s)$, $f_t(s)$, and $f_r(s)$, respectively. We define $f_t(s)$ (Fig. 2) using the piecewise linear function:

$$f_t(s) = \begin{cases} 1, & s_t \geq \beta_t \\ C_{U_t} + \frac{(C_{\beta_t} - C_{U_t})(s_t - U_t)}{(\beta_t - U_t)}, & U_t \leq s_t < \beta_t \\ C_{L_t} + \frac{(C_{U_t} - C_{L_t})(s_t - L_t)}{(U_t - L_t)}, & L_t \leq s_t < U_t \\ C_{L_t} \cdot \frac{(s_t - \alpha_t)}{(L_t - \alpha_t)}, & \alpha_t \leq s_t < L_t \\ 0, & s_t < \alpha_t. \end{cases} \tag{5}$$

where s_t denotes the throughput offered by state s, the constant parameters L_t and U_t denote the *desired* minimum and maximum throughput, respectively, and the constant parameters α_t and β_t denote the *acceptable* minimum and maximum throughput, respectively. The constant parameters C_{L_t}, C_{U_t}, and C_{β_t} in (5) denote the $f_t(s)$ value at L_t, U_t, and β_t, respectively. A piecewise linear objective function captures accurately the desirable and acceptable ranges of a particular application metric. We consider piecewise linear objective functions as a typical example, however, our methodology works well for any other objective function characterization (e.g., linear, non-linear) [13].

The $f_l(s)$ and $f_r(s)$ can be defined similar to (5).

We point out that some tunable parameters may affect multiple application metrics (e.g., sending at a lower power might conserve energy but may increase the packet loss ratio). Our dynamic optimization objective function handles such multi-effect parameters appropriately. Since our dynamic optimization objective function is a weighted sum of objective functions of individual application metrics, the overall solution will be an operating state in which the affect of these parameters is balanced out in a way that gives the maximum overall objective function value. An application metric with a higher weight factor will be given precedence in arbitrating between the tunable parameters that affect multiple application metrics.

D. One-Shot Dynamic Optimization Algorithm

In this subsection, we describe the algorithm for our one-shot dynamic optimization methodology. The algorithm determines initial tunable parameter value settings and exploration order (ascending or descending). This exploration order can be used for the exploration of tunable parameters if further improvement over the one-shot solution is desired, and this improvement is the focus of our future work.

Input: *f(s), N, n, m,* $\boldsymbol{P}$
Output: Initial tunable parameter value settings and exploration order

1 **for** $k \leftarrow 1$ **to** m **do**
2   **for** $P_i \leftarrow P_1$ **to** P_N **do**
3     $f^k_{p_{i_1}} \leftarrow k^{\text{th}}$ metric objective function value when parameter setting is $\{P_i = p_{i_1}, P_j = P_{j_0}, \forall\, i \neq j\}$;
4     $f^k_{p_{i_n}} \leftarrow k^{\text{th}}$ metric objective function value when parameter setting is $\{P_i = p_{i_n}, P_j = P_{j_0}, \forall\, i \neq j\}$;
5     $\delta f^k_{P_i} \leftarrow f^k_{p_{i_n}} - f^k_{p_{i_1}}$;
6   **if** $\delta f^k_{P_i} \geq 0$ **then**
7   explore P_i in descending order ;
8       $P^k_d[i] \leftarrow$ descending ;
9       $P^k_0[i] \leftarrow p^k_{i_n}$;
10   **else**
11   explore P_i in ascending order ;
12       $P^k_d[i] \leftarrow$ ascending ;
13       $P^k_0[i] \leftarrow p^k_{i_1}$;
14   **end**
15   **end**
16 **end**
return $\boldsymbol{P^k_d}, \boldsymbol{P^k_0}, \forall\, k \in \{1, \ldots, m\}$

Algorithm 1: One-shot dynamic optimization algorithm.

Algorithm 1 describes our one-shot dynamic optimization algorithm to determine initial tunable parameter value settings and exploration order. The algorithm takes as input the objective function $f(s)$, the number of tunable parameters N, the number of values for each tunable parameter n (we assume for simplicity that tunable parameters have an equal number of tunable values, however, other values can be taken), the number of application metrics m, and $\boldsymbol{P}$ where $\boldsymbol{P}$ represents a vector containing the tunable parameters, $\boldsymbol{P} = \{P_1, P_2, \ldots, P_N\}$. For each application metric k, the algorithm calculates vectors $\boldsymbol{P^k_0}$ and $\boldsymbol{P^k_d}$ (where d denotes the exploration direction (ascending or descending)), which store the initial value settings and exploration order, respectively, for the tunable parameters. The algorithm determines $f^k_{p_{i_1}}$ and $f^k_{p_{i_n}}$ (the k^{th} application metric objective function values) where the parameter being explored P_i is assigned its first p_{i_1} and last p_{i_n} tunable values, respectively, and the remainder of the tunable parameters $P_j, \forall\, j \neq i$ are assigned initial values (lines 3 - 4). $\delta f^k_{P_i}$ stores the difference between $f^k_{p_{i_n}}$ and $f^k_{p_{i_1}}$. $\delta f^k_{P_i} \geq 0$ means that p_{i_n} results in an equal or greater objective function value as compared to p_{i_1} for parameter P_i (i.e., the objective function value decreases as the parameter value decreases). To reduce the number of states explored while considering that an online optimization algorithm (e.g., greedy-based algorithm) will typically stop exploring a tunable parameter if a tunable parameter's value yields a comparatively lower (or equal) objective function value, P_i's exploration order must be descending (lines 6 - 8). The algorithm assigns p_{i_n} as the initial value of P_i for the k^{th} application metric (line 9). If $\delta f^k_{P_i} < 0$, the algorithm assigns the exploration order as ascending for P_i and p_{i_1} as the initial value setting of P_i (lines 11 - 13). This $\delta f^k_{P_i}$ calculation procedure is repeated for all m application metrics and all N tunable parameters (lines 1 - 16).

Algorithm 1 determines appropriate initial parameter value settings corresponding to individual application metrics, however, further calculations are required to determine intelligent initial parameter value settings suitable for all the application metrics because the best initial value settings for different application metrics may be different. Since some parameters are more critical to meeting application requirements than other parameters depending on the application metric weight factors, more consideration should be given to the initial parameter value settings corresponding to the application metrics with higher weight factors. For example, sensing frequency is a critical parameter for applications with a high responsiveness weight factor and therefore, initial value settings corresponding to the responsiveness application metric should be given priority. We devise a technique for intelligent initial value settings such that the initial value settings consider the impact of these settings on the overall objective function considering all the application metrics and the application metrics' associated weight factors. Our initial value settings technique is based on the calculations performed in Algorithm 1.

The initial value settings vector $\boldsymbol{P^k_0}$ corresponding to application metric k is given by:

$$\boldsymbol{P^k_0} = \{P^k_{0_1}, P^k_{0_2}, \ldots, P^k_{0_N}\}, \forall\, k \in \{1, 2, \ldots, m\} \qquad (6)$$

where $P^k_{0_i}$ denotes the initial value setting for tunable parameter i, $\forall\, i \in \{1, 2, \ldots, N\}$ corresponding to the k^{th} application metric (as given by Algorithm 1). An intelligent initial value setting vector $\widehat{\boldsymbol{P_0}}$ must consider all application metrics' weight factors with higher importance given to higher weight factors, i.e.,:

$$\widehat{\boldsymbol{P_0}} = \{P_{0_1}^1, \ldots, P_{0_{l_1}}^1, P_{0_1}^2, \ldots, P_{0_{l_2}}^2, P_{0_1}^3, \ldots, P_{0_{l_3}}^3, \ldots, P_{0_1}^m, \ldots, P_{0_{l_m}}^m\} \quad (7)$$

where l_k denotes the number of initial value settings taken from $P_0^k, \forall\ k \in \{1, 2, \ldots, m\}$ such that $\sum_{k=1}^m l_k = N$. Our technique allows taking more initial value settings corresponding to application metrics with higher weight factors (since Algorithm 1 gives appropriate initial value settings for each application metric separately), i.e., $l_k \geq l_{k+1} \Leftrightarrow \omega_k \geq \omega_{k+1}, \forall\ k \in \{1, 2, \ldots, m-1\}$. In (7), l_1 initial value settings are taken from vector P_0^1, then l_2 from vector P_0^2, and so on to l_m from vector P_0^m such that $\{P_{0_1}^k, \ldots, P_{0_{l_k}}^k\} \cap \{P_{0_1}^{k-1}, \ldots, P_{0_{l_{k-1}}}^{k-1}\} = \emptyset, \forall\ k \in \{2, 3, \ldots, m\}$. In other words, we select those initial value settings corresponding to the application metrics with lower weight factors that are not already selected based on the application metrics with higher weight factors (i.e., $\widehat{\boldsymbol{P_0}}$ comprises of disjoint or non-overlapping initial value settings).

In the situation where a weight factor ω_1 is much greater than all of the other weight factors, an intelligent initial value setting $\widetilde{\boldsymbol{P_0}}$ would correspond to the initial value settings based on the application metric with weight factor ω_1, i.e.,:

$$\widetilde{\boldsymbol{P_0}} = \boldsymbol{P_0^1} = \{P_{0_1}^1, P_{0_2}^1, \ldots, P_{0_N}^1\} \Leftrightarrow \omega_1 \gg \omega_q, \forall\ q \in \{2, 3, \ldots, m\} \quad (8)$$

E. Computational Complexity

The computational complexity of our one-shot dynamic optimization methodology is $\mathcal{O}(Nm)$, which is comprised of the intelligent initial parameter value settings for individual application metrics and exploration ordering (Algorithm 1) $\mathcal{O}(Nm)$, and intelligent initial value settings considering all the application metrics $\mathcal{O}(N+m)$, based on the Algorithm 1 calculations (Section III-D). This complexity reveals that our one-shot methodology is lightweight and is thus feasible for sensor nodes with tight resource constraints.

IV. Application Metrics Estimation Model

In this section, we propose an application metric estimation model, which is leveraged by our one-shot dynamic optimization methodology. This estimation model estimates high-level application metrics (lifetime, throughput, reliability) from a sensor node's parameters (e.g., processor voltage and frequency, sensing frequency, transceiver voltage, etc.). For brevity, we describe only the model's key elements. Table I presents a summary of key notations used in our application metrics estimation model.

Table I
Summary of application metrics estimation model notations

Notation	Description
$\mathcal{L}_s$	Lifetime in days
E_b	Battery energy in Joules
E_c	Energy consumption per hour
V_b	Battery voltage in volts
C_b	Battery capacity in mA-h
E_{proc}	Processing energy per hour
E_{com}	Communication energy per hour
E_{sen}	Sensing energy per hour
E_{proc}^a	Processing energy per hour in active mode
E_{proc}^i	Processing energy per hour in idle mode
E_{trans}^{tx}	Transceiver transmission energy per hour
E_{trans}^{rx}	Transceiver receive energy per hour
E_{trans}^i	Transceiver idle energy per hour
N_{pkt}^{tx}	Number of packets transmitted per hour
E_{tx}^{pkt}	Transmission energy per packet
V_t	Transceiver voltage
I_t	Transceiver current
t_{tx}^{pkt}	Time to transmit one packet
I_t^s	Transceiver sleep current
t_{tx}^i	Transceiver idle time per hour
P_s	Packet size in bytes
R_{tx}	Transceiver data rate (in bits/second)
E_{sen}^m	Sensing measurement energy
E_{sen}^i	Sensing idle energy
N_r	Number of sensors on the sensing board
N_s	Number of sensing measurements per second
V_s	Sensing board voltage
I_s^m	Sensing measurement current
t_s^m	Sensing measurement time
I_s	Sensing sleep current
t_s^i	Sensing idle time
R	Aggregate throughput
R_{sen}	Sensing throughput
R_{proc}	Processing throughput
R_{com}	Communication throughput
F_s	Sensing frequency
R_{sen}^b	Sensing resolution bits
F_p	Processor frequency
N^b	Number of instructions to process one bit
t_{tx}^{pkt}	Time to transmit one packet
P_s^{eff}	Effective packet size

A. Lifetime Estimation

The *lifetime* of a sensor node is defined as the time duration between the deployment time and the time before which the sensor node fails to perform the assigned task due to sensor node failure, which is normally caused by battery energy depletion. A sensor node typically contains AA alkaline batteries whose energy depletes gradually as the sensor node consumes energy during operation. The critical factors in determining sensor node lifetime are battery energy and energy consumption during operation.

The sensor node lifetime in days $\mathcal{L}_s$ can be estimated as:

$$\mathcal{L}_s = \frac{E_b}{E_c \times 24} \quad (9)$$

where E_b denotes the sensor node's battery energy (Joules) and E_c denotes the sensor node's energy consumption per hour. The battery energy in mWh $E_b^{'}$ can be given by:

$$E_b^{'} = V_b \cdot C_b \ \ (mWh) \tag{10}$$

where V_b denotes the battery voltage (Volts) and C_b denotes the battery capacity (typically mA-h). Since 1J = 1 Ws, E_b can be calculated as:

$$E_b = E_b^{'} \times 3600/1000 \ \ (J) \tag{11}$$

We model E_c as the sum of the processing, communication, and sensing energies, i.e.,:

$$E_c = E_{proc} + E_{com} + E_{sen} \ \ (J) \tag{12}$$

where E_{proc}, E_{com}, and E_{sen} denote the processing energy per hour, communication energy per hour, and sensing energy per hour, respectively.

The *processing energy* accounts for the processor energy consumed in processing the sensed data. We assume that the sensor node's processor operates in two modes, active mode and idle mode [28]. We point out that although we consider active and idle modes only, a processor operating in other sleep modes (e.g., power-down, power-save, standby, etc.) can also be incorporated in our model. E_{proc} is given by:

$$E_{proc} = E_{proc}^a + E_{proc}^i \tag{13}$$

where E_{proc}^a and E_{proc}^i denote the processor's energy consumption per hour in active mode and idle mode, respectively.

The sensor nodes communicate with each other (e.g., send packets containing the sensed data information) to accomplish the assigned application task and this communication process consumes *communication energy*. The communication energy is the sum of the transmission, receive, and idle energies for a sensor node's transceiver, i.e.,:

$$E_{com} = E_{trans}^{tx} + E_{trans}^{rx} + E_{trans}^i \tag{14}$$

where E_{trans}^{tx}, E_{trans}^{rx}, and E_{trans}^i denote the transceiver's transmission energy per hour, receive energy per hour, and idle energy per hour, respectively. E_{trans}^{tx} is given by:

$$E_{trans}^{tx} = N_{pkt}^{tx} \cdot E_{tx}^{pkt} \tag{15}$$

where N_{pkt}^{tx} denotes the number of packets transmitted per hour and E_{tx}^{pkt} denotes the transmission energy per packet. E_{tx}^{pkt} is given as:

$$E_{tx}^{pkt} = V_t \cdot I_t \cdot t_{tx}^{pkt} \tag{16}$$

where V_t denotes the transceiver voltage, I_t denotes the transceiver current, and t_{tx}^{pkt} denotes the time to transmit one packet. t_{tx}^{pkt} is given by:

$$t_{tx}^{pkt} = P_s \times 8/R_{tx} \tag{17}$$

where P_s denotes the packet size in bytes and R_{tx} denotes the transceiver data rate (in bits/second).

E_{trans}^{rx} can be calculated using a similar procedure as E_{trans}^{tx}. E_{trans}^i can be calculated as:

$$E_{trans}^i = V_t \cdot I_t^s \cdot t_{tx}^i \tag{18}$$

where V_t denotes the transceiver voltage, I_t^s denotes the transceiver sleep current, and t_{tx}^i denotes the transceiver idle time per hour.

The energy consumed by the sensors during sensing the observed phenomenon accounts for the *sensing energy*. The sensing energy mainly depends upon the sensing (sampling) frequency and the number of sensors attached to the sensor board (e.g., the MTS400 sensor board [29] has Sensirion SHT1x temperature and humidity sensors [30]). The sensors consume energy while taking sensing measurements and switch to an idle mode for energy conservation while not sensing. E_{sen} is given by:

$$E_{sen} = E_{sen}^m + E_{sen}^i \tag{19}$$

where E_{sen}^m denotes the sensing measurement energy per hour and E_{sen}^i denotes the sensing idle energy per hour. E_{sen}^m can be calculated as:

$$E_{sen}^m = N_s \cdot V_s \cdot I_s^m \cdot t_s^m \times 3600 \tag{20}$$

where N_s denotes the number of sensing measurements per second, V_s denotes the sensing board voltage, I_s^m denotes the sensing measurement current, and t_s^m denotes the sensing measurement time. E_{sen}^i is given by:

$$E_{sen}^i = V_s \cdot I_s \cdot t_s^i \times 3600 \tag{21}$$

where I_s denotes the sensing sleep current and t_s^i denotes the sensing idle time.

B. Throughput Estimation

In the context of dynamic optimizations, *throughput* can be interpreted as the sensor node's sensing, processing, and transmission rate to observe a phenomenon. Three processes contribute to the sensor node's throughput (i.e., sensing, processing, and communication). The throughput interpretation may vary depending upon the WSN application design as sensing, processing, and communication throughputs can have different relative importance for different applications. The aggregate throughput R (typically measured in bits/second) can be considered as a weighted sum of constituent throughputs:

$$R = \omega_s R_{sen} + \omega_p R_{proc} + \omega_c R_{com} \ : \ \omega_s + \omega_p + \omega_c = 1 \tag{22}$$

where R_{sen}, R_{proc}, and R_{com} denote the sensing, processing, and communication throughputs, respectively. ω_s, ω_p, and ω_c denote the weight factors for the sensing, processing, and communication throughputs, respectively.

The sensing throughput is the throughput due to sensing activity and measures the sensing bits sampled per second. R_{sen} is given by:

$$R_{sen} = F_s \cdot R^b_{sen} \quad (23)$$

where F_s and R^b_{sen} denote the sensing frequency and sensing resolution bits, respectively.

The processing throughput is the throughput due to the processor's processing of sensed measurements and measures the bits processed per second. R_{proc} is given by:

$$R_{proc} = F_p / N^b \quad (24)$$

where F_p and N^b denote the processor frequency and the number of processor instructions to process one bit, respectively.

The communication throughput R_{com} results from the transfer of data packets over the wireless channel and is given by:

$$R_{com} = P^{eff}_s \times 8 / t^{pkt}_{tx} \quad (25)$$

where t^{pkt}_{tx} denotes the time to transmit one packet and P^{eff}_s denotes the effective packet size excluding the packet header overhead.

C. Reliability Estimation

The reliability metric measures the number of packets transferred reliably (i.e., error free packet transmission) over the wireless channel. Accurate reliability estimation is challenging because of dynamic changes in the involved factors, such as network topology, number of neighboring sensor nodes, wireless channel fading, sensor network traffic, etc. The two main factors that affect reliability are the transceiver transmission power P_{tx} and receiver sensitivity. For example, the AT86RF230 transceiver [31] has a receiver sensitivity of -101 dBm with a corresponding packet error rate (PER) $\leq$ 1% for an additive white Gaussian noise (AWGN) channel with a physical service data unit (PSDU) equal to 20 bytes. Reliability can be estimated using Friis free space transmission equation [32] for different P_{tx} values, distance between transmitting and receiving sensor nodes, and fading models (e.g., shadowing fading model). Reliability values can be assigned corresponding to P_{tx} values such that the higher P_{tx} values give higher reliability, however, more accurate reliability estimation requires profiling statistics for the number of packets transmitted and the number of packets received.

V. Experimental Results

In this section, we describe our experimental setup and results for our one-shot dynamic optimization methodology.

Table II
CROSSBOW IRIS MOTE PLATFORM HARDWARE SPECIFICATIONS.

Notation	Description	Value
V_b	Battery voltage	3.6 V
C_b	Battery capacity	2000 mA-h
N_b	Processing instructions per bit	5
R^b_{sen}	Sensing resolution bits	24
V_t	Transceiver voltage	3 V
R_{tx}	Transceiver data rate	250 kbps
I^{rx}_t	Transceiver receive current	15.5 mA
I^s_t	Transceiver sleep current	20 nA
V_s	Sensing board voltage	3 V
I^m_s	Sensing measurement current	550 μA
t^m_s	Sensing measurement time	55 ms
I_s	Sensing sleep current	0.3 μA

A. Experimental Setup

We base our experimental setup on the Crossbow IRIS mote platform [33], which has a battery capacity of 2000 mA-h with two AA alkaline batteries. The IRIS mote platform integrates an Atmel ATmega1281 microcontroller [28], an Atmel AT-86RF230 low power 2.4 GHz transceiver [31], an MTS400 sensor board [29] with Sensirion SHT1x temperature and humidity sensors [30]. Table II shows the sensor node hardware specific values, corresponding to the IRIS mote platform, which are used by the application metrics estimation model [28][30][31][33].

In our experimental setup, we consider a WSN topology where each sensor node has two neighbors, although our topology can be extended for any number of neighboring sensor nodes. The number of neighboring sensor nodes in a topology determines the number of packets received by a sensor node, which affects the expended communication energy. This expended communication energy affects the lifetime of the sensor nodes in the WSN. Our work assumes that the medium access control (MAC) layer handles collisions and packet loss. The packet loss due to any reason (e.g., low transmission power, collision, etc.) is taken into account at a high level by our reliability application metric. The accurate determination of the packet loss requires gathering of profiling statistics, which is the focus of our future work.

We analyze six tunable parameters: processor voltage V_p, processor frequency F_p, sensing frequency F_s, packet size P_s, packet transmission interval P_{ti}, and transceiver transmission power P_{tx}. In order to evaluate our methodology across small and large design spaces, we consider two design space cardinalities (number of states in the design space): $|S| = 729$ and $|S| = 31,104$. The tunable parameters for $|S| = 729$ are V_p = {2.7, 3.3, 4} (volts), F_p = {4, 6, 8} (MHz) [28], F_s = {1, 2, 3} (samples per second) [30], P_s = {41, 56, 64} (bytes), P_{ti} = {60, 300, 600} (seconds), and P_{tx} = {-17, -3, 1} (dBm) [31]. The tunable parameters for $|S| = 31,104$ are V_p = {1.8, 2.7, 3.3, 4, 4.5, 5} (volts), F_p = {2, 4, 6, 8, 12, 16} (MHz) [28], F_s = {0.2, 0.5, 1, 2, 3, 4} (samples per second) [30], P_s = {32, 41, 56, 64, 100, 127} (bytes), P_{ti} = {10, 30, 60, 300, 600, 1200} (seconds), and P_{tx} = {-17, -3, 1, 3} (dBm) [31]. All state space tuples are feasible for $|S| = 729$, whereas $|S| = 31,104$ contains 7,779 infeasible state space tuples (e.g., all V_p and F_p pairs are not feasible). Our consideration of two different design space cardinalities ($|S| = 729$ and $|S| = 31,104$) is important because this consideration helps in investigating the impact of the design space cardinality on dynamic optimization methodologies.

We assign application specific values for the desirable minimum L, desirable maximum U, acceptable minimum α, and acceptable maximum β objective function parameter values for the application metrics (Section III-C). We specify the objective function parameters as a multiple of base units for lifetime, throughput, and reliability, however, our application metrics estimation model and one-shot dynamic optimization methodology works equally well for any set of application requirements, weight factors, and assumption of base units. We assume that one lifetime unit is 5 days, one throughput unit is 20 kbps, and one reliability unit is 0.05 (reliability measures error-free packet transmissions on a scale from 0 to 1). Table III depicts the application requirements for the application domains in terms of objective function parameter values and Table IV depicts the associated weight factors used in our experiments. Weight factors for a given application domain depend upon specific application requirements. For example, a security/defense application that requires prolonged operation requires a higher weight factor for the lifetime application metric as compared to the other application metrics, whereas a different security/defense application that requires gathering high resolution images requires a higher weight factor for the throughput application metric as compared to the other application metrics.

Table III
DESIRABLE MINIMUM L, DESIRABLE MAXIMUM U, ACCEPTABLE MINIMUM α, AND ACCEPTABLE MAXIMUM β OBJECTIVE FUNCTION PARAMETER VALUES FOR A SECURITY/DEFENSE (DEFENSE) SYSTEM, HEALTH CARE, AND AN AMBIENT CONDITIONS MONITORING APPLICATION. ONE LIFETIME UNIT = 5 DAYS, ONE THROUGHPUT UNIT = 20 KBPS, ONE RELIABILITY UNIT = 0.05.

Notation	Defense	Health Care	Ambient Monitoring
L_l	8 units	12 units	6 units
U_l	30 units	32 units	40 units
α_l	1 units	2 units	3 units
β_l	36 units	40 units	60 units
L_t	20 units	19 units	15 unit
U_t	34 units	36 units	29 units
α_t	0.5 units	0.4 units	0.05 units
β_t	45 units	47 units	35 units
L_r	14 units	12 units	11 units
U_r	19.8 units	17 units	16 units
α_r	10 units	8 units	6 units
β_r	20 units	20 units	20 units

Table IV
WEIGHT FACTORS FOR DIFFERENT APPLICATION DOMAINS FOR $|S| = 729$ AND $|S| = 31,104$.

–	$\boldsymbol{|S| = 729}$ & $\boldsymbol{|S| = 31,104}$		
Application Domain	$\boldsymbol{\omega_l}$	$\boldsymbol{\omega_t}$	$\boldsymbol{\omega_r}$
Security/Defense System	0.25	0.35	0.4
Health Care	0.25	0.35	0.4
Ambient Conditions Monitoring	0.4	0.5	0.1

Since the objective function values corresponding to different states depends upon the estimation of high-level metrics, we present an example throughput calculation to explain this estimation process using our application metrics estimation model (Section IV) and the IRIS mote platform hardware specifications (Table II). We consider a state $s_y = (V_{p_y}, F_{p_y}, F_{s_y}, P_{s_y}, P_{ti_y}, P_{tx_y}) = (2.7, 4, 1, 41, 60, -17)$ for our example. (17) gives $t_{tx}^{pkt} = 41 \times 8/(250 \times 10^3) =$ 1.312 ms. (23), (24), and (25) give $R_{sen} = 1 \times 24 = 24$ bps, $R_{proc} = 4 \times 10^6/5 = 800$ kbps, and $R_{com} = 21 \times 8/(1.312 \times 10^{-3}) = 128.049$ kbps, respectively ($P_s^{eff} = 41 - 21 = 20$ where we assume $P_h = 21$ bytes). (22) gives $R = (0.4)(24) + (0.4)(800 \times 10^3) + (0.2)(128.049 \times 10^3) =$ 345.62 kbps where we assume ω_s, ω_p, and ω_c equal to 0.4, 0.4, and 0.2, respectively.

In order to evaluate our one-shot dynamic optimization solution quality, we compare the solution from the one-shot initial parameter settings $\widehat{\boldsymbol{P_0}}$ with the solutions obtained from the following four potential initial parameter value settings (although any feasible n-tuple $s \in S$ can be taken as the initial parameter settings):

- $\mathcal{I}_1$ assigns the first parameter value for each tunable parameter, i.e., $\mathcal{I}_1 = p_{i_1}, \; \forall \, i \in \{1, 2, \ldots, N\}$.
- $\mathcal{I}_2$ assigns the last parameter value for each tunable parameter, i.e., $\mathcal{I}_2 = p_{i_n}, \; \forall \, i \in \{1, 2, \ldots, N\}$.
- $\mathcal{I}_3$ assigns the middle parameter value for each tunable parameter, i.e., $\mathcal{I}_3 = \lfloor p_{i_n}/2 \rfloor, \; \forall \, i \in \{1, 2, \ldots, N\}$.
- $\mathcal{I}_4$ assigns a random value for each tunable parameter, i.e., $\mathcal{I}_4 = p_{i_q} : q = \texttt{rand()} \% n, \; \forall \, i \in \{1, 2, \ldots, N\}$ where `rand()` denotes a function to generate a random/pseudo-random integer and % denotes the modulus operator.

Although we analyzed our methodology for the IRIS motes platform, three application domains, two design spaces, and four potential initial parameter value settings, our one-shot dynamic optimization methodology and application metrics estimation model are equally applicable to any platform, application domain, and design space.

B. Results

We implemented our one-shot dynamic optimization methodology in C++. To evaluate the effectiveness of our one-shot solution, we compare the one-shot solution's results with four alternative initial parameter arrangements (Section V-A). We normalize the objective function values corresponding to the operating states attained by our dynamic optimization methodology with respect to the optimal solution obtained using an exhaustive search. We compare the relative complexity of our one-shot dynamic optimization methodology with two other dynamic optimization methodologies, which leverage greedy- and SA-based algorithms for design space exploration [34]. Although for brevity we present results for only a subset of the initial parameter value settings, application domains, and design spaces, we observed that results for extensive application domains, design spaces, and initial parameter settings revealed similar trends.

1) Percentage Improvements over other Initial Parameter Settings: Table V depicts the percentage improvements attained by the one-shot parameter settings $\widehat{\boldsymbol{P_0}}$ over other parameter settings for different application domains and weight factors (Table IV). We point out that different weight factors could result in different percentage improvements, however, we observed similar trends for other weight factors. Table V shows that the one-shot initial parameter settings can result in as high as a 155% improvement as compared to other initial value settings. We observe that some arbitrary settings may give a comparable or even a better solution for a particular application domain, application metric weight factors, and design space cardinality, but that arbitrary setting would not scale to other application domains, application metric weight factors, and design space cardinalities. For example, $\mathcal{I}_3$ obtains a 12% better quality solution than $\widehat{\boldsymbol{P_0}}$ for the ambient conditions monitoring application for $|S| = 31,104$, but yields a 10% lower quality solution for the security/defense and health care applications for $|S| = 31,104$, and a 57%, 31%, and 20% lower quality solution than $\widehat{\boldsymbol{P_0}}$ for the security/defense, health care, and ambient conditions monitoring applications, respectively, for $|S| = 729$. The percentage improvement attained by $\widehat{\boldsymbol{P_0}}$ over all application domains and design spaces is 33% on average. Our one-shot methodology is the first approach (to the best of our knowledge) to leverage intelligent initial tunable parameter value settings for sensor nodes to provide a good quality operating state, as arbitrary initial parameter value settings typically result in a poor operating state. Results reveal that on average $\widehat{\boldsymbol{P_0}}$ gives a solution within 8% of the optimal solution.

2) Comparison with Greedy- and SA-based Dynamic Optimization Methodologies: In order to investigate the effectiveness of our one-shot methodology, we compare the one-shot solution's quality (indicated by the attained objective function value) with two other dynamic optimization methodologies, which leverage an SA-based and a greedy-based (denoted by GD^{asc} where asc stands for ascending order of parameter exploration) exploration of the design space. We assign initial parameter value settings for the greedy- and SA-based methodologies as $\mathcal{I}_1$ and $\mathcal{I}_4$, respectively. Note that, for brevity, we present results for $\mathcal{I}_1$ and $\mathcal{I}_4$, however, other initial parameter settings such as $\mathcal{I}_2$ and $\mathcal{I}_3$ would yield similar trends when combined with greedy-based and SA-based design space exploration.

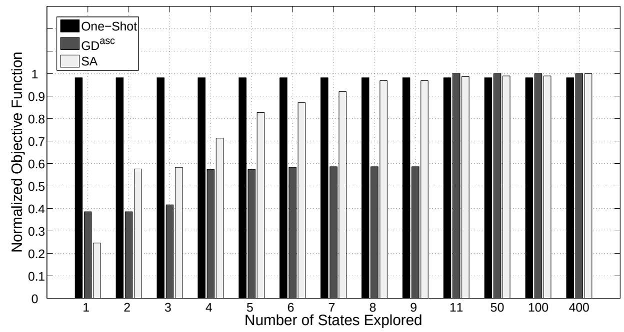


Figure 3. Objective function value normalized to the optimal solution for a varying number of states explored for the one-shot, greedy, and SA algorithms for a security/defense system where $\omega_l = 0.25$, $\omega_t = 0.35$, $\omega_r = 0.4$, $|S| = 729$.

Fig. 3 shows the objective function value normalized to the optimal solution versus the number of states explored for the one-shot, GD^{asc}, and SA algorithms for a security/defense system for $|S| = 729$. The one-shot solution is within 1.8% of the optimal solution. The figure shows that GD^{asc} and SA explore 11 states (1.51% of the design space) and 10 states (1.37% of the design space), respectively, to attain an equivalent or better quality solution than the one-shot solution. Although, greedy- and SA-based methodologies explore few states to reach a comparable solution as that of our one-shot methodology, the one-shot methodology is suitable when design space exploration is not an option due to an extremely large design space and/or extremely stringent computational, memory, and timing constraints. These results indicate that other arbitrary initial value settings (e.g., $\mathcal{I}_1$, $\mathcal{I}_4$, etc.) do not provide a good quality operating state and necessitate design space exploration by online algorithms (e.g., greedy) to provide a good quality operating state. We point out that if the greedy- and SA-based methodologies leverage our one-shot initial tunable parameter value settings $\mathcal{I}$, further improvements over the one-shot solution can produce a very good quality (optimal or near-optimal) operating state [34].

Fig. 4 shows the objective function value normalized to the optimal solution versus the number of states explored for a security/defense system for $|S| = 31,104$. The one-shot solution is within 8.6% of the optimal solution. The figure

Table V
PERCENTAGE IMPROVEMENTS ATTAINED BY $\widehat{\boldsymbol{P_0}}$ OVER OTHER INITIAL PARAMETER SETTINGS FOR $|S| = 729$ AND $|S| = 31,104$.

–	$\boldsymbol{\vert S\vert = 729}$				$\boldsymbol{\vert S\vert = 31,104}$			
Application Domain	$\mathcal{I}_1$	$\mathcal{I}_2$	$\mathcal{I}_3$	$\mathcal{I}_4$	$\mathcal{I}_1$	$\mathcal{I}_2$	$\mathcal{I}_3$	$\mathcal{I}_4$
Security/Defense System	155%	10%	57%	29%	148%	0.3%	10%	92%
Health Care	78%	7%	31%	11%	73%	0.3%	10%	45%
Ambient Conditions Monitoring	52%	6%	20%	7%	15%	-7%	-12%	18%

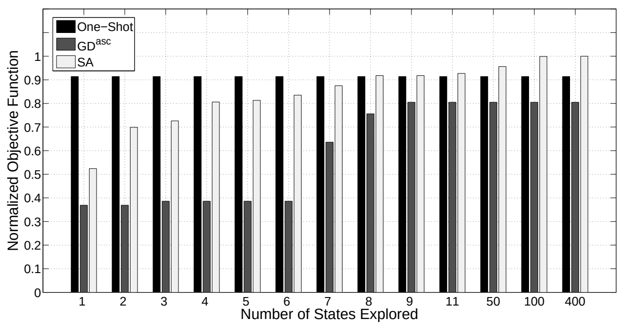


Figure 4. Objective function value normalized to the optimal solution for a varying number of states explored for the one-shot, greedy, and SA algorithms for a security/defense system where $\omega_l = 0.25$, $\omega_t = 0.35$, $\omega_r = 0.4$, $|S| = 31,104$.

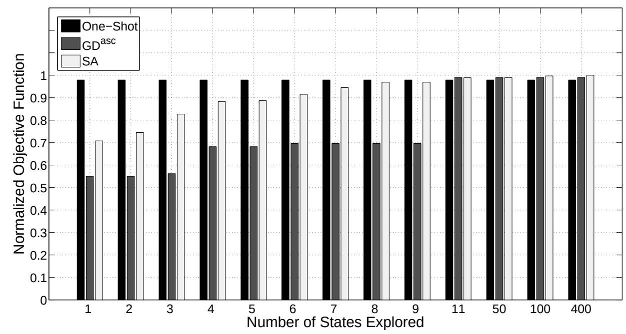


Figure 5. Objective function value normalized to the optimal solution for a varying number of states explored for the one-shot, greedy, and SA algorithms for a health care application where $\omega_l = 0.25$, $\omega_t = 0.35$, $\omega_r = 0.4$, $|S| = 729$.

shows that GD^{asc} converges to a lower quality solution than the one-shot solution after exploring 9 states (0.029% of the design space) and SA explores 8 states (0.026% of the design space) to yield a better quality solution than the one-shot solution. These results reveal that the greedy exploration of parameters may not necessarily attain a better quality solution than our one-shot solution.

Fig. 5 shows the objective function value normalized to the optimal solution versus the number of states explored for a health care application for $|S| = 729$. The one-shot solution is within 2.1% of the optimal solution. The figure shows that GD^{asc} converges to an almost equal quality solution as compared to the one-shot solution after exploring 11 states (1.5% of the design space) and SA explores 10 states (1.4% of the design space) to yield an almost equal quality solution as compared to the one-shot solution. These results indicate that further exploration of the design space is required to find an equivalent quality solution as compared to one-shot if the intelligent initial value settings leveraged by one-shot are not used.

Fig. 6 shows the objective function value normalized to the optimal solution versus the number of states explored for a health care application for $|S| = 31,104$. The one-shot solution is within 1.6% of the optimal solution. The figure shows that GD^{asc} converges to a lower quality solution than the one-shot solution after exploring 9 states (0.029% of the design space) and SA explores 6 states (0.019% of the design space) to yield a better quality solution than the one-shot solution. These results confirm that the greedy exploration of the parameters may not necessarily attain a better quality solution than our one-shot solution.

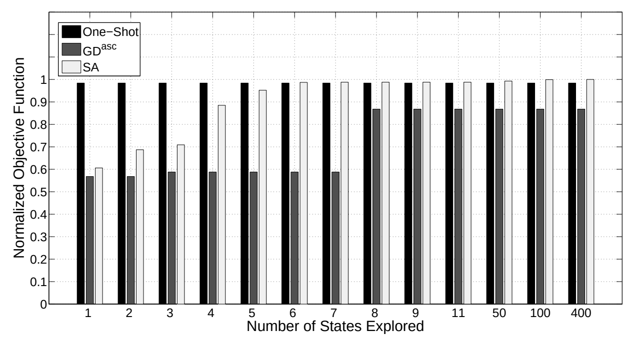


Figure 6. Objective function value normalized to the optimal solution for a varying number of states explored for the one-shot, greedy, and SA algorithms for a health care application where $\omega_l = 0.25$, $\omega_t = 0.35$, $\omega_r = 0.4$, $|S| = 31,104$.

Fig. 7 shows the objective function value normalized to the optimal solution versus the number of states explored for an ambient conditions monitoring application for $|S| = 729$. The one-shot solution is within 7.7% of the optimal solution. The figure shows that GD^{asc} and SA converge to an equivalent or better quality solution than the one-shot solution after exploring 4 states (0.549% of the design space)

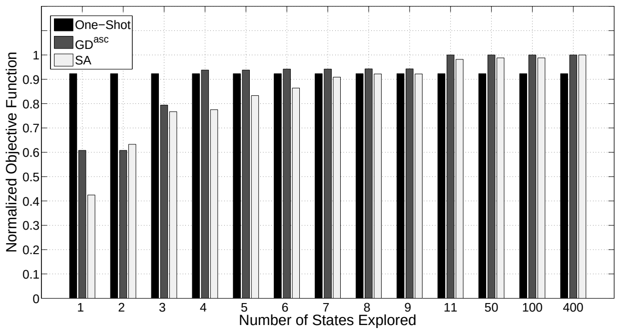


Figure 7. Objective function value normalized to the optimal solution for a varying number of states explored for the one-shot, greedy, and SA algorithms for an ambient conditions monitoring application where $\omega_l = 0.4$, $\omega_t = 0.5$, $\omega_r = 0.1$, $|S| = 729$.

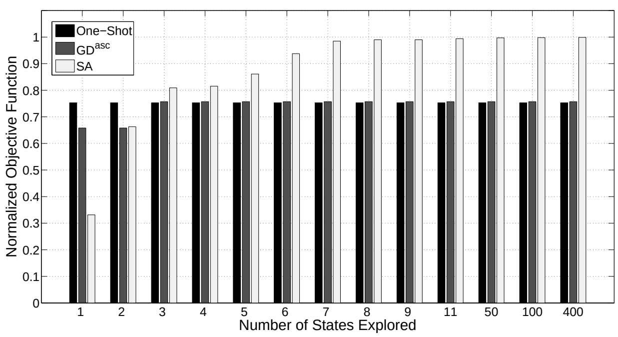


Figure 8. Objective function value normalized to the optimal solution for a varying number of states explored for the one-shot, greedy, and SA algorithms for an ambient conditions monitoring application where $\omega_l = 0.4$, $\omega_t = 0.5$, $\omega_r = 0.1$, $|S| = 31,104$.

and 10 states (1.37% of the design space), respectively. These results again confirm that the greedy- and SA-based explorations can provide improved results over the one-shot solution, but require additional state exploration.

Fig. 8 shows the objective function value normalized to the optimal solution versus the number of states explored for an ambient conditions monitoring application for $|S| = 31,104$. The one-shot solution is within 24.7% of the optimal solution. The figure shows that both GD^{asc} and SA converge to an equivalent or better quality solution than the one-shot solution after exploring 3 states (0.01% of the design space). These results indicate that both greedy- and SA-based methods can give good quality solutions after exploring a very small percentage of the design space and both greedy- and SA-based methods enable lightweight dynamic optimizations [34]. The results also indicate that the one-shot solution provides a good quality solution when further design space exploration is not possible due to resource constraints.

3) Computational Complexity: To verify that our one-shot dynamic optimization methodology (Section III) is lightweight, we compared the data memory requirements and execution time of our one-shot dynamic optimization methodology with the greedy- and SA-based dynamic optimization methodologies.

The data memory analysis revealed that our one-shot methodology requires only 150, 188, 248, and 416 bytes for (number of tunable parameters N, number of application metrics m) equal to (3, 2), (3, 3), (6, 3), and (6, 6), respectively. The greedy-based methodology requires 458, 528, 574, 870, and 886 bytes, whereas the SA-based methodology requires 514, 582, 624, 920, and 936 bytes of storage for design space cardinalities of 8, 81, 729, 31,104, and 46,656, respectively. The data memory analysis shows that the SA-based methodology has comparatively larger memory requirements than the greedy-based methodology. Our analysis reveals that the data memory requirements for our one-shot methodology increases linearly as the number of tunable parameters and the number of application metrics increases. The data memory requirements for the greedy- and SA-based methodologies increase linearly as the number of tunable parameters and tunable values (and thus the design space) increases. The data memory analysis verifies that although the one-shot, greedy- and SA-based methodologies have low data memory requirements (on the order of hundreds of bytes), the one-shot solution requires 204% and 458% less memory on average as compared to the greedy- and SA-based methodologies, respectively.

We measured the execution time for our one-shot and the greedy- and SA-based methodologies averaged over 10,000 runs (to smooth any discrepancies in execution time due to operating system overheads) on an Intel Xeon CPU running at 2.66 GHz [35] using the Linux/Unix `time` command [36]. We scaled the execution time results to the Atmel ATmega1281 microcontroller [28] running at 8 MHz. Although microcontrollers have different instruction set architectures and scaling does not provide 100% accuracy for the microcontroller runtime, scaling enables relative comparisons and provides reasonable runtime estimates. Results showed that one-shot required 1.66 ms both for $|S| = 729$ and $|S| = 31,104$. GD^{asc} explored 10 states and required 0.887 ms and 1.33 ms on average to converge to the solution for $|S| = 729$ and $|S| = 31,104$, respectively. SA took 2.76 ms and 2.88 ms to explore the first 10 states (to provide a fair comparison with GD^{asc}) for $|S| = 729$ and $|S| = 31,104$, respectively. The execution time analysis revealed that our dynamic optimization methodologies required execution times on the order of milliseconds, and the one-shot solution required 18% less execution time on average as compared to greedy- and SA-based methodologies. The one-shot solution required 66% and 73% less execution time for the SA-based methodology when $|S| = 729$ and $|S| = 31,104$. These results indicate that the design space cardinality affects the execution time linearly for greedy- and SA-based methodologies whereas the one-shot solution's execution time is affected negligibly by the design space cardinality and hence our one-shot methodology's advantage increases as the design space cardinality increases. We verified our execution time analysis using the `clock()` function [37], which revealed similar trends.

VI. Conclusions and Future Work

In this paper, we proposed a lightweight dynamic optimization methodology for WSNs, which provided a high-quality solution in just one-shot using intelligent initial tunable parameter value settings for highly constrained applications. To assist dynamic optimization methodologies for operating states' comparisons, we proposed an application metric estimation model to estimate high-level metrics (lifetime, throughput, and reliability) from sensor node's parameters. This estimation model was leveraged by our one-shot dynamic optimization methodology and provided a prototype model for application metric estimation. To evaluate the effectiveness of the initial parameter settings by our one-shot methodology, we compared the one-shot solution quality with four other typical initial parameter settings. Results revealed that the percentage improvement attained by our one-shot solution over other initial parameter settings for different application domains and design spaces was 33% on average and as high as 155%. Results indicated that our one-shot solution was within 8% of the optimal solution obtained from exhaustive search. We compared the computational complexity of our one-shot dynamic optimization methodology with two other dynamic optimization methodologies that leveraged greedy- and simulated annealing (SA)-based exploration of the design space. Results showed that the one-shot solution required 204% and 458% less memory on average as compared to the greedy- and SA-based methodologies, respectively. The one-shot solution required 18% less execution time on average as compared to the greedy- and SA-based methodologies even if these methodologies were restricted to explore only 0.03% of the design space on average. The execution time and data memory analysis confirmed that our one-shot methodology is lightweight and suitable for time-critical or highly constrained applications.

Future work includes the incorporation of profiling statistics into our one-shot dynamic optimization methodology to provide feedback with respect to changing environmental stimuli. We plan to further verify our one-shot dynamic optimization methodology using implementation on a hardware sensor node platform. We also plan to further investigate online optimization algorithms leveraging our one-shot initial value settings for further higher quality solutions for comparatively less constrained applications.

Acknowledgments

This work was supported by the National Science Foundation (NSF) (CNS-0834080) and the Natural Sciences and Engineering Research Council of Canada (NSERC). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the NSF and the NSERC.

References

[1] A. Munir, A. Gordon-Ross, S. Lysecky, and R. Lysecky, "A One-Shot Dynamic Optimization Methodology for Wireless Sensor Networks," in *Proc. IARIA IEEE International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM)*, Florence, Italy, October 2010.

[2] D. Brooks and M. Martonosi, "Value-based Clock Gating and Operation Packing: Dynamic Strategies for Improving Processor Power and Performance," *ACM Trans. on Computer Systems*, vol. 18, no. 2, pp. 89–126, May 2000.

[3] S. Patel and S. Lumetta, "rePLay: A Hardware Framework for Dynamic Optimization," *IEEE Trans. on Computers*, vol. 50, no. 6, pp. 590–608, June 2001.

[4] C. Zhang, F. Vahid, and R. Lysecky, "A Self-Tuning Cache Architecture for Embedded Systems," *ACM Trans. on Embedded Computing Systems*, vol. 3, no. 2, pp. 407–425, May 2004.

[5] K. Hazelwood and M. Smith, "Managing Bounded Code Caches in Dynamic Binary Optimization Systems," *ACM Trans. on Architecture and Code Optimization*, vol. 3, no. 3, pp. 263–294, September 2006.

[6] S. Hu, M. Valluri, and L. John, "Effective Management of Multiple Configurable Units using Dynamic Optimization," *ACM Trans. on Architecture and Code Optimization*, vol. 3, no. 4, pp. 477–501, December 2006.

[7] (2012, January) Dynamic Profiling and Optimization (DPOP) for Sensor Networks. [Online]. Available: http://www.ece.arizona.edu/~dpop/

[8] A. Gordon-Ross, F. Vahid, and N. Dutt, "Fast Configurable-Cache Tuning With a Unified Second-Level Cache," *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, vol. 17, no. 1, pp. 80–91, January 2009.

[9] C.-Y. Seong and B. Widrow, "Neural Dynamic Optimization for Control Systems," *IEEE Trans. on Systems, Man, and Cybernatics*, vol. 31, no. 4, pp. 482–489, August 2001.

[10] H. Hamed, A. El-Atawy, and A.-S. Ehab, "On Dynamic Optimization of Packet Matching in High-Speed Firewalls," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 10, pp. 1817–1830, October 2006.

[11] S. Sridharan and S. Lysecky, "A First Step Towards Dynamic Profiling of Sensor-Based Systems," in *Proc. IEEE Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON'08)*, San Francisco, California, June 2008, pp. 600–602.

[12] A. Shenoy, J. Hiner, S. Lysecky, R. Lysecky, and A. Gordon-Ross, "Evaluation of Dynamic Profiling Methodologies for Optimization of Sensor Networks," *IEEE Embedded Systems Letters*, vol. 2, no. 1, pp. 10–13, March 2010.

[13] A. Munir and A. Gordon-Ross, "An MDP-based Application Oriented Optimal Policy for Wireless Sensor Networks," in *Proc. ACM International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS'09)*, Grenoble, France, October 2009, pp. 183–192.

[14] S. Kogekar, S. Neema, B. Eames, X. Koutsoukos, A. Ledeczi, and M. Maroti, "Constraint-Guided Dynamic Reconfiguration in Sensor Networks," in *Proc. ACM International Symposium on Information Processing in Sensor Networks (IPSN'04)*, Berkeley, California, April 2004, pp. 379–387.

[15] X. Wang, J. Ma, S. Wang, and D. Bi, "Distributed Energy Optimization for Target Tracking in Wireless Sensor Networks," *IEEE Trans. on Mobile Computing*, vol. 9, no. 1, pp. 73–86, January 2009.

[16] L. Liu, X. Zhang, and H. Ma, "Dynamic Node Collaboration for Mobile Target Tracking in Wireless Camera Sensor Networks," in *Proc. IEEE (INFOCOM'09)*, Rio de Janeiro, Brazil, April 2009, pp. 1188–1196.

[17] R. Khanna, H. Liu, and H.-H. Chen, "Dynamic Optimization of Secure Mobile Sensor Networks: A Genetic Algorithm," in *Proc. IEEE International Conference on Communications (ICC'07)*, Glasgow, Scotland, June 2007, pp. 3413–3418.

[18] R. Min, T. Furrer, and A. Chandrakasan, "Dynamic Voltage Scaling Techniques for Distributed Microsensor Networks," in *Proc. IEEE Workshop on VLSI (WVLSI'00)*, Orlando, Florida, April 2000, pp. 43–46.

[19] L. Yuan and G. Qu, "Design Space Exploration for Energy-Efficient Secure Sensor Network," in *Proc. IEEE International Conference on Application-Specific Systems, Architectures, and Processors (ASAP'02)*, San Jose, California, July 2002, pp. 88–97.

[20] S. Lysecky and F. Vahid, "Automated Application-Specific Tuning of Parameterized Sensor-Based Embedded System Building Blocks," in *Proc. of the International Conference on Ubiquitous Computing (UbiComp)*, Orange County, California, September 2006, pp. 507–524.

[21] R. Verma, "Automated application specific sensor network node tuning for non-expert application developers," Master's thesis, Department of Electrical and Computer Engineering, University of Arizona, 2008.

[22] R. Mannion, H. Hsieh, S. Cotterell, and F. Vahid, "System Synthesis for Networks of Programmable Blocks," in *Proc. IEEE Conference on Design, Automation and Test in Europe (DATE)*, Munich, Germany, March 2005, pp. 888–893.

[23] A. Meier, M. Weise, J. Beutel, and L. Thiele, "NoSE: Efficient Initialization of Wireless Sensor Networks," in *Proc. ACM Conference on Embedded Networked Sensor Systems (SenSys'08)*, Raleigh, North Carolina, November 2008, pp. 397–398.

[24] D. Ma, J. Wang, M. Somasundaram, and Z. Hu, "Design and Optimization on Dynamic Power System for Self-Powered Integrated Wireless Sensing Nodes," in *Proc. IEEE International Symposium on Low Power Electronics and Design (ISLPED'05)*, San Diego, California, August 2005, pp. 303–306.

[25] R. Jurdak, P. Baldi, and C. Lopes, "Adaptive Low Power Listening for Wireless Sensor Networks," *IEEE Trans. on Mobile Computing*, vol. 6, no. 8, pp. 988–1004, August 2007.

[26] M. Hasegawa, T. Kawamura, N. Tran, G. Miyamoto, Y. Murata, H. Harada, and S. Kato, "Decentralized Optimization of Wireless Sensor Network Lifetime based on Neural Network Dynamics," in *Proc. IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'08)*, Cannes, France, September 2008, pp. 1–5.

[27] X. Ning and C. Cassandras, "Optimal Dynamic Sleep Time Control in Wireless Sensor Networks," in *Proc. IEEE Conference on Decision and Control (CDC'08)*, Cancun, Mexico, December 2008, pp. 2332–2337.

[28] Atmel. (2012, January) Atmel atmega1281 microcontroller with 256k bytes in-system programmable flash. [Online]. Available: http://www.atmel.com/dyn/resources/prod_documents/2549S.pdf

[29] Crossbow. (2012, January) MTS/MDA sensor board users manual. [Online]. Available: http://www.xbow.com/

[30] Sensirion. (2012, January) Datasheet sht1x (sht10, sht11, sht15) humidity and temperature sensor. [Online]. Available: http://www.sensirion.com/

[31] Atmel. (2012, January) Atmel at86rf230 low power 2.4 ghz transceiver for zigbee, ieee 802.15.4, 6lowpan, rf4ce and ism applications. [Online]. Available: http://www.atmel.com/dyn/resources/prod_documents/doc5131.pdf

[32] H. Friis, "A Note on a Simple Transmission Formula," *Proc. IRE*, vol. 34, p. 254, 1946.

[33] Crossbow. (2012, January) Crossbow iris datasheet. [Online]. Available: http://www.xbow.com/

[34] A. Munir, A. Gordon-Ross, S. Lysecky, and R. Lysecky, "A Lightweight Dynamic Optimization Methodology for Wireless Sensor Networks," in *Proc. IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, Niagara Falls, Canada, October 2010, pp. 129–136.

[35] (2012, January) Intel Xeon Processor E5430. [Online]. Available: http://processorfinder.intel.com/details.aspx?sSpec=SLANU

[36] (2012, January) Linux Man Pages. [Online]. Available: http://linux.die.net/man/

[37] (2012, January) C++ reference library. [Online]. Available: http://cplusplus.com/reference/clibrary/ctime/clock/

REST-Event: A REST Web Service Framework for Building Event-Driven Web

Li Li
Avaya Labs Research
Avaya Inc.
Basking Ridge, New Jersey, USA
lli5@avaya.com

Wu Chou
Avaya Labs Research
Avaya Inc.
Basking Ridge, New Jersey, USA
wuchou@avaya.com

***Abstract*—As the World Wide Web is becoming a communication and collaboration platform, there is an acute need for an infrastructure to disseminate real-time events over the Web. However, such infrastructure is still seriously lacking as conventional distributed event-based systems are not designed for the Web. To address this issue, we describe a REST web service framework, REST-Event. It represents and organizes the concepts and elements of Event-Driven Architecture (EDA) as REST (Representational State Transfer) web services. Our approach leads to a layered event-driven web, in which event actors, subscriptions and event channels are separated. As an integration framework, REST-Event specifies a set of minimal REST services to support event systems, such that generic two-way event channels can be created and managed seamlessly through a process called subscription entanglement. A special form of event-driven web, called topic web, is proposed and built based on REST-Event. The advantages and applications of topic web are presented and discussed, including addressability, connectedness, dynamic topology, robustness and scalability. In addition, a prototype topic web for presence driven collaboration is developed. Preliminary performance tests show that the proposed approach is feasible and advantageous.**

Keywords - Web service, REST, Topic Hubs, Event-driven, EDA.

I. INTRODUCTION

The Web has undergone a rapid evolution from an informational space of static documents to a space of dynamic communication and collaboration. In the early days of Web, changes to web content were infrequent and a user could rely on web portals, private bookmarks, or search engines to find information and follow them. However, in the era of social media, information updates become frequent and rapid. People need timely and almost instant availability of these dynamic contents to interactively use this information without being overwhelmed by the information overload. This demands the Web to evolve rapidly from a static and reactive informational space to a dynamic communication and collaboration oriented environment. As a consequence, this migration of Web can affect the application spaces of both consumers and enterprises for future services. The trend of a communication and collaboration Web pushes for an event-driven web, in which information sharing is driven by asynchronous events to support dynamic, real-time, or near real-time information exchange.

Despite many existing event notification systems developed over the years, infrastructures and technologies for such an event-driven web are still seriously lacking for the following reasons.

First, most of the architectures, protocols, and programming languages for conventional distributed event notification systems were developed prior to the Web. As a result, these notification systems are not accessible to each other on the Web or fit the infrastructure of the Web.

Secondly, the current web technologies related to event notifications, including Atom [4][5], Server-Sent Events [9], Web Sockets [10], Bidirectional HTTP [33] and HTML5 [8], focus mainly on client-server interactions and are not sufficient to support integrations between web sites and web applications across organizational boundaries.

Therefore, there is an acute need for a unifying framework that can provide seamless integration of these notification systems with the infrastructure of web and web based services. Such a unifying framework can transform conventional notification systems into web services such that they become part of the Web. It can also be used to integrate and enable the existing web based applications, including those social network sites, which currently do not have a way to share events. If these two goals can be achieved effectively, then it could lead to a nested event notification system on the Web - an event-driven web extension to the current Web.

To develop such a unifying framework, we lay our foundation on Event-Driven Architecture (EDA) [12], in which information is encapsulated as asynchronous events propagated to the interested components when they occur. EDA defines the principles and architecture for event discovery, subscription, delivery and reaction, which are also key components in event-driven web for real-time communication and collaboration. Moreover, EDA is a natural fit for the event-driven web as both architectures assume a distributed system that are developed and maintained independently by different organizations without any centralized control.

To apply EDA to the web architecture, we represent and organize EDA concepts and elements as REST [1][2] web services in a framework called REST-Event. As an integration framework, REST-Event demands a set of minimal REST services supported by the systems to be integrated but at the same time supports different event channels between the systems. For this reason, we generalize the traditional one-way event channels, in which event notifications flow in one direction to two-way event channels, in which event notifications can flow in both directions. To hide complexity of managing two-way event channels, we introduce a process called subscription entanglement based on REST protocols. With this framework, the event-driven web can be built and operated as a distributed hypermedia system, for which REST is optimized.

By projecting EDA to REST, many important problems in conventional event notification systems can be resolved efficiently. The uniform interface, connectedness, and addressability of REST can facilitate the discovery of notification web services. The idempotent operations and statelessness of REST can enhance robustness and scalability to notification web services. Subscription entanglement hides system complexity from the clients.

To test REST-Event framework, a prototype event-driven web, topic web, which consists of distributed topic hubs, is implemented using REST-Event to demonstrate the feasibility and advantages of this approach.

The rest of the paper is organized as follows. Section II introduces the background and related work. Section III describes our vision of event-driven web. Section IV introduces the REST-Event framework. Section V describes a special event-driven web called topic web that can be built from REST-Event. Section VI summarizes the advantages of the topic web. Section VII is dedicated to a prototype implementation and experimental study results. Findings of this paper are summarized in Section VIII.

II. RELATED WORK

This paper extends our previous work [1] published in Service Computation 2010 in the following aspects: 1) the new framework is based on a new layered system with generalized event channels, whereas our previous work assumes all event channels are HTTP; 2) the new framework supports creation of two-way event channels in one transaction, whereas the previous framework only supports one-way event channels; 3) this paper clarifies the notions of entangled subscriptions; 4) the core resources and protocols of the framework are separated from the topic web, which is a system built from the described framework.

REST stands for REpresentational State Transfer. It is an architecture style optimized for distributed hypermedia system as described in [2][3][4]. The fundamental constraint of REST is that the interactions between a client and servers should be driven by hypermedia. In other words, a client should be able to start from a single URI and transition to a desired state by following the links in the hypermedia provided by the servers. This constraint is realized by the following architectural constraints: 1) Addressability: each resource can be addressed by URI. 2) Connectedness: resources are linked to enable transitions. 3) Uniform Interface: all components in the system support the same interface, namely HEAD, GET, PUT, DELETE and POST. HEAD and GET are safe and idempotent. PUT and DELETE are not safe but idempotent. Idempotent operations can be executed many times by a server and have the same effect as being executed once. This property allows a client to resubmit a request in case of failures without worrying about undesired side-effect, such as paying something twice. 4) Statelessness: all requests to a resource contain all information necessary to process the requests, and the servers do not need to keep any context in order to process the requests. Stateless servers have much less failure conditions than stateful ones and are easy to scale and migrate. 5) Layering: intermediate proxies between clients and servers can be used to cache responses, enforce security polices, or distribute workloads.

RSS [7] and Atom [5] are two data formats that describe web feeds to be consumed by feed readers. A feed can be news, blogs, wikis, or any resource whose content may be updated frequently by the content providers. The content providers are responsible to publish their feeds. This is usually done by embedding the feed URI in a web page. The feed readers are responsible to find the feeds, for example by following feed URI. Once a feed is found, the feed reader fetches the updates by periodically polling the feed. However, such polling is very inefficient in general, because the timing of the updates is unpredictable. Polling too frequently may waste a lot of network bandwidth, when there is no update. On the other hand, polling too infrequently may miss some important updates and incur delay on information processing.

To address the inefficiency of poll style feed delivery, Google developed a topic based subscription protocol called PubSubHubbub [23]. In this protocol, a hub web server acts as a broker between feed publishers and subscribers. A feed publisher indicates in the feed document (Atom or RSS) its hub URL, to which a subscriber (a web server) can registers a listener. Whenever there is an update, the feed publisher notifies its hub, which then fetches the feed and multicasts (push) it to the registered listeners. While this protocol enables more efficient push style feed updates, it does not describe how hubs can be federated to provide a global feed update service across different web sites. This protocol only supports one-way event channels from a topic hub to its listeners. The event channels are also fixed to be Atom over HTTP. Also the system does not use subscription entanglement to manage event channels.

Many techniques have been developed over the years to address the asynchronous event delivery to the web browsers, such as Ajax, Pushlet [8], and most recently Server-Sent Events [10] and Web Sockets [11]. However, these techniques are not applicable to federated notification services where server to server relations and communication protocols are needed.

Bayeux [34] is a protocol that supports both HTTP long-polling and streaming mechanisms to allow a HTTP server to push notifications to a HTTP client. This protocol can be combined with normal HTTP request/response to support two-way event channels. But this protocol does not specify how to create and manage subscriptions.

BOSH (XEP-0124) [35] uses HTTP long-polling to emulate bidirectional TCP streams. XMPP also supports a publish/subscribe extension (XEP-0060) [36] to allow XMPP entities to subscribe and publish events to topics. But these protocols do not specify how to create and manage subscriptions.

Server-Sent Events [10] defines a protocol that uses HTTP streaming to allow a HTTP server to push notifications to a HTTP client. This protocol can be combined with normal HTTP request/response to create two-way event channels. However, the protocol does not specify how to create or manage subscriptions.

Google Wave is a platform and protocol to provide near real-time communication and collaboration between web browsers. Since Google Wave Federation Protocol [37] is based on XMPP, it does not support integration with the Web directly. Google Wave Client-Server Protocol [38] is based on WebSockets and JSON. The protocol does not specify how to create subscriptions.

Microsoft Azure cloud platform [39][40] has several built-in mechanism to support EDA (Event-Driven Architecture), including server-to-server event subscriptions, for example between an event queue and a router. But the platform does not support two-way event channels through subscription entanglement.

WIP [41] uses WS-Eventing to negotiate media transports for IP based multimedia communication. The basic idea in WIP is to treat media streams as two-way events and WS-Eventing is used to negotiate the media transport parameters. Although WIP supports a form of two-way event channels, there are some significant differences. First, WIP is based on WS-* instead of REST. Second, WIP is aimed to establish media transports (RTP) between two endpoints, whereas REST-Event is aimed to integrate different notification systems.

In software engineering, Publisher-Subscriber [16] or Observer [12] is a well-known software design pattern for keeping the states of cooperative components or systems synchronized by event propagation. It is widely used in event-driven programming for GUI applications. This pattern has also been standardized in several industrial efforts for distributed computing, including Java Message Service (JMS) [25], CORBA Event Service [26], CORBA Notification Service [26], which are not based on web services.

Recently, two event notification web services standards, WS-Eventing [19] and WS-Notification [20][21] are developed. However, these standards are not based on REST. Instead they are based on WSDL [28] and SOAP [29], which are messaging protocols alternative to REST [1]. WS-Topic [22] is an industrial standard to define a topic-based formalism for organizing events. However, these topics are not REST resources but are XML elements in some documents.

Recently, much attention has been given to Event-Driven Architecture (EDA) [13][17] and its interaction with Service-Oriented Architecture (SOA) [18] to enable agile and responsive business processes within enterprises. The fundamental ingredients of EDA are the following actors: event publishers that generate events, event listeners that receive events, event processors that analyze events and event reactors that respond to events. The responses may cause more events to occur, such that these actors form a closed loop.

A comprehensive review on the issues, formal properties and algorithms for the state-of-the-art event notification systems is provided in [14]. The system model of the notification services is based on an overlay network of event brokers, including those based on DHT [15]. There are two types of brokers: the inner brokers that route messages and the border brokers that interact with the event producers and listeners. A border broker provides an interface for clients to subscribe, unsubscribe, advertise, and publish events. An event listener is responsible to implement a notify interface in order to receive notifications. However, none of the existing notification systems mentioned in [14] is based on RESTful web services.

III. EVENT-DRIVEN WEB

To project EDA to REST, we model the EDA concepts, such as subscription, publisher, listener and broker, as interconnected resources that support the uniform interface of REST. As the result, a distributed event notification system becomes the event-driven web: a web of resources represented as distributed hypermedia that propagates and responds to events as envisioned by EDA. There is no longer any boundary between different event notification systems as they can expose their interfaces through these resources and become part of the event-driven web.

The event-driven web is a layered system with the following layers as shown in Figure 1.

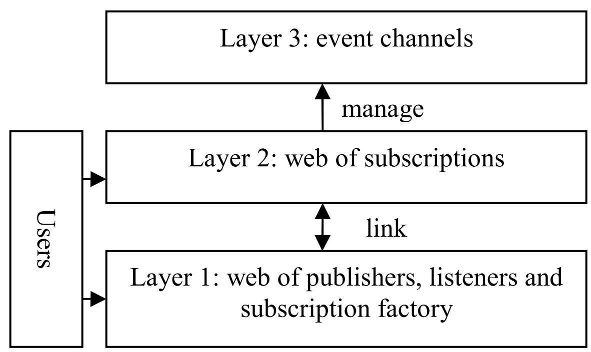


Figure 1. Three layers of the event-driven web

Layer 1 is a web of event publishers, listeners and subscription factories. A publisher is a resource that advertises events. A listener is a resource that accepts event notifications. A subscription factory is a resource that accepts subscriptions on behalf of a publisher. These resources expose their functions through REST services. They serve as the access points to a complex event notification system treated as a black box to the Web.

Layer 2 is a web of subscription resources that are created from the resources in Layer 1. Subscription resources exist as entangled pairs and each pair defines an association between an event publisher and an event listener, in which event notifications are sent. These associations are referred as event channels. These entangled subscriptions resources provide REST services to manage the event channels. Layer 1 and 2 resources are connected so navigations between layers are supported.

Layer 3 consists of event channels created from subscription resources in Layer 2. Unlike resources in layers 1 and 2 that provide REST services, the event channels may use any protocol to transmit event notifications, such as HTTP, JMS, and TCP/IP. In fact, the event notifications do not have to be discrete messages and can be media streams that are transmitted over RTP as proposed in WIP [41]. If two event systems do not use the same protocol for notifications, adaptors may be used to convert notifications.

To build such a layered system through integration of existing distributed event systems, REST-Event framework defines a set of minimal REST services required for the resources in Layer 1, upon which subscriptions can be created. The REST-Event itself defines the protocols for the subscription management. To be flexible, REST-Event does not define the protocols for Layer 3 since these protocols are defined by the existing event systems. REST-Event does not define any event filter languages either as they are also defined by the existing event systems. However, the notification protocols and filters can be specified in subscriptions.

IV. REST-EVENT FRAMEWORK

REST-Event framework defines a minimal set of resources and protocols to support the creation and management of event channels through subscriptions. The following subsections describe the core resources and protocols, namely: Discovery, Creation and Deletion in this framework.

A. Discovery Protocol

REST requires that a REST API should be entered with a single URI without any prior knowledge except media types and link relations [3]. This means, when being provided with a URI, an event subscriber must be able to determine if the URI points to an event publisher, and if so, which resource can be used to create subscriptions. To satisfy this requirement, REST-Event requires an event publisher to support the Discovery Protocol. The protocol contains three elements: a HTTP HEAD request from a subscriber to a publisher, a HTTP response from the publisher to the subscriber, and a special link relation `subscribe` in the response message. Suppose the given URI is http://www.host1.com/topic1, then the HEAD request and response could be:

```
HEAD /topic1 HTTP 1.1
Host: www.host1.com

200 OK HTTP 1.1
Link: </topic1/factory1>; rel=subscribe
```

The special `subscribe` link in the response tells the subscriber two things: 1) the requested resource is an event publisher; and 2) the location of its subscription factory that accepts subscription requests to this event publisher.

The link is specified in the header following RFC5988 [42]. This approach allows an event publisher to delegate its subscription management to another resource. Specifying the link in the header instead of the body has the advantage that the link is independent of the outgoing representations.

B. Creation Protocol

In conventional event-based systems, a subscription represents a one-way event channel, in which event notifications flow from the publisher to the listener. However, in many cases, two-way event channels, in which event notifications can flow in both directions are necessary in communication and collaboration systems. It is possible for a subscriber to create a two-way event channel in these systems with two separate subscriptions, each representing a one-way event channel. But this approach has the following drawbacks. First, the system complexity is exposed to the subscriber, which is typically a human user. Second, the system relies on an external entity (subscriber) to control its state. If the external entity leaves the system in an inconsistent state, for example failing to delete one subscription, then it is difficult for the system to detect and recover from the inconsistence.

To address this issue, REST-Event supports creation and management of two-way event channels in one transaction initiated by a subscriber. The conventional one-way event channels are a special case of two-way event channels. This approach hides the complexity and keeps the system in the loop so that any failure can be detected and recovered. Two-way event channels are created by a process called subscription entanglement, in which a pair of subscriptions are created and linked to have the same lifecycles. Subscription entanglement is realized by the subscription protocol that involves interactions between three entities: an event subscriber, a subscription factory and an event listener. REST-Event therefore requires that event systems to be integrated exposing a subscription factory resource and a listener resource that support the Creation Protocol.

Without losing generality, we assume the subscription protocol is defined in terms of HTTP and XML according to REST. Figure **2** illustrates the protocol messages exchanged between the involved resources: `topic1` is an event publisher resource and `facotory1` is the subscription factory of `topic1` of the first event system; `topic2` is an event listener resource and `factory2` is the subscription factory of `topic2` of the second event system.

The Creation Protocol creates a two-way event channel between the two systems through subscription entanglement as follows.

Step 1: the subscriber sends a HTTP POST request to `http://www.host1.com/topic1/factory1` to create a subscription that specifies a two-way event channel consisting of two one-way channels. Both outbound and inbound channels specify a source URI and a sink URI. In this case, the outbound channel tranmits notifications from the source `topic1` to the sink `topic2` and the inbound channel goes the opposite direction. In general, the two event channels can be separated with 2 different pairs of sources and sinks, but always have the same lifetime as specified by the expiry element. Each event channel can have its own filter:

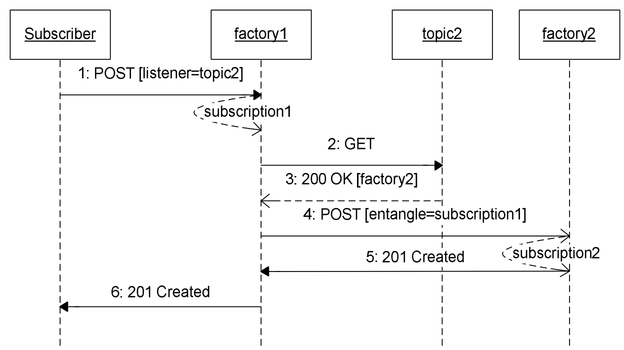


Figure 2. Creation Protocol

```
POST /factory1 HTTP 1.1
Host: www.host1.com
Accept: application/xml

<subscription>
    <expiry>…</expiry>
    <outbound>
       <source
href="http://www.host2.com/topic1" />
       <sink
href="http://www.host2.com/topic2" />
       <filter>…</filter>
    </outbound>
    <inbound>
       <source
href="http://www.host2.com/topic2" />
       <sink
href="http://www.host2.com/topic1" />
       <filter>…</filter>
    </inbound>
</subscription>
```

The `factory1` resource will process this request and create a subscription resource `subscription1` that represents the outbound event channel from `topic2` to `topic1` in the request.

Steps 2-3: The `factory1` resource sends a HTTP GET to the `topic2` resource to discover its factory using the Discovery Protocol discussed before.

Step 4: The `factory1` resource sends a POST request to `factory2` found above:

```
POST /factory2 HTTP 1.1
Host: www.host2.com
Accept: application/xml

<subscription>
    <expiry>…</expiry>
    <link                    rel="entangle"
href="http://www.host1.com/subscription1" />
    <outbound>
       <source
href="http://www.host2.com/topic2" />
       <sink
href="http://www.host1.com/topic1" />
       <filter>…</filter>
    </outbound>
</subscription>
```

Step 5: The `factory2` resource creates `subscription2` and links it with `subscription1`. On success, it responds with a URI to the created `subscription2`:

```
201 Created HTTP 1.1
Content-Type: application/xml
Location: http://www.host2.com/subscription2
```

Upon receiption of the response, the `factory1` links `subscription1` to `subscription2`.

Step 6: The `factory1` resources returns a response to the subscriber that contains the link to `subscription1` for the subscriber to access the entangled subscriptions:

```
201 Created HTTP 1.1
Location: http://www.host1.com/subscription1
```

This completes the creation of entangled subscriptions that are mutually linked. If there is a failure in steps 2-5, the partial subscriptions will be deleted and the subscriber will receive an error status code in response.

To create a one-way event channel from `topic1` to `topic2`, we just remove the `<inbound>` element from the request message in Step 1 and the `<outbound>` element from the request in Step 4. The rest messages will be the same.

The `factory2` discovered in steps 2-3 can be cached so the total number of messages can be reduced to 4 when the Subscription Protocol is repeated on the same resources.

C. *Deletion Protocol*

Since the entangled subscriptions have the same lifecycle, deleting any one of them will delete the other. The Deletion Protocol is illustrated by the following sequence diagram (Figure **3**).

When a client deletes a subscription, the resource will delete the local subscription state to reclaim the space. It then deletes the entangled subscription to maintain the same lifecycle. The deletions can also be initiated by a HTTP server that terminates a local subscription for various reasons, such as the server is shutting down or reclaiming spaces.

V. TOPIC WEB

This section demonstrates the use of REST-Event framework in creating a form of event-driven web called topic web. A topic web consists of federated topic hubs that implement the REST-Event protocols. A topic hub hosts many topic resources that are linked into a topic tree. A topic resource is a kind of event broker. A topic web can be regarded as the event delivery backbone in the conventional distributed event systems. But a topic web offer more flexibility and extensibility than conventional event delivery backbones.

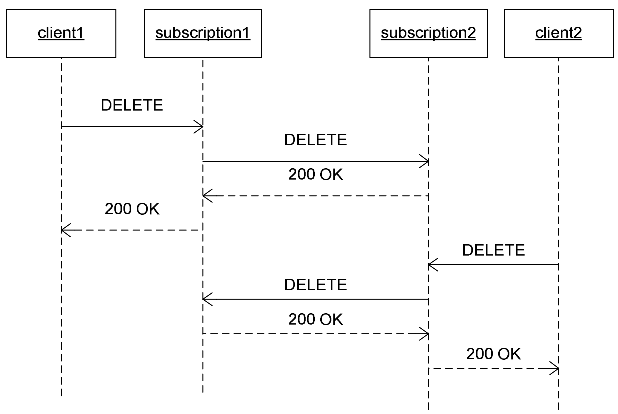


Figure 3. Deletion Protocol

A topic hub hosts resources required by REST-Event: topic, which is a publisher and listener, subscription factories and subscriptions. Each hub also hosts a presence resource, through which an administrator can start or shut down the services. A hub can be owned and operated by a single user or shared by a group of users. A topic hub can also invoke distributed event processors to process notifications. The high level interactions between a topic hub and its clients and servers are illustrated in Figure **4**.

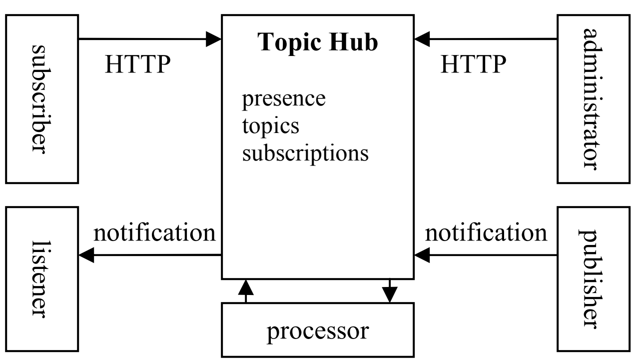


Figure 4. Topic hub resources and interactions

The topic hub is a light weight component and it can be run on any devices, including mobile phones that support HTTP protocol. It can be a Java Servlet on a HTTP server, a standalone HTTP server, or embedded in another application.

A topic hub can be a gateway between conventional event systems and the REST web services. In this sense, a topic hub represents a complex event system hidden to the Web. This approach can significantly reduce the cost of web service development while reusing the existing event infrastructures to ensure quality of services.

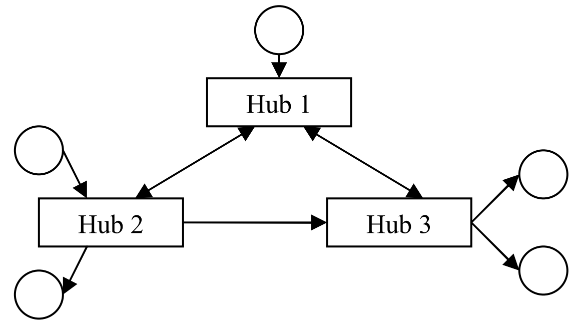


Figure 5. A topic web

Because a topic hub is based on REST design, it is stateless. Consequently, a topic hub can shut down and restart safely without losing any of its services, provided that the resource states are persisted. This is especially useful when the hubs are hosted on mobile devices, which can be turned on and off. Because a topic hub is stateless, it is also scalable. We can add more topic hubs to support more clients without worrying about client session replica or affinity.

Event channels between topic hubs are created and managed by REST-Event protocols. An example topic web is illustrated in Figure **5**, where topic hubs are represented as rectangles and publishers/listeners are represented by circles. The arrows indicate the event channels.

The following paragraphs describe the elements in topic web in a more formal setting with set-theoretic notations.

A topic tree is a set of topics organized as a tree. A topic is a resource, to which events can be published and subscribed. More formally, a topic t has a set of events E, a set of children topics C:

$$t = (E, C),\ C=\{\ t_j \mid t_j \text{ is a child topic of } t\}.$$

Given a set of topic hubs $H=\{h_i\}$ where each hub hosts a set of topic trees $T(h_i)=\{t|t \text{ is a topic on } h_i\}$, these topic trees form a web of topics linked by entangled subscriptions. More formally, a topic web $W(H)$ on top of a set of hubs H is defined as:

$$W(H) = \bigcup_{h_i \in H} T(h_i)$$

A. *Resource Design*

The key properties, interfaces and relations of the resources are depicted in the UML class diagram in Figure **6**.

Each resource on a hub is addressed by a URI. The following templates are used to reflect the subordinate relations defined above:

- Topic t: */topics/{t}*;
- Child topic t_j of topic t: */topics/{t}/topics/{t_j}*;
- Subscription factory of topic t: */topics/{t}/subscriptions;*
- Subscription s of topic t: */topics/{t}/subscriptions/{s}*;

Entangled subscriptions between topic *ta* on hub A to topic *tb* on hub B is established by a user using a web browser following the REST-Event Creation Protocol.

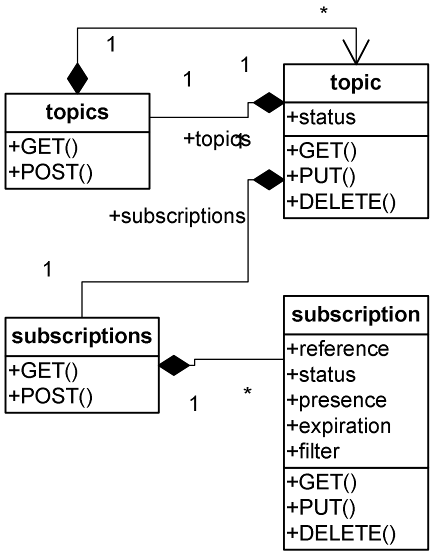


Figure 6. Main resources on topic hub

A notification is propagated between hubs as follows:

1. The user posts a notification to a topic on a hub from a web browser using HTTP POST.
2. The notification is delivered by a scheduler to all listening topics with PUT that maintains the original UUID assigned to the notification by the original hub; as the result, all the propagated notifications on different hubs can be identified by the same UUID.

The topic web does not define the representations of its resources, which is left to the implementations. Different representations (media types) of the same resource are supported through HTTP content negotiation. The communications between web browsers and the topic hubs are also outside the scope of this framework, as we expected they can be addressed by the upcoming W3C standards [10].

B. Security

The communication between the topic hubs are secured using HTTPS with PKI certificates based mutual authentication. For this to work, each topic hub maintains a X.509 certificate issued by a CA (Certificate Authority) that is trusted by other hubs. It is possible or even preferable to obtain two certificates for each topic hub: one for its client role and one for its server role, such that these two roles can be managed separately.

The communications between the topic hubs and web browsers (users) are also secured by HTTPS. In this case, the browser authenticates the topic hub certificate against its trusted CA. In return, users authenticate themselves to the hub using registered credentials (login/password or certificate). Once a user is authenticated to a topic hub A, it employs role-based authorization model to authorize a user for his actions.

If the user wants to create a subscription link from hub A to hub B, B has to authorize A for the inbound subscription. To satisfy this condition, the user first obtains an authenticated authorization token from hub B. The user then sends this token with the subscription message to hub A. Hub A uses this token to authorize itself to hub B for the inbound subscription creation. Once hub B creates the resource, it returns an access token to hub A to authorize it for future notifications to that topic.

An alternative to the above scheme is to use the OAuth 1.0 Protocol [32] that allows a user to authorize a third-party access to his resources on a server. In this case, hub A becomes the third-party that needs to access the topic resources on hub B owned by the user. Here is how it works at a very high level: 1) the user visits hub A to create a subscription to hub B; 2) hub A obtains a request token from hub B and redirects the user to hub B to authorize it; 3) the user provides his credentials to hub B to authorize the request token and hub B redirects the user back to hub A; 4) hub A uses the authorized request token to obtain an access token from hub B and creates the inbound subscription on B.

In both approaches, the user does not have to share his credentials on hub B with hub A.

VI. Features of Topic Web

On surface, the topic web built by REST-Event framework, as described in the previous section, appears similar to the broker overlay network in the conventional notification architecture [14]. However, it has the following advantages due to a REST based design.

A. Addressability and Connectedness

Unlike conventional broker overlay networks that are closed systems whose accessibility are prescribed by the APIs, a topic web is open, addressable and connected. Unlike in a conventional broker overlay network that distinguishes between inner, border, or special rendezvous brokers, a topic web consists of homogeneous topic hubs with the same type of web services. Users can navigate and search the topic web to find the interested information using regular web browsers or crawlers. The addressability and connectedness increase the "surface areas" of the web services such that the information and services in a topic web can be integrated in many useful ways beyond what is anticipated by the original design.

B. Dynamic and Flexible Topology

Unlike in conventional broker network where brokers have fixed routing tables, a topic web can be dynamically assembled and disassembled by users for different needs. Its topology can be changed on the fly as subscriptions are created and deleted and hubs join and leave the topic web. For example, a workflow system can be created where work items are propagated as notifications between users. In an emergence situation, a group of people can create an ad-hoc notification network to share alerts and keep informed. In an enterprise, a topic web about a product can be created on-demand such that alerts from field technicians can propagate to proper sales and supporting engineers who are in charge of the product to better serve the customers. In any case, once the task is finished, the topic web can be disassembled or removed completely. In this sense, a topic web is similar to an ad-hoc peer-to-peer network. However, a topic web is based on REST web services, whereas each type of P2P network depends on its own protocols.

In conventional notification services, a broker routes all messages using one routing table. Therefore, it cannot participate in more than one routing topology. In our framework, a hub can host many topics, each having its own routing table (subscriptions). As a result, a hub can simultaneously participate in many different routing networks. This gives the users the ability to simultaneously engage in different collaboration tasks using the same topic web.

C. Robustness and Scalability

Topic hubs can be made robust because its resource states can be persisted and restored to support temporary server shutdown or failover.

The safe and idempotent operations, as defined by HTTP 1.1 [30] also contribute to the robustness. Our framework uses nested HTTP operations where one operation calls other operations. We ensure that such a chain of operations is safe and idempotent by limiting how operations can be nested inside each other as follows:

nested(GET)={GET}
nested(POST)={GET,POST,PUT,DELETE}
nested(PUT)={GET,PUT,DELETE}
nested(DELETE)={GET,PUT,DELETE}

The robustness and scalability also come from the statelessness of REST design. The statelessness means that a topic hub can process any request in isolation without any previous context. By removing the need for such context, we eliminate a lot of failure conditions. In case we need to handle more client requests, we can simply add more servers and have the load balancer distribute the requests to the servers who share the resources. If the resource access becomes a bottleneck, we can consider duplication or partition of resources. Robustness and scalability can be crucial when a topic hub serves as the gateway to large-scale notification systems.

VII. Implementation and Experiments

A prototype topic web has been developed based on the described REST-Event framework. The notification system allows users within a group to publish and subscribe presence information and text messages. Users can respond to received messages to enable real-time collaboration. For example, when an expert becomes available through his presence notification, a manager may respond to the notification and propose a new task force be formed with the expert as the team leader. This response is propagated to the group over the event channels so that interested members can set up a new workflow using the proposed topic web.

Users interact with the topic web with Web browsers without any download. The following is a screenshot of a web page of a particular topic (Figure **7**).

From a topic page, a user can follow the link to the subscription factory page to create subscriptions (Figure **9** and Figure **9**).

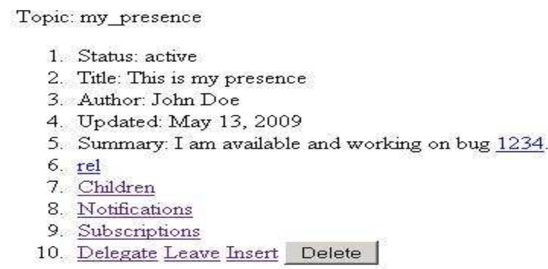


Figure 7. A topic web page

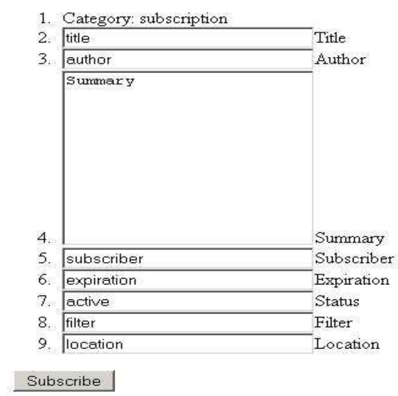


Figure 8. Page for creating subscription

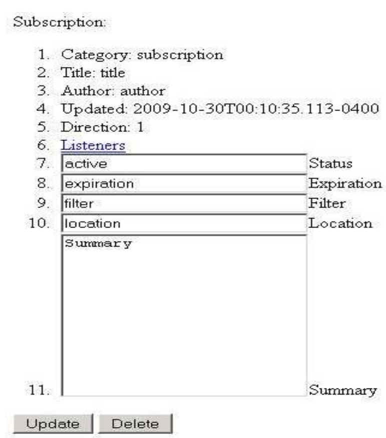


Figure 9. Page for a created subscription

In this prototype system, a user can post a message to a topic using a web browser (Figure **10**). The topic hub will propagate the message over the event channels. All users who subscribe to the topic directly or indirectly through other topics will receive the message in a notification. In this topic web, notifications for text messages are also modeled as resources that can be linked to track the interaction history. When a user posts a message to a topic, it is saved by the topic hub and all notifications for this message are linked to the original copy. If another user responds to this message, the response is again saved in a topic hub and linked to the original message. A user can follow this response chain through the hyperlinks embedded in the notifications. In some sense, the messages are like tweets. However, the topic web is not a single web site as www.twitter.com. Instead, the topic web is a distributed system consists of many such web sites.

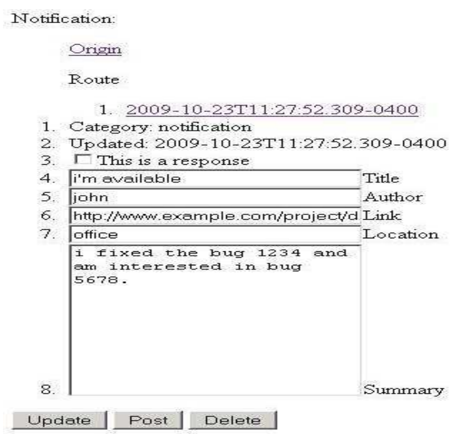


Figure 10. Page for posting a message

The prototype was written in Java using Restlet 1.1.4 [24]. The implementation followed the Model-View-Controller (MVC) design pattern. The Model contains the persistent data stored on disk. The Controller contains the resources and the View contains the view objects that generate XHTML pages from the XHTML templates. The topic hub stack was implemented by four Java packages, as illustrated in Figure **11**.

For this prototype, we used OpenSSL package [31] as the CA to generate certificates for the topic hubs, and Java keytool to manage the keystores for the hubs. Resources states are managed by a file manager that synchronizes the access to them. A hub used a separate thread to dispatch notifications from a queue shared by all resources. Because HTML form only supports POST and GET, we used JavaScript (XMLHttpRequest) to implement the PUT and DELETE operations for pages that update or delete resources.

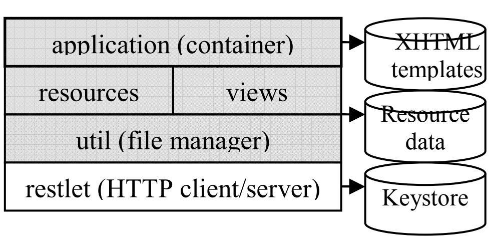


Figure 11. Topic hub stack

Users interact with the services using web browsers (Firefox in our case). For demo purpose, the notifications were delivered to the browsers using automatic page refreshing. This is a temporary solution as our focus is on communications between hubs, instead of between browser and server. However, the REST-Event framework should work with any client side technologies, such as Ajax or Server-Sent Event technologies.

We measured the performance of the prototype system in a LAN environment. The hubs were running on a Windows 2003 Server with 3GHz dual core and 2GB RAM. The performances of several key services were measured, where S means subscription, L means listener, and N means notification. The time durations for each method are recorded in the following table. The time duration includes processing the request, saving data to the disk, and assembling the resource representation.

TABLE 1. PERFORMANCE MEASURED IN MILLISECONDS

task/time	POST S	POST L	PUT S	POST N	PUT N
avg	14.1	38.9	6.2	9.5	0
std	13.7	16.8	8.0	8.1	0

The table shows that adding a listener (POST L) takes the longest time and this is expected because it is a nested operation, where

t(POST L)=processing time + network latency + t(PUT S).

The time to update a notification (PUT N) is ignorable (0 ms) and this is good news, since we use PUT to propagate notifications.

VIII. CONCLUSIONS

In this paper, we described an approach - REST-Event framework for event-driven web, in which elements of EDA (event-driven-architecture) can be projected and represented by REST resources, protocols and services. The basic REST resources, protocols, services and securities in this framework were specified and constructed. Moreover, a special event-driven web, topic web, was proposed and built based on REST-Event. We studied features in REST-Event approach, including addressability, dynamic topology, robustness, and scalability, etc., and compared them with the conventional notification systems.

In addition, we developed a prototype REST-Event based system using secure HTTP. Preliminary performance tests showed that the proposed approach is feasible and advantageous.

Our plan is to test the framework in a larger scale network environment and analyze its behaviors and performance in those deployments.

REFERENCES

[1] Li Li and Wu Chou: R-Event: A RESTful Web Service Framework for Building Event-Driven Web, Service Computation 2010, pages 7-13, Lisbon, Portugal, November 21-26, 2010.

[2] Richardson, L. and Ruby, S., *RESTful Web Services*, O'Reilly Media, Inc. 2007.

[3] Fielding, R., *Architectural Styles and the Design of Network-based Software Architectures*, Ph.D. Dissertation, 2000, http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm. Last Accessed: January 5, 2012.

[4] Jacobs, I. and Walsh, N., (eds), *Architecture of the World Wide Web, Volume One*, W3C Recommendation 15 December 2004. http://www.w3.org/TR/webarch/, Last Accessed: January 5, 2012.

[5] The Atom Syndication Format, 2005, http://www.ietf.org/rfc/rfc4287.txt, Last Accessed: January 5, 2012.

[6] The Atom Publishing Protocol, 2007, http://www.ietf.org/rfc/rfc5023.txt, January 5, 2012.

[7] RSS 2.0 Specification, 2006, http://www.rssboard.org/rss-specification, Last Accessed: January 5, 2012.

[8] Pushlets, http://www.pushlets.com/, Last Accessed: January 5, 2012.

[9] HTML Working Group, 2009, http://www.w3.org/html/wg/, Last Accessed: January 5, 2012.

[10] Hickson, I. (ed), Server-Sent Events, W3C Working Draft 29 October 2009, http://www.w3.org/TR/eventsource/, Last Accessed: January 5, 2012.

[11] Hickson, I. (ed), The Web Sockets API, W3C Working Draft 29 October 2009, http://www.w3.org/TR/websockets/, Last Accessed: January 5, 2012.

[12] Gamma, E., Helm, R., Johnson, R., and Vlissides, J., *Design Patterns*, Addison-Wesley, 1995

[13] Taylor, H., Yochem, A., Phillips, L., and Martinez, F., *Event-Driven Architecture, How SOA Enables the Real-Time Enterprise*, Addison-Wesley, 2009.

[14] Mühl, G., Fiege, L., and Pietzuch, P.R., *Distributed Event-Based Systems*, Springer, 2006.

[15] Rowstron, A., Kermarrec, A.M., Castro, M., and Druschel, P., SCRIBE: The design of a large-scale event notification infrastructure, Proc. of 3rd International Workshop on Networked Group Communication, November 2001, pp 30-43.

[16] Buschmann, F., Meunier, R., Rohnert, H., Sommerlad, P., and Stal, M. (1996). *Pattern-Oriented Software Architecture: A System of Patterns*. West Sussex, England: John Wiley & Sons Ltd., 1996.

[17] Chandy, K. M. (2006). Event-Driven Applications: Costs, Benefits and Design Approaches, Gartner Application Integration and Web Services Summit 2006, http://www.infospheres.caltech.edu/node/38, Last Accessed: January 5, 2012.

[18] Michelson, B. M. (2006). Event-Driven Architecture Overview, http://soa.omg.org/Uploaded%20Docs/EDA/bda2-2-06cc.pdf, Last Accessed: January 5, 2012.

[19] Davis, D., Malhotra, A., Warr, K., and Chou, W. (eds), Web Services Eventing (WS-Eventing), W3C Working Draft, 5 August 2010. http://www.w3.org/TR/ws-eventing/, Last Accessed: January 5, 2012.

[20] Graham, S., Hull, D., and Murray, B. (eds), Web Services Base Notification 1.3 (WS-BaseNotification), OASIS Standard, 1 October 2006. http://docs.oasis-open.org/wsn/wsn-ws_base_notification-1.3-spec-os.pdf, Last Accessed: January 5, 2012.

[21] Chappell, D. and Liu, L. (eds), Web Services Brokered Notification 1.3 (WS-BrokeredNotification), OASIS Standard, 1 October 2006. http://docs.oasis-open.org/wsn/wsn-ws_brokered_notification-1.3-spec-os.pdf, Last Accessed: January 5, 2012.

[22] Vambenepe, W., Graham, S., and Biblett, P. (eds), Web Services Topics 1.3 (WS-Topics), OASIS Standard, 1 October 2006. http://docs.oasis-open.org/wsn/wsn-ws_topics-1.3-spec-os.pdf, Last Accessed: January 5, 2012.

[23] Fitzpatrick, B., Slatkin, B., and Atkins, M., PubSubHubbub Core 0.2, Working Draft, 1 September 2009, http://code.google.com/p/pubsubhubbub/, Last Accessed: January 5, 2012.

[24] Restlet, RESTful Web framework for Java, http://www.restlet.org/, Last Accessed: January 5, 2012.

[25] JMS (2002). Java Message Service, version 1.1, 2002, http://www.oracle.com/technetwork/java/index-jsp-142945.html, Last Accessed: January 5, 2012.

[26] Event Service Specification, Version 1.2, October 2004, 2004.

[27] Notification Service Specification, Version 1.1, October 2004.

[28] Christensen, E., Curbera, F., Meredith, G., and Weerawarana, S., Web Services Description Language (WSDL 1.1), W3C Note, 15 March 2001. http://www.w3.org/TR/wsdl, Last Accessed: January 5, 2012.

[29] Gudgin, M., et al, SOAP Version 1.2 Part 1: Messaging Framework (Second Edition), W3C Recommendation, 27 April 2007. http://www.w3.org/TR/soap12-part1/, Last Accessed: January 5, 2012.

[30] Fielding, R., et al. Hypertext Transfer Protocol – HTTP/1.1. http://www.w3.org/Protocols/rfc2616/rfc2616.html, Last Accessed: January 5, 2012.

[31] OpenSSL: http://www.openssl.org/, Last Accessed: January 5, 2012.

[32] The OAuth 1.0 Protocol: http://tools.ietf.org/html/rfc5849, Last Accessed: January 5, 2012.

[33] RFC6202: Known Issues and Best Practices for the Use of Long Polling and Streaming in Bidirectional HTTP, http://tools.ietf.org/html/rfc6202, Last Accessed: January 5, 2012.

[34] The Bayeux Specification: http://svn.cometd.com/trunk/bayeux/bayeux.html, Last Accessed: January 5, 2012.

[35] XEP-0124: Bidirectional-streams Over Synchronous HTTP (BOSH): http://xmpp.org/extensions/xep-0124.html, Last Accessed: January 5, 2012.

[36] XEP-0060: Publish-Subscribe: http://xmpp.org/extensions/xep-0060.html, Last Accessed: January 5, 2012.

[37] Google Wave Federation Protocol: http://wave-protocol.googlecode.com/hg/spec/federation/wavespec.html, Last Accessed: January 5, 2012.

[38] Google Wave Client-Server Protocol: http://wave-protocol.googlecode.com/hg/whitepapers/client-server-protocol/client-server-protocol.html, Last Accessed: January 5, 2012.

[39] Event driven architecture onto the Azure Services Platform: http://www.microsoft.com/belux/architect/issue_3/azure_services_platform.aspx, Last Accessed: January 5, 2012.

[40] Event-Driven Architecture: SOA Through the Looking Class: http://msdn.microsoft.com/en-us/architecture/aa699424, Last Accessed: January 5, 2012.

[41] Wu Chou, Li Li, Feng Liu, Web Services for Communication over IP, IEEE Communication Magazine, vol. 46 no. 3, page 136-143, March 2008.

[42] RFC9588: Web Linking, http://tools.ietf.org/html/rfc5988, Last Accessed: January 5, 2012.

Generic Function Schema as a Means for Similar-Fashioned Operations on Heterogeneous Connection Properties

Mark Yampolskiy[1,4], Wolfgang Hommel[2,4], David Schmitz[2,4], Michael Schiffers[3,4]
myy@isis.vanderbilt.edu, hommel@lrz.de, schmitz@lrz.de, schiffer@nm.ifi.lmu.de

[1]*Vanderbilt University (VU),* [2]*Leibniz Supercomputing Centre (LRZ),* [3]*Ludwig Maximilians University Munich (LMU),*
[4]*Munich Network Management (MNM) Team*

***Abstract*—Graphs are often used to model interconnected topological objects with different connection properties. Path finding in a weighted graph belongs to the classical problems of graph theory. Whereas the addition of the edges' weights as an aggregation and the interpretation of a smaller resulting sum as the preferable path works very well in applications like path computations, e.g., for road maps, it is not always applicable to those connections in computer networks that need to fulfill multiple independent Quality of Service (QoS) criteria simultaneously. Until now, special solutions are implemented – often manually – for each new service and for each QoS parameter separately. As the development of novel customer-tailored network services often relies on different connection properties and their combinations, a generic treatment of QoS parameters becomes a critical factor for rapid development and network service rollout. In this article, we present a generic function schema for treating multiple independent QoS parameters in a similarly fashioned way. Our work fosters efficient routing algorithms that are considering multiple connection properties and corresponding constraints at the same time, as they are required, for example, in Future Internet infrastructures with end-to-end QoS guarantees and in dynamic survivability-aware environments.**

Keywords-graph theory; multi-weighted graphs; QoS; QoS aggregation; QoS comparison.

I. Introduction

Obviously, network connections are meanwhile broadly used as a basis for or as an integral part of the services that are realized upon them. Examples can be found in areas like Internet-telephony, video-conferencing and video-on-demand, connectivity for Grid cooperation, IT service outsourcing, etc. Common to all these examples is that the overall service quality directly depends on the combination of multiple *Quality of Service* (QoS) parameters of the underlying network connections. For instance, services like telephony, e.g., realized using standard VoIP (Voice over IP) protocols, are very sensitive to transmission jitter, as the human ear is very sensitive to the delay variation; on the other hand, video streaming depends primarily on the data rate, so that the end-users do not have to wait periodically for the transmission of the next portion of a high-resolution video; multi-player gaming in LANs and e-Sports over the Internet often demand low latency and are sensitive to packet or even connection loss; network connections for business and scientific applications often combine requirements for multiple QoS parameters at the same time and specify tighter thresholds than the usual consumer applications' demands.

The fulfillment of the service- or customer-specific *end-to-end* (E2E) requirements is only possible if all respective QoS parameters are considered during the path computation (routing), either as basis or as derived parameters . The state-of-the-art routing algorithms that are taking into account E2E requirements typically operate on weighted graphs in almost the same manner as approaches operating on multi-weighted graphs, i.e., on graphs with multiple weights associated with the single edge representing values of multiple independent QoS parameters.

In graph theory, it is common to use the *addition* of edge-weights as an aggregation function with the path with the smaller sum as the most preferred alternative. Such procedures are very well suited in applications like path computations for road maps in mobile navigation assistants. However, the approach is not always applicable to connections in computer networks with combined QoS parameters as the two network QoS parameters considered most often (bandwidth and delay) show. Whereas the typical parameter treatment is applicable to "delay", a different function is needed for "bandwidth": The aggregation function needs to choose the minimum bandwidth of all involved connection segments and larger values are preferred over smaller ones. In general, adequate QoS aggregation functions are significantly more complex than sum-of or minimum-of if other QoS parameters like reliability and availability have to be regarded as well.

Until now, there is no generally applicable solution to overcome these difficulties. On the other hand, however, the time for the development of new services with customer-specific QoS parameters is becoming a crucial success factor.

In order to cope with the high variety of customer- and service-specific requirements, we have presented in [1] a generic function schema which allows the treatment of various arbitrary network QoS parameters in a similar-

This work was done while Mark Yampolskiy was employed by the LRZ and worked for the DFN in the Géant project.

fashioned way. The proposal in [1] includes an efficient way to distinguish between different QoS parameters; a standardized general treatment for the aggregation of and the comparison between values of a particular QoS parameter; and the support of customer-relevant combinations of arbitrary QoS parameters. The functions proposed in [1] can be used by routing procedures in order to find a path fulfilling the E2E constraints or as part of the monitoring of established connections in order to ensure the fulfillment of committed E2E connection qualities. However, [1] is limited to operations on properties that could be called *basis QoS parameters*. These are QoS parameters like bandwidth or delay, which can be measured directly. [1] is not applicable to *derived QoS parameters* that result from combining multiple basis QoS parameters with arbitrary formulae. A good example of such a parameter is "availability", defined as the ratio of the total time a component is capable of being used during a given time interval to the length of this interval.

Consequently, while basis parameters induce metrics the dependencies of which on other variables is not explicit, derived parameters induce metrics where the influencing variables are explicitly considered. The important practical benefit of derived QoS parameters is their conceptual separation from basis parameters (and thus from dedicated measurements). For example, reliability (a derived QoS parameter) – defined as the probability that a service will perform its intended function during a specified period of time –, is not solely tied to uptime (a basis parameter). It could also be coupled to reaction times, response times or – in the context of a reliability-scalability metric – even to the number of users when defining it as the Mean Time to Failure (MTTF) in dependency of the number of users.

Despite these obvious advantages, however, the determination of derived QoS parameters induces a significant increase in complexity as the operations to aggregate the parameters are not necessarily homogeneous (summations may be combined with min-max-considerations and set theoretic intersections).

The remainder of the paper is organized as follows: We first analyze the state of the art in Section II. In Section III, we summarize first the main results of our original paper [1] which are necessary to understand the subsequent sections. We then generalize from the basis QoS parameters to derived ones and show how operations on them can be formally specified. In Section IV, we extend the discussion to properly treating the special problem of value ranges. In Section V we demonstrate the applicability of our approach to path finding problems in networks. Section VI discusses the operations required for aggregating derived QoS parameters, before we present an information model for derived QoS parameters in Section VII. Section VIII summarizes the resulting big picture and presents a concrete example. Finally, we conclude the paper in Section IX, where we also give an outlook to our ongoing work.

II. State of the Art

Most routing algorithms are based on graphs that have a single fixed value associated with their edges as weights. This representation is then used for finding a path (often the shortest path) between arbitrary nodes of the graph. However, such graphs do not reflect all specifics of computer networks (see Figure 1). For instance, different quality classes of the network infrastructure canonically lead to significant variances in parameter values. In order to process such value ranges, graphs can be transformed into so called *multigraphs* where nodes may be directly connected by one or more edges. Even in the simple case of weight ranges for a single property, such transformations can significantly increase the graph processing complexity. If multiple connection properties with value ranges have to be considered at the same time, the complexity increases even more drastically. Therefore, in [2], we proposed an information model which is able to describe value ranges. Consequently, such a description raises the necessity of adequate operations on ranges.

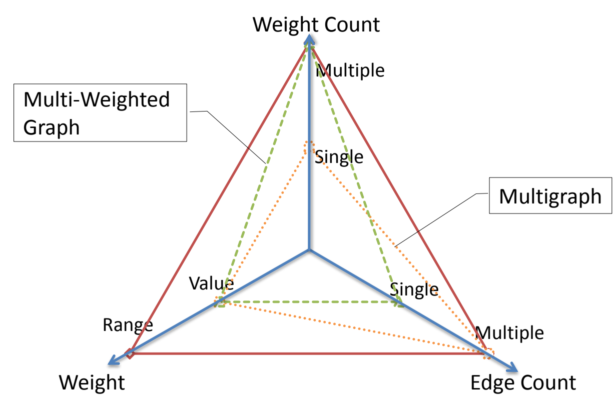


Figure 1. Classification of graph properties [1]

Graphs that support multiple weights at the same time are known as *multi-weighted graphs*. Such graphs are hardly investigated yet. In [7], a very good overview of the state of the art is given. It shows that path finding in multi-weighted graphs is in general an NP-complete problem. As path finding in multi-weighted graphs violates Bellman's optimality principle [8], routing algorithms that require this principle, e.g., Dijkstra's algorithm, cannot be used. Additionally, the handling of multiple properties at the same time is not solved adequately. Currently, the common understanding is to describe multiple properties as value vectors. This allows the use of vector-addition as property aggregation operation. For a comparison of weight vectors, the concept of *non-dominance* has been established [4]: A vector A is non-dominant to vector B only if all of its weight elements, i.e., property values, are smaller or equal to the corresponding elements of vector B.

As for single-weighted graphs, for multi-weighted graphs addition is also the pre-dominant aggregation function with a smaller value being the better one. Even if limitations of these operations w. r. t. the application to computer networks are long known, only workarounds have been proposed so far. For instance, in [9] the addition of $log(weight)$ is proposed if the true aggregation function for weights is multiplicative.

Alongside with the directly measurable QoS parameters, the necessity of parameters like service availability or reliability is very well agreed. Such quality parameters are defined in *IT Service Management* (ITSM) frameworks like ITIL [11] or e-TOM [12] and are widely used in *Service Level Agreements* (SLA) among network providers and carriers. Furthermore, more comprehensive research areas like survivability and dependability rely on derived QoS parameters (as, for example, pointed out in [10]) with arbitrary aggregation functions, corresponding weight comparison procedures, and improved handling of value ranges. While solutions to the first two aspects will be described in Section III, an efficient approach for handling value ranges during path finding will be presented in Section IV-A.

Besides these purely technical aspects, organizational specifics have to be considered as well. The so called policy-based routing between domains, does not only take technical aspects into account. Rather, it focuses on provider-specific interests. Along with very restrictive information and management policies, which are out of the scope of this paper, network operators and service providers are generally interested in the reduction of the resources that are required for a high-quality service delivery.

III. Operations on Connection Properties and their Generalization

In this section, we present a solution to function generalization regarding both single properties and property sets.

A. Functions for operations on a single property

During path finding the properties of the edges have to be aggregated. Typically, simple arithmetical addition is used as an aggregation function. As discussed in Section I, this is not necessarily the case for every QoS parameter. Furthermore, as discussed in [2], in the case of inter-domain connections each *Service Provider* (SP) may have access only to information regarding his own technical infrastructure which may not be sufficient to determine all relevant connection properties. In this case also the aggregation of the partial views of involved SPs at the same inter-domain connection is needed. The calculation of QoS properties of the inter-domain link from two partial views is not necessarily identical to the aggregation of two physical connections of the same type and length. For instance, when describing a connection with the property *delay*, not only the delay caused by the network cable should be considered, but also the delay caused by the active and passive network components used by each single SP; obviously, this varies between SPs.

If customer-specific end-to-end quality-of-service constraints need to be met, the value of the already found (partial) route has to be compared to these constraints during the path finding process. For path optimization it is also necessary to compare the values of several alternatives in order to choose the better one. In opposite to the case classically treated in graph theory, the meaning of what is *better* may vary between different QoS parameters. Regarding the examples mentioned above, for bandwidth a bigger value can be considered as a better one, however for delay a smaller value is the more preferred one.

Consequently, with each supported connection property operations for value aggregation and comparison have to be associated. As we will see, the necessary mathematical operations are not always as simple as adding values or selecting the minimum.

B. Associating operations with properties

In IT industry, new technologies and services are evolving very fast. Therefore prior to the association of operations with properties, a distinction between existing and projected properties is needed. We propose to assign a globally unique ID to each supported property. In order to ensure the global uniqueness of IDs, we propose to use a registration tree. Additionally to the distinction between properties, using a registration tree has another very important advantage. As multiple functions have to be associated with each supported property, it can be realized by the definition of the functions together with the registration of their property-ID (see Figure 2). Additionally, this will ensure the identity of functions used among SPs.

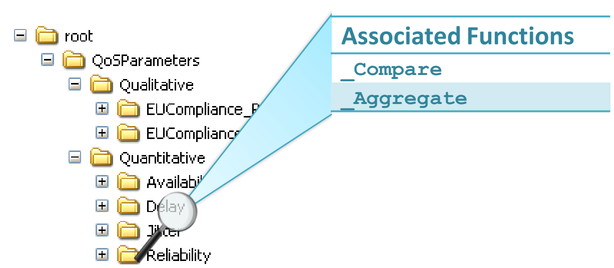


Figure 2. Registration tree example [1]

C. Comparison and aggregation of multiple properties

Based on the previous definition, we introduce an approach for the handling of m different properties with the globally unique IDs $\mathrm{ID}_1, \ldots, \mathrm{ID}_m$. In graph theory, it is common practice to use vectors in order to describe multiple weights associated with a single edge or a path in general. For any path in a graph with m properties, the weight can be specified as $\overrightarrow{U} ::= (u_1, \ldots, u_m) \in \mathbb{R}^m$. In this definition, u_j is the weight of the $j^{\underline{th}}$ property with ID_j. The order of properties in the weight vector can be arbitrary, as long as the placement of the properties is identical among all weight vectors. Further, for the edges of a path being enumerated from 1 to n, the weight of an edge with index i will be referred to as follows: $\overrightarrow{W}^i ::= (w_1^i, \ldots, w_m^i) \in \mathbb{R}^m$.

In order to calculate the weight vector $\overrightarrow{P}$ of the path consisting of n edges with weights $\overrightarrow{W}^1, \ldots, \overrightarrow{W}^n$, we first introduce an aggregation function for two weight vectors as follows:

$$\overrightarrow{\mathrm{Aggr}}(\overrightarrow{U}, \overrightarrow{V}) ::= (\mathrm{Aggr}_1(u_1, v_1), \ldots, \mathrm{Aggr}_m(u_m, v_m))$$

This definition is based on m aggregation functions for each property. The aggregation functions Aggr_i $(i = 1, \ldots, m)$ are associated with the property ID in the registration tree. We assume that all properties are independent of each other, i.e., they can vary without influencing the values of other properties. Furthermore, we assume that the binary operations defined by aggregation functions fulfill associative and commutative laws. Then we inductively define the computation of the whole path weight from weights of involved segments as follows:

$$\overrightarrow{\mathrm{Aggr}}(\overrightarrow{W}^1, ..., \overrightarrow{W}^n) ::= \overrightarrow{\mathrm{Aggr}}(\overrightarrow{\mathrm{Aggr}}(\overrightarrow{W}^1, \overrightarrow{W}^2), ..., \overrightarrow{W}^n)$$

Similar to the aggregation, we define the comparison of property vectors based on the comparison between identical properties. Corresponding to the *non-dominance* concept described in [4] we define that vector $\overrightarrow{U}$ is better than $\overrightarrow{V}$ if and only if all properties in the first vector are better than the corresponding properties of the second vector. In order to denote that property u_i of vector $\overrightarrow{U}$ is better than the corresponding property v_i of vector $\overrightarrow{V}$, we use the symbol "$\prec$". In contrast to the comparison of single values, it is possible that some properties of the first vector are better and some others are worse than those of the second vector. This situation should be treated as "indefinite". We depict this with the symbol "$\neq$". The comparison of two property sets can thus be defined as follows:

$$\overrightarrow{Compare}(\overrightarrow{U}, \overrightarrow{V}) ::= \begin{cases} =, \text{ if } & \forall 1 \le i \le m : u_i = v_i \\ \prec, \text{ if } & \forall 1 \le i \le m : (u_i \prec v_i \\ & \quad \vee\ u_i = v_i) \wedge \\ & \exists 1 \le j \le m : u_j \prec v_j \\ \succ, \text{ if } & \forall 1 \le i \le m : (u_i \succ v_i \\ & \quad \vee\ u_i = v_i) \wedge \\ & \exists 1 \le j \le m : u_j \succ v_j \\ \neq, \text{ if } & \exists 1 \le i \le m : u_i \prec v_i \wedge \\ & \exists 1 \le j \le m : u_j \succ v_j \end{cases}$$

IV. Treatment of Value Ranges

Some typical aspects of computer networks are not directly addressed by classical graph theory. In this section we propose the treatment of value ranges which can be associated with connection segments (graph edges) instead of multigraphs.

A. Path finding with value ranges

Physical network connections usually cannot be realized with a single property set because properties like bandwidth might vary in a wide range. A good example is the variation of achievable delays for a single logical connection, as it can be realized by different physical connections. Consequently, the property of the whole end-to-end (E2E) path between two endpoints may vary as well. We will refer to the value range of a particular path $path$ as

$$\overrightarrow{\overleftarrow{W}}^{path} = (\overrightarrow{W}^{path}_{min}, \overrightarrow{W}^{path}_{max}) \in \mathbb{R}^m \times \mathbb{R}^m,$$

i.e., the supported value range for the given path can vary from $\overrightarrow{W}^{path}_{\mathrm{min}}$ to $\overrightarrow{W}^{path}_{\mathrm{max}}$.

It is obvious that the path found between two endpoints can only be feasible if the best possible value fulfills the E2E constraints specified by customer (see Figure 3). Therefore, we propose to operate with the best values of the available connection segments during the path finding process.

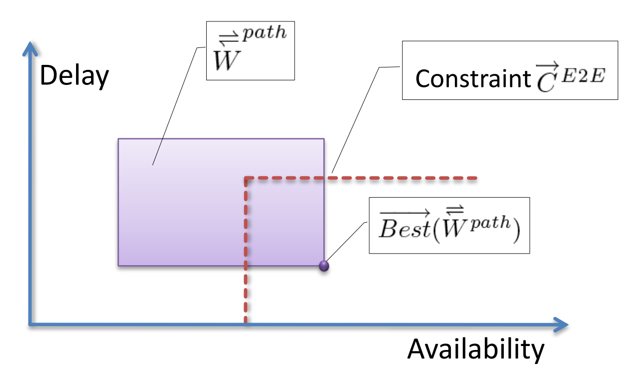


Figure 3. Fulfillment of end-to-end constraints [1]

We assume that all simultaneously considered path properties can vary independent of each other. Under this assumption, we define the selection function $Best$ for the best possible value of a path as follows:

$$\overrightarrow{Best}(\overrightarrow{\overrightarrow{W}}^{path}) = \overrightarrow{Best}(\overrightarrow{W}^{path}_{min}, \overrightarrow{W}^{path}_{max})$$

$$\overrightarrow{Best}(\overrightarrow{U}, \overrightarrow{V}) ::= (Best_1(u_1, v_1), \ldots, Best_m(u_m, v_m))$$

$$Best_i(u_i, v_i) ::= \begin{cases} u_i, & \text{if } u_i \prec v_i \\ v_i, & \text{otherwise} \end{cases} \quad \text{for } 1 \leq i \leq m$$

Please note that this definition is applicable not only to a path as a whole but also to any path segment.

B. Considering service provider interests: Optimization of resource usage

In contrast to customers, the service providers are usually interested in a reduction of resources used for service realization. This means that the requested service quality should not be the best possible one, but rather the one closest to the customer constraints. For paths complying with the E2E constraints, i.e., $\overrightarrow{Best}(\overrightarrow{\overrightarrow{W}}^{path}) \prec \overrightarrow{C}^{E2E}$, we distinguish between three cases as depicted in Figure 4, given the weights of alternative paths A, B and C:

- All worst properties of the considered path are worse than the constraints (see ”Path A”)
- All worst properties of the path are better than the constraints (see ”Path B”)
- The worst properties of the path are for some properties worse and for other properties better than the constraints (see ”Path C”)

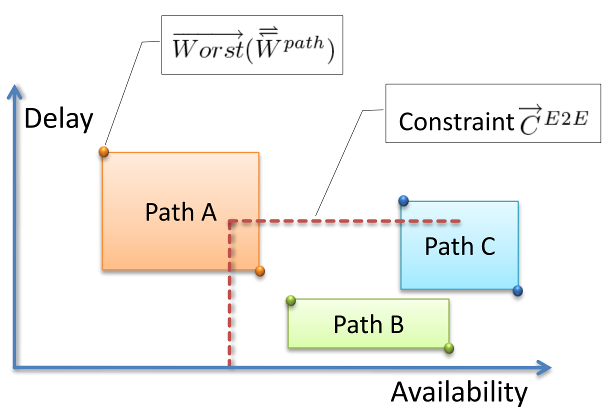


Figure 4. Pathweights of paths complying to constraints [1]

In order to distinguish between these alternatives, the function $Worst$ for the selection of the worst possible value of the found path can be defined as the opposite to $Best$.

In the case equivalent to ”Path B”, the worst possible value can be requested during the link ordering process. In the two remaining cases, an approximation to the constraint value should be performed. As the properties are independent of each other, such an approximation can be done separately (or even in parallel) for each affected property.

The whole E2E path weight is the aggregation of the weights of the involved parts. A possible gradation between the maximum and minimum values of connection parts is depicted in Figure 5. The E2E approximation of the path weight for a single property can be done in different ways. It can be seen as a knapsack-like problem with an intention to find a fit most close to the E2E constraint. We argue against this approach, as it may prevent the on-demand adaptation of requested service parts parameters. Instead we favor a ”fair split” among all connection parts. For each property i, we propose to use a divide-and-conquer strategy as follows:

1) For each connection part j with a value range between $w^j_{i,min}$ and $w^j_{i,max}$ we compute values $w^j_{i,best}$=Best$_i$ ($w^j_{i,min}$, $w^j_{i,max}$) and $w^j_{i,worst}$=Worst$_i$($w^j_{i,min}$, $w^j_{i,max}$).
2) For each connection part j we compute the realizable value $\left\lfloor \frac{w^j_{i,best}+w^j_{i,worst}}{2} \right\rfloor$.
3) If the computed path value $\sum_{j=1}^{k} \left\lfloor \frac{w^j_{i,best}+w^j_{i,worst}}{2} \right\rfloor$ is equivalent to the E2E constraint for the selected property, the selected values can be used as a result of this optimization.
4) If the computed path value is better than the E2E constraint, the computed values for connection parts should be used in the next step as $w^j_{i,best}$, otherwise as $w^j_{i,worst}$.
5) We propose to limit the number of optimization steps. If the number of maximal optimization steps is reached, the latest $w^j_{i,best}$ for each connection part should be used as an approximation value. If the amount of the maximum optimization steps is not reached yet, this procedure shall be repeated beginning with step (2).

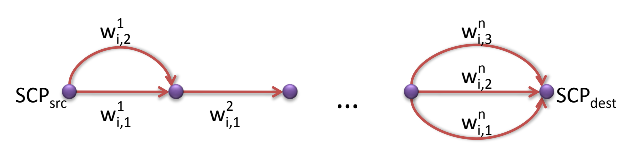


Figure 5. Possible gradation of values for different path segments for property i [1]

Please note that in order to reflect the ”better/worse” comparison instead of ”smaller/bigger” one, we define the unary operator ”$\lfloor\ \rfloor$” as follows: the result should be the worst realizable value which is equal or better than the value enclosed in the brackets.

V. Application to Search Problems

In Figure 6, we present a path finding algorithm, which illustrates the usage of our operators. In the pseudo-code, a Deep First Search (DFS) strategy is used for finding a path complying with multiple QoS constraints $\overrightarrow{C}^{E2E}$.

MCP (*nodeCur*, *nodeDest*, $\overrightarrow{W}^{path2cur}$, $\overrightarrow{C}^{E2E}$)

if (nodeCur == nodeDest)
  BacktracePath (*nodeCur*);
  return TRUE;
end if

MarkNode (*nodeCur*);

for each neighbor *nodeNbr* of *nodeCur*
  if (not Marked (*nodeNbr*))
    $\overrightarrow{W}^{path2nbr} = \overrightarrow{Aggr}\ (\overrightarrow{W}^{path2cur}, \overrightarrow{Best}(\overset{\rightrightarrows}{W}^{cur2nbr}))$

    if ($\overrightarrow{W}^{path2nbr} \prec \overrightarrow{C}^{E2E}$)
      if (MCP(*nodeNbr*, *nodeDest*, $\overrightarrow{W}^{path2nbr}$, $\overrightarrow{C}^{E2E}$))
        BacktracePath (*nodeCur*);
        return TRUE;
      end if
    end if
  end if
end for

UnmarkNode (nodeCur);
return FALSE;

Figure 6. Use of the new operators in a path finding algorithm [1]

The presented algorithm solves the so-called *multi constrained path finding* (MCP) problem. The function requires four parameters. The first two parameters ($nodeCur$ and $nodeDest$) specify nodes in the graph, between which a path has to be found. As the MCP function is called recursively, the $nodeCur$ specifies the end of the intermediately considered path. The weight of the intermediate path is given in the third parameter $\overrightarrow{W}^{path2cur}$. Finally, $\overrightarrow{C}^{E2E}$ are always the E2E-constraints between two endpoints.

The function first checks whether the destination node is reached yet. If it is the case, the $BacktracePath$ function is called in order to memorize the node in the path between two endpoints. Then the value $TRUE$ is returned which signals that a path with acceptable properties has been found.

If the end node is not yet reached, the $nodeCur$ is marked using the function $MarkNode$. This is a common practice in DFS-algorithms in order to prevent loops. In the following $for\ each$ loop all neighbors of $nodeCur$ are considered that have not been marked. For each neighbor $nodeNbr$ a weight $\overrightarrow{W}^{path2nbr}$ of an path between start and $nodeNbr$ nodes is computed. Corresponding to Section IV-A, the best possible value of the considered segment weight $\overset{\rightrightarrows}{W}^{cur2nbr}$ is aggregated with the intermediate sum $\overrightarrow{W}^{path2cur}$. If the computed weight of the new intermediate path is still better than E2E-constraint $\overrightarrow{C}^{E2E}$, the MCP function is called recursively. This time, $nodeNbr$ is used to mark the end of the intermediate path. If the function returns TRUE, the node is saved in order to backtrace the path; subsequently TRUE is returned. If the call to the MCP function was not successful, the next neighbor has to be considered likewise. If all neighbors have been considered without any success, the node $nodeCur$ is unmarked and the value $FALSE$ is returned.

Please note that for the sake of simplicity in this algorithm at most one connection between two nodes is supported. An extension for multigraphs would require an additional loop for all edges between two interconnected nodes. Furthermore, also the backtracking function should be extended in this case, in order to track not only nodes along the path, but also along used edges.

VI. Aggregation of Derived QoS Parameters

The comparison of the derived QoS parameters does not differ from the one of basis QoS parameters. The reason is that the derived QoS parameters are commonly defined as values that have to be compared with the defined thresholds. However, the aggregation of the values of derived QoS parameters is significantly more complex.

In the organizational domain of a single service provider, all basis QoS parameters required for the computation of derived QoS parameters are available. Therefore the derived QoS parameter – from the perspective of this provider – can be calculated end-to-end. Consequently, there is no necessity to aggregate values of derived QoS parameters. However, this is not applicable to multi-domain network connections. Figure 7 presents an end-to-end connection crossing organizational boundaries of three providers and consisting of five connection segments.

The complexity of the aggregation can be demonstrated using the QoS parameter *Availability* typically found in *Service Level Agreements* (SLA). This QoS parameter is computed as *up-time*, i.e., the time during which the service was fully operational, divided by *total-time*, i.e., the time during which the availability should be measured. If in the example depicted in Figure 7 all connection segments are unavailable for one hour, the availability of each single segment is 95.83%. However, the availability of the whole end-to-end connection cannot be exactly computed based on these values. Instead, the up-time of the whole connection needs to be calculated first. If we abstain from the consideration of the possible time deviations among different domains, the total time of the whole connection need not have to

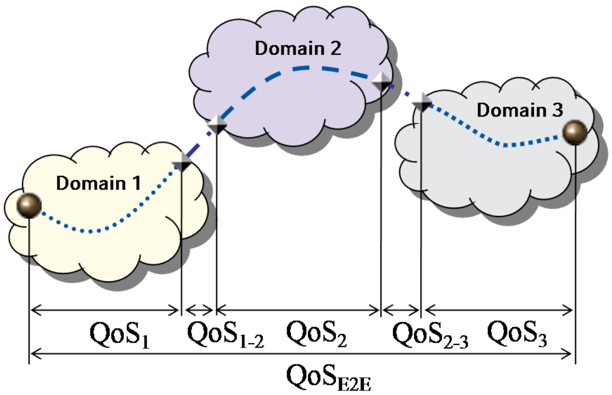


Figure 7. Composition of E2E connection quality [2]

be computed, instead it can be used as defined in the SLA. Returning to up-time calculation, this is again a derived QoS parameter, which can only be computed as the length of the intersection of up-times in all involved segments.

Generalizing our previous discussion, the values of derived QoS parameters can be seen as a projection from an m-dimensional vector of underlying (basis or derived) QoS parameters to the single value of the derived QoS parameter:

$$\text{Derive}_{QoS_{derived}} : QoS^m_{base} \rightarrow QoS_{derived}$$

Please note that the values of QoS parameters have not necesserily be the real numbers, e.g., up-time periods can be described as an conjunction of contiguous up-time periods, which in turn can be described based on the start and end time of this period. Further, the projection from the source to the target spece depends on the QoS_{dest}, i.e., the exact formula how the derived QoS value is calculated from the underlying QoS parameters. Furthermore, the destination QoS parameter is not necessarily one of the source QoS parameters, but we do not restrict this case.

In order to aggregate derived QoS values of two connection segments, we first have to calculate the aggregate of the underlying QoS values. This means that we have to access the values used in the $Derive$ projection. We call this access to underlying values $Base$:

$$\text{Base}_{QoS_{derived}} : QoS_{derived} \rightarrow QoS^m_{base}$$

Please note that the $Base$ operation cannot be realized on values alone, as the $Derive$ projection applied on various inputs in the source space can produce identical results in the target space. For instance, the mentioned availability of 95.83% can be achieved regardless at which time the service was unavailable for 1 hour; moreover, the unavailability time should not be a single continuous time period.

The necessity for the Base operation raises the requirements on the information model, which is used for the representation of the values of the various QoS parameters. In the case of derived QoS parameters not only the derived value itself, but also the underlying values used for their computations should be represented. Furthermore, such *derived/base*-relations should be recursive, with the leafs of the QoS-derivation-tree representing only basis QoS parameters.

Based on our previous discussion, we refine the function for the aggregation of two values belonging to derived QoS as follows:

$$Aggr(q, v) ::= \begin{cases} Aggr^{basis}(q, v), & \text{if } q, v \text{ are basis QoS} \\ Aggr^{derived}(q, v), & \text{if } q, v \text{ are derived QoS} \end{cases}$$

$$Aggr^{derived}(q, v) ::= Derive(\overrightarrow{Aggr}(Base(p), Base(q)))$$

where $Base$ and $Derive$ operations are specific to the QoS parameter of q, v, and the result of their aggregation. The vector aggregation $\overrightarrow{Aggr}$ is defined in Section III-C.

Please note that in opposite to our previous discussion these are not precise mathematical formulas that can immediately be applied on values. Instead these pseudo-formulas specify the order of computations, or the order and encapsulation of function calls as they are understood in computer science. For aggregation of two values of the same QoS type one has to distinguish first whether these QoS parameters are basis ones or derived ones. If they are basis ones, the aggregation of the values can be done according to the mathematical formula specified for the particular QoS in the registration tree. If they are derived ones, a more complex computation of the aggregated value is required. This include that the basis values of the derived QoS parameter should be accessed first. As the $Base$ operation can result in one or more underlying QoS parameters, the aggregation of two vectors is required. After aggregation of the underlying QoS values the values of the resulting vector have to be used in the formula for the computation of the derived QoS parameter.

Please note that following the $Base$ operation, a vector aggregation was used. This means that according to the definition of vector aggregations in Section III, this results in the aggregation of vector elements, which are values of either basis or derived QoS parameters. Consequently, again either basis or derived aggregation rules have to be applied, which may result in further $Base$ operations, until the aggregation on the values of the underlying basis QoS parameters can be performed. This can be seen as the navigation in the QoS parameter *derivation tree*.

Please note further that the distinction between aggregation of basis and derived QoS parameters determines how the aggregated value have to be computed. Regardless of the QoS class, the aggregation function has to be associated with its ID as proposed in Section III.

VII. Information Model supporting Basis and Derived QoS Parameters

As discussed above, the distinction between different QoS parameters can be realized based on their globally unique IDs. In [2], we have introduced an advanced information model, which can be used to describe available network connection segments as well as their properties, which includes various combinations of qualitative and quantitative QoS parameters as well as of the available management functionality (see Figure 8). However, the information model in its original form is not sufficient for the description of derived QoS parameters, as the computation on them, as defined in Section VI, requires the knowledge of the basis properties, from which the particular QoS parameter is derived.

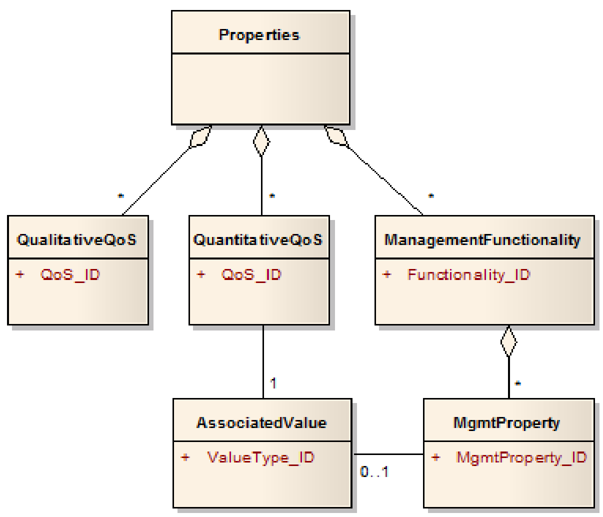


Figure 8. Connection segment properties [2]

In order to support the derivation of QoS parameters from each other, we extend this model with an additional view (see Figure 9). We define that every QoS parameter is derived from a basis class PROPERTY. The only purpose of this class is to show that every property can be either standalone, i.e., basis QoS, or derived from one or more other properties, i.e., derived QoS. The relation between derived and underlying QoS parameters is specified as a reflexive aggregation of the PROPERTY class. This aggregation called DERIVEDFROM has the cardinality $*$, which means that the same class can be used for the description of both basis and derived QoS parameters. Further, this aggregation involves the important operations $Base$ and $Derive$, as they have been defined in Section VI.

Please note that the proposed information model is highly extensible. For instance, it can be easily extended to support properties of energy consumption which are relevant, e.g., for energy efficient routing. The aggregation and comparison rules for such parameters can be defined as we have already

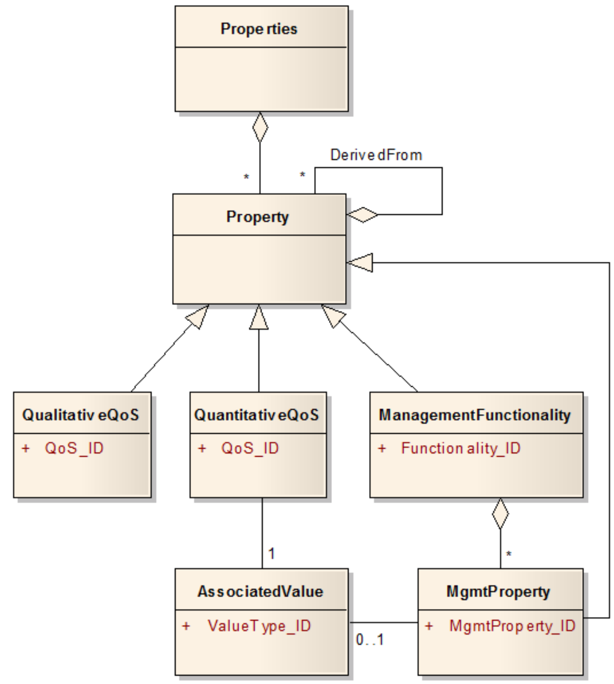


Figure 9. Deriving properties from each other

described for the QoS parameters. As the focus of the particular article lies on the function model, we abstain from the further discussion of the information model.

VIII. Putting it all together in an example

In order to illustrate the practicability of the proposed solution we refer to the situation of an E2E connection consisting of multiple segments (see Figure 7). As mentioned above, the aggregation is always needed during the routing process; it might also be needed for the monitoring of established connections. The latter is only necessary if no end-to-end measurements are possible and only connection segments can be monitored instead, which, however, is a very common situation in multi-domain connections. Further we only consider the situation that properties of different connection segments have to be aggregated together.

For the sake of simplicity, we say that only two QoS parameters (one basis and one derived QoS) are relevant for the connection: bandwidth and availability. The information about the connection segment's properties has to be structured according to the information model we have defined in Section VII. Currently, it is common that the components of distributed architectures communicate with each other via *web services*. This includes that the communication artifacts are encoded in an XML format. This is fully sufficient for the illustration purpose, as the XML format is human readable.

An XML schema can be derived directly from the defined information model. The choice of XML has an additional advantage, as it enables the description of service- or customer-requirements-adjusted combinations of supported connection properties. Please note that we intentionally avoid an explicit definition of the XML schema, as different realizations might be optimized for different purposed, e.g., for better parser performance or for human-readability.

One possible XML representation of the bandwidth and availability values of a single connection segment is depicted in Figure 10. In the XML code, the $Properties$ element encloses all properties of a segment, which are in this particular case two $QuantitativeQoS$ elements of the mentioned QoS parameters that are identified through their QoS_ID. Bandwidth belongs to the basis QoS class and is therefore structured very simple; it contains only the value and its metrics. The availability belongs to the class of derived QoS parameters. Therefore, for this parameter also basis values should be provided, which have been used for the calculation of the availability, i.e., in this case properties with IDs $UpTime$ and $TotalTime$. Please, note that also for these values properties are provided, from which they are derived.

```xml
<Properties>
  <QuantitativeQoS QoS_ID="Bandwidth">
    <AssociatedValue ValueType_ID="SingleValue">
      <SingleValue Value="1" Metric="Gbps"/>
    </AssociatedValue>
  </QuantitativeQoS>

  <QuantitativeQoS QoS_ID="Availability">
    <AssociatedValue ValueType_ID="SingleValue">
      <SingleValue Value="95.83" Metric="%"/>
    </AssociatedValue>
    <DerivedFrom>
      <QuantitativeQoS QoS_ID="UpTime">
        <AssociatedValue ValueType_ID="SingleValue">
          <SingleValue Value="23" Metric="Hrs"/>
        </AssociatedValue>
        <DerivedFrom>
          <AssociatedValue ValueType_ID="TimeRange">
            <TimeRange Begin="00:00" End="23:00" Metric="GMT"/>
          </AssociatedValue>
        </DerivedFrom>
      </QuantitativeQoS>
      <QuantitativeQoS QoS_ID="TotalTime">
        <AssociatedValue ValueType_ID="SingleValue">
          <SingleValue Value="23" Metric="Hrs"/>
        </AssociatedValue>
        <DerivedFrom>
          <AssociatedValue ValueType_ID="TimeRange">
            <TimeRange Begin="00:00" End="24:59" Metric="GMT"/>
          </AssociatedValue>
        </DerivedFrom>
      </QuantitativeQoS>
    </DerivedFrom>
  </QuantitativeQoS>
</Properties>
```

Figure 10. Example, basis and derived QoS parameters in XML

As discussed above, for the aggregation of values belonging to different connection properties, the property-relevant aggregation function should be used. According to our proposal, the definition of such functions should be associated with the property IDs in the registration tree. We assume that the QoS_IDs specified in XML file are sufficient for the unambiguous identification of QoS parameters and for access to the associated functions. Please note that the realization will require URNs with structure reflecting paths in the registration tree. Such functions can be defined, e.g., as the set of computation rules which should be executed or as a module of some interpreting programming language, which can be applied on demand.

For presentation purposes only, we define the aggregation functions for the example's QoS parameters as a pseudo-code (see Figure 11). All functions presented in the figure should be accessed from different locations in the registration tree. The aggregation rule for the bandwidth is very simple – it returns the smaller of to the two values. The rule for the calculation of the availability is more complex and follows the three steps described above. First, the basis values are obtained, which have been used for the calculation of the values p and q. Second, the aggregation function is executed on these basis values. Please note that the used aggregation functions correspond to the QoS parameters they are executed upon. Third, based on the results of the calculation in the previous step, the combined availability of two segments is calculated.

```
// Agggregation function for basis QoS "Bandwidth"
Bandwidth_Aggregate (p, q)
{
      if (p < q) return p;
      return q;
}

// Agggregation function for derived QoS "Availability"
Availability_Aggregate (p, q)
{
      // 1. Basis-operation for values
      //    Get basis values/value-vectors of derived QoS parameters
      p_base = Availability_Base (p);
      q_base = Availability_Base (q);

      // 2. Vector-aggregation of underlying QoS parameters
      //    Use for this purpose function defined for the basis QoS parameters
      //    In this case QoS parameters are time interval durations
      UpTime    = TimeDuration_Aggregate(p_base.UpTimeLength, p_base.UpTimeLength);
      TotalTime = TimeDuration_Aggregate(p_base.TotalTimeLength, p_base.TotalTimeLength);

      // 3. Derive-projection to the derived QoS "availability"
      Availability = UpTime / TotalTime;

      return Availability;
}

// Agggregation function for derived QoS "TimeDuration"
TimeDuration_Aggregate (p, q)
{
      // 1. Basis-operation for values
      //    Get basis values/value-vectors of derived QoS parameters
      //    In this case these are lists of all time intervals
      p_base = TimeDuration_Base (p);
      q_base = TimeDuration_Base (q);

      // 2. Vector-aggregation of underlying QoS parameters
      //    Use for this purpose function defined for the basis QoS parameters
      //    In this case QoS parameters are lists of time intervals
      TimeIntervalIntersection =
            TimeIntervalList_Aggregate (p_base.TimeIntervals, q_base.TimeIntervals);

      // 3. Derive-projection to the derived QoS "TimeDuration"
      JointTimeIntervalDuration = 0;
      for all TimeInterval in TimeIntervalIntersection
            JointTimeIntervalDuration += TimeInterval.IntervalDuration;

      return JointTimeIntervalDuration;
}

// Agggregation function for basis QoS "TimeInterval"
TimeIntervalList_Aggregate (p, q)
{
      // compute intersection of the all intervals in the interval lists "p" and "q"
      return IntervalListIntersection (p, q);
}
```

Figure 11. Pseudo-code for aggregation functions

The access to and execution of the comparison function is similar to the aggregation one. Therefore, we omit its explicit treatment. Instead we would like to discuss the disadvantage of the computational procedure we have defined for the aggregation of derived QoS parameters. Even though the defined procedure leads to the desired results, it requires $Base$ and $Derive$ operations every time values of two segments have to be aggregated. This means that for a connection with n segments, such operations should be done $2 * (n-1)$ times ($n-1$ pairs of 2 segments every time). We see a big potential for the improvement of this situation through the use of dynamic programming in the definition of aggregation functions. However, the exact optimizations have to be analyzed case by case for each supported derived QoS parameter.

IX. Conclusion and Future Work

In this article, we have defined a novel schema for the generic treatment of network connection properties. In order to support operations on arbitrary properties of network connections, we propose to associate five functions with the ID of each supported property. These functions are summarized in Table I. Three of these functions, which are used for property aggregation and comparison, are mandatory. The mandatory function _AGGREGATE_LINKPART is dedicated to compute the property of connection based on only partial views at the same inter-domain connection. For elaborated discussion about its necessity we refer to [2]. The remaining selection functions aim to simplify handling with value ranges. These functions are not mandatory, as they can be easily derived based on comparison function. Additionally to the mentioned five functions, for derived QoS parameters two operations $Base$ and $Derive$ should be defined. They can be either defined implicitly as a part of mentioned aggregation functions or explicitly as functions associated with the QoS ID in the registration tree.

Function class	Purpose
_COMPARE	Compare two values *a* and *b*. Result can be: "*a* is better", "*a* is worse", "*a* and *b* are equivalent"
_SELECT_BEST	Optional function returning the best value of a given value set
_SELECT_WORST	Optional function returning the worst value of a given value set
_AGGREGATE_LINKS	Aggregate property values of two links or paths
_AGGREGATE_LINKPARTS	Aggregate two partial views at the same link to a single link weight

Table I
FUNCTIONS FOR OPERATIONS ON A SINGLE QOS PARAMETER

Together with [2] and [3], which present an information model and a multi-domain routing procedure, the solution presented here is an integral part of our ongoing work enabling user-tailored connection services with guaranteed E2E quality. However, the generic operation handling proposed in this article is not restricted to the problem space described here. It can be used in alternative routing algorithms that are considering multiple properties, such as [5] and [6].

The presented proposal leaves some aspects unsolved, they will be addressed in further research as follows: In the first place, a meta-language for the description of property-related functions has to be selected; also, a concrete structure for the registration tree has to be proposed. In order to achieve this, a profound evaluation of alternatives is needed. In the case that a single global registration tree has to be used by multiple organizations, like it is the case for the internet registration tree, the description of equivalence relationships between different entries has to be addressed. Furthermore, the quality parameters of different network layers as well as user-faced services depend on the quality of the underlying layers they are realized upon. Therefore, a general description of such interdependencies and parameter transformations is essential in order to offer customer-demanded quality based on network-specific information.

Additionally, we plan to investigate the applicability of our approach on the services with other composition structures. For instance, the quantification of survivability capabilities requires conditional computations [13] leaving the strictly sequential compositions in favor of more flexible procedures including branches, loops, and cases.

Acknowledgment

The authors wish to thank the members of the Munich Network Management Team (MNM Team) [14] for fruitful discussions and valuable comments on previous versions of this paper. The MNM Team directed by Prof. Dr. Dieter Kranzlmüller and Prof. Dr. Heinz-Gerd Hegering is a group of researchers at Ludwig-Maximilians-Universität München, Technische Universität München, the University of the Federal Armed Forces and the Leibniz Supercomputing Centre of the Bavarian Academy of Science.

References

[1] M. Yampolskiy, W. Hommel, D. Schmitz, and M. K. Hamm, *Generic Function Schema for Operations on Multiple Network QoS Parameters*, Proceedings of The Second International Conference on Advanced Service Computing (SERVICE COMPUTAION 2010), pp. 126–131. Valencia, 2010.

[2] M. Yampolskiy, W. Hommel, P. Marcu, and M. K. Hamm, *An information model for the provisioning of network connections enabling customer-specific End-to-End QoS guarantees*, Proceedings of 7th IFIP/IEEE International Conference on Services Computing (SCC 2010), pp. 138–145. Miami, 2010.

[3] M. Yampolskiy, W. Hommel, B. Lichtinger, W. Fritz, and M. K. Hamm, *Multi-Domain End-to-End (E2E) Routing with multiple QoS Parameters. Considering Real World User Requirements and Service Provider Constraints*, Proceedings of The Second International Conference on Evolving Internet (INTERNET 2010), pp. 9–18. Valencia, 2010.

[4] F. A. Kuipers, *Quality of service routing in the internet: Theory, complexity and algorithms*, PhD thesis. Delft University Press, 2004.

[5] T. Korkmaz and M. Krunz, *Multi-constrained optimal path selection*, Proceedings of Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2001), pp. 834–843. 2001.

[6] P. Van Mieghem, H. De Neve, and F. A. Kuipers, *Hop-by-hop quality of service routing*, Computer Networks, pp. 407–423. Elsevier, 2001.

[7] M. Ziegelmann, *Constrained Shortest Paths and Related Problems*, PhD thesis. VDM, 2007.

[8] R. Bellman, *The theory of dynamic programming*, Proceedings of the National Academy of Sciences of the United States of America, pp. 716–719. 1952.

[9] G. Bertrand, S. Lahoud, M. Molnar, and G. Texier, *Inter-Domain Path Computation with Multiple Constraints*. 2008.

[10] A. Avizienis, J.-C. Laprie, B. Randell, C. Landwehr, *Basic Concepts and Taxonomy of Dependable and Secure Computing*, IEEE Transactions on Dependable and Secure Computing, VOL. 1, NO. 1, Jan-Mar 2004, pp. 11-33

[11] V. Lloyd, C. Rudd, ITIL Service Design. The Stationery Office, 2007, ISBN 9780113310470

[12] TMForum, SLA Management Handbook, Release 2.5, GB917, 2005

[13] M. Schiffers, D. Kranzlmüller, *Folded Interaction Systems and their Application to the Survivability Analysis of Unbounded Systems*, Proceedings of 33th International Conference on Information Technology Interfaces (ITI 2011), pp. 97–102, Dubrovnik, Croatia, 2011

[14] *Munich Network Management Team (MNM Team) Homepage*, [Online: http://www.mnm-team.org], August 2010.

Community Tools for Massively Multiplayer Online Games

Shakeel Ahmad*, Christos Bouras†, Raouf Hamzaoui*, Jiayi Liu§, Andreas Papazois†, Erez Perelman‡, Alex Shani ‡, Gwendal Simon §, George Tsichritzis †

*Department of Engineering
De Montfort University, Leicester, UK
sahmad@dmu.ac.uk, rhamzaoui@dmu.ac.uk

†Computer Technology Institute & Press "Diophantus"
N. Kazantzaki, GR26500 Patras, Greece
bouras@cti.gr, papazois@ceid.upatras.gr, tsixritzis@cti.gr

§Institut Telecom
Telecom Bretagne, France
jiayi.liu@telecom-bretagne.eu, gwendal.simon@telecom-bretagne.eu

‡Exent Technologies
Bazel 25, 49125 Petach-Tikva, Israel
eperelman@exent.com, ashani@exent.com

***Abstract*—One of the most attractive features of Massively Multiplayer Online Games (MMOGs) is the possibility for users to interact with a large number of other users in a variety of collaborative and competitive situations. Gamers within an MMOG typically become members of active communities with mutual interests, shared adventures, and common objectives. We present the EU funded Community Network Game (CNG) project. The CNG project provides tools to enhance collaborative activities between online gamers and offers new tools for the generation, distribution and insertion of user-generated content in MMOGs. CNG allows the addition of new engaging community services without changing the game code and without adding new processing or network loads to the MMOG central servers. The user-generated content considered by the CNG project includes 3D objects and graphics, as well as screen-captured live video of the game, which is shared using peer-to-peer technology. We survey the state of the art in all areas related to the project and present its concept, objectives, and innovations.**

***Keywords*-Massively Multiplayer Online Games; user generated content; P2P video streaming; graphics insertion.**

I. Introduction

Massively Multiplayer Online Games (MMOGs) allow a large number of online users (in some cases millions) to inhabit the same virtual world and interact with each other in a variety of collaborative and competing scenarios. MMOG gamers can build and become members of active communities with mutual interests, shared adventures, and common objectives. Players can play against other players (player versus player) or build groups (guilds) to compete against other groups (realm versus realm) or against computer-controlled enemies. MMOGs are rapidly gaining in popularity. According to IDATE [1], the number of MMOG players worldwide is expected to grow from 82 million in 2008 to more than 140 million by 2012.

This paper presents the Community Network Game (CNG) project [2], an EU funded project within the Seventh Framework Programme. The project, which started in February 2010 and has a duration of 30 months, aims at enhancing community activities for MMOG gamers. This is achieved by providing Web collaboration tools and developing new tools for the generation, distribution and insertion of User-Generated Content (UGC) into existing MMOGs. This UGC may include textures and 3D objects to be added to the game, live video captured from the game screen and streamed to other players, as well as videos showing walkthroughs, game tutorials, or changes in the virtual world to be watched on demand.

The main technologies proposed by the CNG project are the In-game Graphical Insertion Technology (IGIT) and a Peer-to-Peer (P2P) system for the distribution of live video. IGIT is an innovative technology of replacing or inserting content into a game in real time without the need to change the game code in the client or server. For example, billboards can be inserted, tattoos can be added to in-game characters, an area on the screen can be assigned to display user information, and any type of window (browser, chat, etc.) can be inserted floating on or outside the game area. The technology can be implemented on multiple games, making it possible to create a community that is not limited to a specific game or publisher.

Enabling thousands of users to communicate UGC represents a significant challenge to networks already occupied with the MMOG client-server data. The CNG project develops new techniques for UGC distribution that are "friendly" (supportive and not disruptive) to the MMOG client-server traffic. The key innovation is a P2P system that allows MMOG gamers to stream live video of the game without interrupting the MMOG data flow and without the need to upload the video data to a central server.

The remainder of the paper is organized as follows. Section II gives an overview of the state-of-the art in the areas of UGC, Web collaboration tools, P2P live video streaming, and game adaptation technologies. Section III presents the project's concept, objectives and technologies. Section IV concludes the paper by discussing the project's benefits and expected impact. The paper is an extension of the conference paper [3].

II. Related Work

In this section, we review the state-of-the art in the areas of UGC, Web collaboration tools, P2P live video streaming, and game adaptation technologies.

A. UGC

UGC includes various kinds of media content produced by end-users. In a game context, for example, this may be screen-captured video. Another example of UGC is the various mods created by the users. Sharing and remixing UGC is a widespread online activity that crosses borders of age and gender. Avid players go to great lengths in their efforts to create shared content in which they reveal their mastery. Additional data layers are always included: narration, animation and primarily soundtrack. Most MMOG-based UGC content is confined to dedicated game company sites as in World of Warcraft [4]. Many MMOG games also have their own community pages in social networking sites such as Facebook. In April 2010, Facebook released significant updates to its API by allowing external websites to uniformly represent objects in the graph (e.g., people, photos, events, and community pages) and the connections between them (e.g., friend relationships, shared content, and photo tags). As a result, the Facebook API [5] can provide an unprecedented bridge between gamespaces and the social web.

Current UGC tools can be classified into three categories: tools for capturing, for editing, and for uploading/broadcasting.

Capturing: Capturing videos can be done within the gamespace as in Spore [6]. This is not a common feature in MMOGs. More commonly, capturing is done with external video capture software such as Camtasia [7] or Fraps [8]. Fraps is the preferred software for users who want to capture high quality video. However, its free version has a 30 s recording limitation.

Editing: UGC sharing and remixing within game platforms is currently not supported. To edit the video and add effects, narration, soundtrack and text overlays, users tend to use readily available software such as Windows Movie Maker for Windows or iMovie for Mac that allow for the inclusion of additional content: audio, images and other videos. Annotations can only be added after the capture is done and cannot include other participants' comments.

Uploading/broadcasting: Once a user has captured and edited the video, a final step is needed to upload it for viewing. Many MMOG players use sites such as YouTube to share their game-based UGC. In 2008, Maxis incorporated YouTube APIs within their game, Spore, enabling players to upload videos of their creations to their YouTube account with only two clicks [9]. The collaboration between YouTube and a game creator (Electronic Arts), including revenue share from advertisements, is unique to date. Players of other games need to upload their video creation from their computer and cannot do it from within the game itself [10].

A user can capture the video of the game and broadcast it live to other users via a video server. This feature is offered by Xfire [11], which allows anyone to watch a live feed of a user's game screen. When a user begins a stream, a chat room is opened that anyone watching the live feed can join.

B. Web collaboration tools

Web 2.0 based collaboration applications may include instant messaging, audio and video chat, file sharing and online voting and polling. For audio/video capturing and playback the Flash software platform [12] is commonly deployed. Other solutions are the Java Applet technology or standalone applications which run on a Web browser and offer interoperability over different platforms. For instant messaging, online polling/voting and file sharing, Asynchronous JavaScript and XML (AJAX) [13] are commonly used. AJAX allows Web applications to retrieve data from the server asynchronously in the background without interfering with the display and behavior of the existing page. The use of AJAX techniques has led to an increase in interactive interfaces on webpages. Finally, for WWW client-server communication, most of the Web 2.0 applications are based on Simple Object Access Protocol (SOAP) [14]. SOAP relies on XML as its message format, and usually relies on other Application Layer protocols, most notably the Remote Procedure Call (RPC) and HTTP.

In the following, we give examples of Web 2.0 collaboration software.

1) Instant messaging: Instant messaging software is mainly based on AJAX technology. A typical AJAX chat application uses a database (MySQL) and AJAX to store and retrieve the users' messages and pass them between the client and the server. Examples of instant messaging software include AJAX Chat [15], Google Talk [16], ChatZilla [17], Mibbit [18], and Java/JavaScript Chat [19].

2) File sharing: Examples of popular Web2.0 file sharing systems include Meebo [20], iGoogle [21], Orkut [22], and FlashComs Community chat [23].

3) Audio and video chat: Audio and video chat applications are based on the Flash Platform. Some typical examples of Web-based audio and video chat tools are AVChat 3 [24], Red5Chat [25], MeBeam [26], Web Voice Chat [27], and 123 Live Help Chat Server Software [28].

Table I: POPULAR CHAT TOOLS.

Tools	Instant Messaging	Audio and video chat	File sharing	Protocols
CGI:IRC	Perl/CGI	Not supported	Not supported	IRC
PJIRC	Java Applet	Not supported	Not supported	IRC
qwebirc	Ajax Applet	Not supported	Not supported	IRC
Parachat	Java Applet	Not supported	Not supported	Jabber/XMPP
Pichat	Ajax	Not supported	Not supported	Unknown
Facebookchat	Ajax	Not supported	Not supported	Jabber/XMPP
eBuddy	Ajax	Not supported	Not supported	Jabber/XMPP
Omegle	Ajax	Flash	Not supported	Jabber/XMPP
webcamnow	Ajax	Flash	Not supported	Jabber/XMPP
JatChat	Java Applet	Java Applet	Not supported	Jabber/XMPP
campfire	Ajax	Not supported	Ajax	Unknown
Single Operator Ajax chat	Ajax	Not supported	Ajax	Unknown

Table I lists some widely used chat tools together with their underlying technology.

Table II: POPULAR VOTING AND POLLING APPLICATIONS.

Tools	Technology
Poll4Web	Flash
Flash Web Poll	Flash
ABPollMaster Polling	Java Applet
Fly06 Poll	Ajax

Table III: POPULAR BLOGWARES.

Tool	Technology
Kontain	Flash
Blogsmith	Ajax
TypePad	Ajax
Gawker bespoke software	Ajax

4) Online voting and polling: Examples of Web-based collaborative voting and polling tools are VotingPoll [29], DPolls [30], and XML Flash Voting Poll [31]. Table II lists other examples and the technology used for their implementation.

5) Blogging: Important tools used for the building of online blogging applications are WordPress [32], and Movable Type [33]. Table III lists popular blogwares.

C. P2P live video systems

Traditional client-server video streaming systems have critical issues of high cost and low scalability on the server. P2P networking has been shown to be cost effective and easy to deploy. The main idea of P2P is to encourage users (peers) to act as both clients and servers. A peer in a P2P system not only downloads data, but also uploads it to serve other peers. The upload bandwidth, computing power and storage space on the end user are exploited to reduce the burden on the servers.

Viewers of a live event wish to watch the video as soon as possible. That is, the time lag between the video source and end users is expected to be small. In a live streaming system, the live video content is diffused to all users in real time and video playback for all users is synchronized. Users that are watching the same live video can help each other to alleviate the load on the server. P2P live streaming systems allow viewers to delete the historic data after the playback, and hence have no requirement for any data storage and backup.

Based on the overlay network structure, the current approaches for P2P live streaming systems can be broadly classified into two categories: tree-based and mesh-based. In tree-based systems, peers form an overlay tree, and video data are pushed from the parent node to its children. However, a mesh-based system has no static streaming topology. Peers pull video data from each other for content delivery. Over the years, many tree-based systems have been proposed and evaluated, however, never took of commercially. Mesh-based P2P streaming systems achieve a large-scale deployment successfully, such as PPLive [34], PPStream [35], etc.

1) Tree-based systems: Many early P2P streaming systems use a tree-based approach that is typically based on application-level multicast architectures. Tree-based systems, such as ESM [36] and P2Cast [37], organize peers into a tree structure for delivering data. The data are diffused following this well-defined structure, typically pushed from a peer to its children. Tree-based solutions are perhaps the most natural and efficient approach, but they face several challenges. One major drawback of tree-based systems is the system fragility due to peer churn. A peer departure will disrupt data delivery to all its descendants, particularly for the peers in the higher level of the tree. The high dynamicity of peers in a P2P network potentially deteriorates transient performance. Another drawback is the under-utilized upload bandwidth of the peers. The leaf nodes in the tree cannot contribute any upload bandwidth resource to the system. Since a majority of nodes are leaves in the tree structure, this significantly reduces the overall efficiency. To address the issues of leaf nodes, multi-tree structures were introduced [38], [39]. In a multi-tree system, the source encodes the stream into several sub-streams and diffuses each sub-stream along one sub-tree. Each peer participates in many or all

Table IV: TRANSPORT PROTOCOLS IN P2P LIVE VIDEO STREAMING SYSTEMS.

System	Protocol
CoolStreaming [41]	TCP
PPStream [65]	TCP
PPLive [66]	Combination of TCP and UDP
TVAnts [65]	Combination of TCP and UDP
Joost [67]	UDP and TCP with UDP being the dominant traffic
SopCast [66]	Combination of TCP and UDP
[49]	UDP with FEC
[53]	UDP with ARQ
[61], [62], [68]	UDP
GridMedia [69]	UDP
iGridMedia [70]	UDP
PULSE [71]	UDP for control messages and TCP for data exchange
R2 [72]	UDP or TCP when UDP cannot be used due to firewall blocking

sub-trees to retrieve sub-streams. Hence, a peer might be deployed on an intermediate position in one sub-tree or a leaf position in another sub-tree. Compared with the single-tree approach, the multiple-tree solution has two advantages. First, the system's robustness is improved, as the failure of a high-level node would not completely disrupt all its descendants. Second, the upload bandwidth of all nodes could be well utilized, since each node stands a good chance to be both a leaf and an intermediate node. However, since the multiple-tree approach is still a tree-based solution, the drawbacks of tree-based systems remain basically unsolved. First, the construction and maintenance of the multiple-tree structure are costly because of frequent peer churn behaviours. Second, the upload bandwidth contribution of a node, which depends on the position in each sub-tree, is deficient. Furthermore, the design involves overhead.

Viewers in P2P live streaming systems only focus on the live video data that currently are output from the source. Hence, the video playbacks for all users are synchronized. In tree-based P2P live systems [36], all users participating in a video streaming session can form a tree at the application layer with the root at the video source. Each peer receives the live video data from its parent and immediately forwards the data to its children. Usually peers at lower levels receive the live data after peers at upper levels. The major consideration is to balance the depth of the tree and the out-degree of the intermediate nodes. Multi-Tree based approaches for P2P live systems are described in [38].

2) Mesh-based systems: In a mesh-based P2P streaming system, peer relationships are built and terminated according to data availability and bandwidth availability on peers. A video is typically divided into many chunks. Moreover, a tracker server maintains the relationship between peers and video data. Then, a peer can dynamically connect to a peer list that is chosen randomly from the tracker server according to which chunk the peer requests. After that, the peer maintains multiple neighbours and exchanges chunks with these neighbours simultaneously. A gossip protocol [40] is typically used for the topology management. Peers also periodically exchange information of the chunk availability using a buffer map. Usually, a chunk is pulled by a peer from its neighbours who have the requested chunk. The pull policy can avoid redundant chunk transmission.

If a neighbour leaves, the peer can continue retrieving chunks from other neighbours. The peer also explores some new neighbours to keep a certain number of neighbours. Due to the maintenance of multiple neighbours, mesh-based systems are highly robust against peer churns and fully utilize users upload bandwidth. However, transmission delay presents a challenge to mesh-based systems (for example, long start-up delay and channel switching problems for live streaming systems).

Many successful P2P live streaming systems [34], [35], [41] use the mesh-based streaming approach. The design of mesh-based P2P live streaming systems is relatively simple. All users are interested in the same live data. All chunks downloaded at a peer are always useful to other peers that have not retrieved these chunks. Some studies investigate the quality of peering connections. Several strategies are proposed to construct the peer relationship. The first consideration is the workload and resource availability on both peers, such as the current number of connections, upload and download bandwidth, and system resources. Other considerations are the network condition, which includes end-to-end delay and loss rate, and the network proximity, including geographical position, bandwidth, delay and network distance.

3) Server-assisted P2P systems: Most peer-to-peer systems rely on a server, either a bootstrapping server or a tracker server. A bootstrapping server is only used when a new peer joins an overlay. The bootstrapping server is expected to give to this newcomer a list of peers that are currently in the system. In this way, the new peer can quickly open connections. It has been shown, however, that a popular P2P streaming system like PPLive fails to provide accurate information to newcomers, resulting in a too long start-up delay [42]. Actually, a large proportion of peers that are given by the bootstrapping server do not answer the initial request of the newcomer, either because they are no longer in the system, or because they do not need any new connections. A tracker server extends the bootstrapping function. Every peer periodically sends a message to the tracker, which gives in return a list of peers (peerlist). That is, the participants to a peer-to-peer system can discover new peers on a periodic basis. The bit-torrent system has popularized this hybrid architecture, which guarantees, among other suitable properties, that new peers can quickly find matching peers.

In general, implementations of tracker-based peer-to-peer systems are simple. The tracker sends to a requesting peer a list of randomly chosen peers among the set of peers that are expectedly active in the system. Interestingly, the resulting topology is a random regular graph: every peer is connected to a given number of randomly chosen other peers. This random-like underlying topology is interesting on several aspects, especially random regular graphs are connected with high probability (so, any information is accessible from any peer), and the diameter of a random regular graph is small (therefore any information is close to any peer, if it is able to find it).

This topology links "acquaintances". A peer can contact any subset of peers in the peerlist, but it is free to choose, among them, some privileged peers with which it will exchange data. This presents some problems. First, the peerlist contains peers exhibiting a broad scope of capacities, although the overlay tends to connect peers having similar characteristics [43]. The overlay would converge faster if the peerlist could contain preferentially the peers having the closest characteristics to the requester. However, it would require to authorize the tracker to determine as accurately as possible the capacity of peers, which appears to be impossible or costly in many cases. Second, the peerlist topology does not take into account the location of peers. Therefore the overlay wastes network resources.

4) Hybrid CDN-P2P systems: Peer-assisted (PAS) Content Delivery Networks (CDNs) have attracted a lot of attention in recent years. In this section, we present the architecture of a real-world CDN-P2P live video streaming system called LiveSky [44], which has been deployed in China. The system is designed to solve a set of problems in current CDN and P2P live video streaming systems such as scaling, fast startup and upload fairness.

Server side organization: The CDN overlay is constructed according to a tree-based structure, where leaves are edge servers, whose role is to serve end users. All other intermediate nodes are core servers, which are responsible for delivering content to edge servers. Because of their work load, edge servers are not allowed to transfer content between each other. To realize a P2P organization at the client side, an edge server has several roles: 1) a regular server for legacy clients; 2) a tracker for the P2P operation; 3) a seed in the P2P system.

Client side organization: There are two types of clients: legacy clients and P2P clients. Legacy clients are served in the traditional CDN manner and receive low quality streams. P2P clients are organized in a hybrid scheme proposed in [45], [46] that combines the multi-tree and mesh topologies. As usual a video is divided into several substreams. Each substream contains nonconsecutive frames. The peers that host the same substream compose a tree-based overlay. This ensures upload fairness of each peer. On the other hand, peers also use a mesh-style pull mechanism to retrieve missing frames for continuous playback. This enhances the robustness of the network. Moreover, P2P clients are allowed to access high quality videos.

Adaptive scaling and improvements: In the system, each edge server decides whether a new arrival client should be treated as a legacy client or a P2P client independently. A threshold is pre-configured in every edge server. When the number of clients is below the threshold, all clients retrieve the content directly from the edge server. If the number of clients exceeds the threshold, new arrival clients will be redirected to other clients to form a P2P organization. Both the threshold and the capacity of an edge server are calculated by some parameters, including the level of the P2P tree overlay, peer arrival rate, peer leaving rate and peer contribution rate. When an edge server reaches its capacity limitation, new clients will be redirected to a less loaded edge server.

Fast startup: Startup time is optimized in LiveSky in two ways. First, the buffer size is reduced to 15 seconds. Second, the first request of a client is always replied directly by an edge server, thus it is very quick to retrieve startup streams.

5) Video transmission protocols: In private, well-managed IP networks, the quality of service (QoS) is maintained by calibrating the end-to-end infrastructure. This is not possible in P2P overlays since they are built on open IP networks, which are best-effort in nature. Real-time video communication over P2P overlays on the public Internet mainly relies on the transmission control protocol (TCP). TCP guarantees reliable transmission of the data by automatic retransmission of lost packets. However, as TCP requires in order delivery of the data and keeps on re-transmitting a packet until an acknowledgement is received, significant delays may be introduced. Further delays are caused by the congestion control algorithm used by TCP, which reacts to packet loss by reducing the transmission rate, leading occasionally to service interruption. This presents a serious drawback for real-time video communication where the data must be available to the receiver at its playback time. Lost and delayed packets that miss their playback deadline not only are useless, they also consume the available bandwidth unnecessarily.

An alternative to TCP is to use UDP as the transport protocol and apply application-layer error control. For example, the Darwin Streaming Server, which is the open-source version of Apple's QuickTime Streaming Server, uses a simple timeout-based ARQ scheme [47]. The Helix DNA streaming system, which is the open-source version of RealNetworks Helix streaming suite, also uses timeout based ARQ [47]. Windows Media uses a selective retransmission scheme. If the client detects gaps in the packet sequence numbers, it sends a retransmission request to the server, which re-transmits the missing packets. Packet retransmissions are limited to a certain percentage of the available bandwidth and packets to be retransmitted are prioritized according to their content. Audio packets are given the highest priority. Video packets close to their playout deadlines are given the lowest priority, on the presumption that retransmissions are most likely to miss the playout deadline [48]. VideoLAN, a popular open-source streaming system, uses either TCP or UDP without packet loss mechanisms [47]. These streaming techniques are suitable for well-managed networks. However, they face considerable problems in open IP networks where the packet loss rate may be significant, and the available bandwidth may be variable. In P2P overlays, in particular, packet loss is not only due to congestion at routers but also to the heterogeneity in node stay-time duration.

Most P2P streaming systems use TCP (CoolStreaming, PPStream), a combination of UDP without error control and TCP (PPLive, TVAnts), UDP with Forward Error Correction (FEC) [38], [49], [50], [51], [52], and ARQ [53]. Another approach is Multiple Description Coding (MDC) [54], [55]. However, MDC schemes are rarely used in practice because they rely on non-standard video coders.

Thomos and Frossard [56] use network coding with rateless codes [57] for P2P video streaming. The technique exploits path diversity and lessens the burden of re-encoding on an intermediate forwarding peer.

Wu and Li [58] also use network coding based on rateless codes for P2P live video streaming. A peer can recover the original video source block by receiving enough encoded symbols from multiple receivers. As soon as a receiving peer successfully decodes the source block, it becomes a source and applies rateless coding on the decoded source block to generate encoded symbols for other peers. To avoid receiving redundant symbols, each peer uses a different seed for rateless encoding. The authors propose a distributed algorithm for best peer selection and optimize rate allocation to guarantee minimum delay.

Grangetto, Gaeta, and Sereno [59] propose an improvement to the method of Wu and Li [58]. In their method, called 'Relay and Encode' (RE), a receiving peer relays the received encoded symbols immediately. Once it decodes the source block, a rateless code is applied on the source block and newly produced encoded symbols are sent to its children. The authors show that RE has a lower delay than the method of [58]. However, the paper does not consider the effects of varying channel conditions and does not exploit feedback to minimize bandwidth usage. Moreover, the protocol is not robust against failures. For example, if one peer cannot decode the source block, all its descendent peers are affected. A similar system is proposed by the same authors in [60]. Here, receiver feedback is used to ask the sender to stop sending more symbols when the source block is decoded.

In [52], rateless coding is used to make a P2P VoD system resilient to peer churn. The source partitions the video into source blocks and applies rateless coding on these blocks. During a push phase, for each source block, distinct groups of encoded symbols are distributed among a number of volunteering peers, which may not be interested to watch the video. This pushing can be done during low network utilization time. In a pull phase, a peer who wants to watch the video needs to collect a minimum number of distinct encoded symbols from any subset of volunteering peers.

Setton, Noh, and Girod [61] propose a system for live video streaming over P2P networks aimed at low latencies and congestion avoidance. Video packets are sent using UDP/IP, and a scheduling algorithm is used to maximize the received video quality, while minimizing network congestion.

In [62], a P2P live video streaming system aimed at low delays is presented. The system uses the Stanford Peer to Peer Multicast Protocol to build multiple complementary multicast trees, all originating at the source. The source exploits path diversity and sends different packets over different multicast trees. The video is compressed with the Scalable Video Coding (SVC) and packets are sent over UDP/IP. Depending on the available bandwidth and packet loss rates, intermediate nodes decide how many layers should be sent to their children. A simple ARQ mechanism is used to deal with packet loss.

One limitation of the UDP protocol is the lack of a congestion control mechanism. Congestion control with UDP can be realized with the Datagram Congestion Control Protocol (DCCP) [63]. DCCP uses an Explicit Congestion Notification (ECN) bit, which is set on by a congested router. When a receiver receives a packet with an ECN bit set on, it asks the sender to react to the congestion accordingly. However, most routers disable ECN [64].

Table IV gives an overview of the transport protocols used in P2P systems.

D. Game adaptation technologies

In-game technologies have been used in the gaming market for several years. The gaming industry has adopted these technologies to increase its revenue by finding more financial sources and by attracting more users. In-game overlay allows to view and interact with windows outside the game, but without "Alt-Tabbing". It does so by rendering the window inside the game. Texture replacement enables to replace an original game texture with a different texture. In this way, the newly placed textures are seen as part of the original game content. This method is commonly used for dynamic

Table V: In-game tools. An x indicates that the tool is offered by the product.

Product	XFIRE	PLAYXPERT	Massive	FreeRideGames	Double Fusion	Steam	Overwolf	Raptr
Texture replacement			x		x			
Game resize				x				
In-game overlay	x	x		x		x	x	x
Video capture	x	x					x	
Video edit								
Video upload	x						x	
Live video	x							
Instant messaging	x	x				x	x	x
Audio chat	x					x		x
File sharing								
Online blogging								
Need for SDK	No	No	Yes	No	Yes	Yes	No	No

in-game advertisement. Game size modification technology adapts the original game by decreasing its original size and surrounding it with an external content.

Some of these technologies are distributed as an external utility that can overlay a pack of games while others are part of MMOG service features which are provided to the users.

The main available in-game adaptation products on the market are Xfire [11], PLAYXPERT [73], FreeRidesGames [74], Massive [75], Double Fusion [76], Steam [77], Overwolf [78], and Raptr [79]. Some of the products require for the game developer to integrate the product's Software Development Kit (SDK) (for each game to be developed the game developer must use the products SDK). The use of those in-game adaptation products is not available for the existing games catalogue. Table V lists the in-game tools surveyed in this paper, showing those offered by existing products.

III. Technologies and Innovations

Fig. 1 shows the CNG architecture. While the MMOG architecture is not modified (the game content and the game data are still transferred through the MMOG servers), the following components are added: (i) Sandbox on the client side that is responsible for modifying the game environment; (ii) CNG Server for monitoring the P2P UGC communication. The CNG server acts as a tracker for the system in the sense that it is in charge of introducing peers to other peers. It has persistent communication with the clients and manages the organization of the P2P exchanges.

A. IGIT

IGIT enables the user to resize the game and surround it with external content, overlay the game, and replace an existing game texture with an external content. This is done in a way that does not harm the game experience and without the need for SDK integration. Fig. 2 and Fig. 3 illustrate some of these features. Fig. 2 is a screenshot from the MMOG game "Roma Victor" [80] by RedBedlam. Fig. 3 shows the same game scene with a mock-up of CNG features. The modifications, which are numbered in Fig. 3, are as follows:

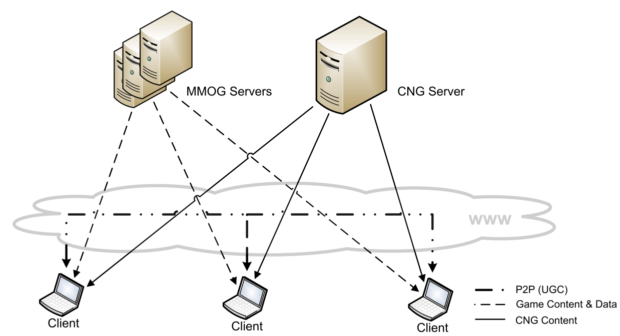


Figure 1. CNG architecture.

- (1) The original resolution of the game was modified to enable an additional frame around the game to hold the in-frame objects. IGIT uses the GPU of the user's machine for changing the resolution of the game to avoid reduction in the image quality;
- (2) Instant messaging window as an example of active Web 2.0 application;
- (3) Web browser that presents online passive information (in this example, a leader board);
- (4) Another Web browser window that presents an updated advertisement;
- (5) MMOG specific chat to enable the users in a specific scene to cooperate;
- (6) In-game 3D UGC. In this example, a user added a note on a tree to publish an eBay auction;
- (7) Two windows of a video chat with casual friends or cooperative players.

The choice of application and the application's screen location are under the control of the user (player). The Web 2.0 applications are browser-based applications that are downloaded from the CNG Server and run in the web-browser instances within the CNG Client. The purpose of Web 2.0 Applications is to offer online collaboration services to the user. They are browser-based and downloaded from the CNG Server and run in the web-browser instances within the CNG Client.

Since they are web-based the CNG client retrieves all the necessary information from the CNG Server

In addition, the CNG toolbox includes video recording and editing tools that allow users to capture the video of the game and

1) trim the captured game video
2) split, duplicate and sequence the recorded video cuts
3) remix trimmed cuts of the recorded video
4) upload the edited video to YouTube

Figure 2. Original MMOG screenshot.

Figure 3. IGIT-modified MMOG screenshot.

B. P2P live video system

In existing MMOGs, a player can capture the video of the game and send it to a central server which broadcasts it live to other users [11]. However, this solution, which heavily relies on central servers, has many drawbacks such as high costs for bandwidth, storage, and maintenance. Moreover, this solution is not easily scalable to increasing number of users. The CNG project proposes a P2P live video streaming system to address the limitations of server-based solutions. The CNG P2P live video system allows every peer to become a source of a user-generated video stream for a potentially large set of receivers. While many P2P live video systems have been proposed, none of them has been specifically designed for the unique environment of MMOGs. In particular, none of the existing P2P live video systems addresses the following challenges:

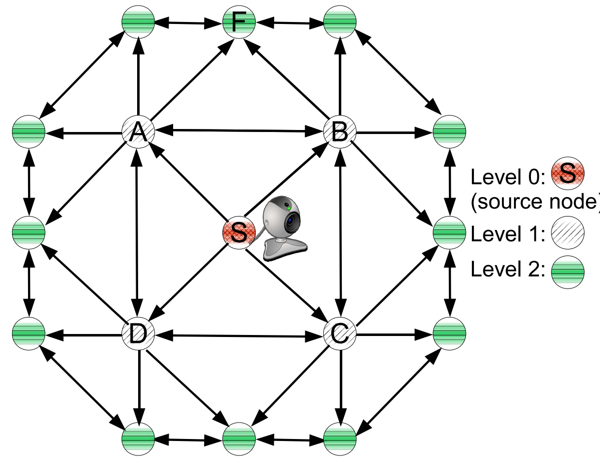


Figure 4. P2P topology.

- Any MMOG player should be able to multicast live video. The video can potentially be received by any other player in the P2P network.
- Live video streaming should not consume the upload and download bandwidth that is necessary for the smooth operation of the MMOG (MMOG client-server traffic).
- Live video should be delivered at about the same time to all peers at the same "level". For example, a level can be a priority class in a multi-tiered premium service. Peers in a higher priority class should be able to watch the video before those in a lower priority class. Alternatively, a level can be defined as the set of MMOG players that are in the same region of the virtual world.

The CNG P2P live video system is designed as follows. Peers are organized in levels with the source peer placed at level 0 (Fig. 4).

The video is captured in real time from the source screen, compressed, and partitioned into source blocks. Each source block corresponds to one GOP (Group of Pictures) and is an independent unit of fixed playback duration (e.g., 1 s). The source peer applies rateless coding on each source block and sends the resulting encoded symbols in successive UDP packets to level-1 peers. Packets are sent according to a scheduling strategy. The strategy specifies the maximum number n of encoded packets that can be sent by the source for this block, the time t_i at which packet i is sent, and a hierarchical forwarding scheme $F_i,\ i = 1, 2, \ldots, n$. An example of a scheduling strategy for $n = 4$ and the four level-1 peers of Fig. 4 is as follows.

- $1 : t_1 : A \to B + D(\to C)$,
- $2 : t_2 : B \to A + C(\to D)$,
- 3: $t_3 : C \to B(\to A) + D$,
- 4: $t_4 : D \to C(\to B) + A$

The strategy says that packet 1 should be transmitted at time t_1 to A. A forwards the packet to B and D. D forwards it to C. Packet 2 should be transmitted at time t_2 to B. B forwards it to A and C. C forwards it to D. Packet 3 should be transmitted at time t_3 to C. C forwards it to B and D. B forwards it to A. Packet 4 should be transmitted at time t_4 to D. D forwards it to C and A. C forwards it to B.

A level-1 peer uses its own scheduling strategy to immediately forward any packet received from the source to level-2 peers. Moreover, as soon as it successfully decodes a source block, it sends an acknowledgment to the source, so that it stops sending it packets. Then it applies rateless coding on the source block and creates new packets. These new packets are sent to level-2 peers according to the scheduling strategy. Since a level-2 peer may receive packets from different level-1 peers, level-1 peers use randomly chosen rateless code seeds to minimize the probability that a level-2 peer receives duplicate packets. The value of the seed is sent as part of the header, so that the receiving peer can generate the same graph as the encoding peer. The procedure described above for two levels is repeated for the next levels.

Note that with the exception of peers situated at the last level, a peer will usually have two phases: a forwarding one (before the decoding is successful) and an encoding one (after decoding the block).

One of the main challenges consists of optimizing the scheduling strategy. The details of this optimization will be presented in a future paper.

Our system extends previous ideas proposed in [58], [59]. However there are many important differences between these works and our scheme. For example, the systems of [58], [59] do not have the notion of scheduling strategy. Also in [58], [59], there is no notion of levels.

As UDP does not have a built-in congestion control mechanism, a pure UDP-based application may overwhelm the network, leading to packet loss and degraded video quality. Therefore, the CNG project proposes an application-layer congestion control mechanism for the P2P system. The source adapts the video bit rate according to feedback received periodically from all peers. This feedback consists of the outage rate, i.e., the percentage of source blocks that were not decoded in time.

IV. Conclusion

We presented the EU funded CNG project. CNG supports and enhances community activities between MMOG gamers by enabling them to create, share, and insert UGC. The UGC considered by the CNG project includes 3D objects, graphics, and video. CNG develops in-game community activities using an in-game graphical insertion technology that replaces or inserts content in real time without the need to change the game's code in the client or server. CNG uses an architecture that efficiently combines the client-server infrastructure for the MMOG activities with a P2P overlay for the delivery of live video. The video traffic represents a real challenge to the network already occupied by the MMOG client-server data. The project proposes new techniques for P2P live video streaming that are "friendly to the MMOG client-server traffic. Since video can be resource heavy, the network indirectly benefits from the increased locality of communication. CNG also provides Web 2.0 tools for audio and video chat, instant messaging, in-game voting, reviewing, and polling. This will reduce the need for visiting forums outside the game and diluting the MMOG experience.

CNG has the potential to provide huge benefits to MMOG developers and operators. New community building tools will be offered cost-effectively and efficiently, without the need to redesign or recode the existing game offerings. The user experience will be enriched, and the needs of the end-users will be better addressed. The community will be brought into the content, and the game communities will become more engaged, reducing churn to other MMOGs. New income streams will be delivered with the help of in-game and around game advertising. Yet, MMOG developers and operators will be able to maintain control over how various commercial and UGC content is displayed, thus keeping editorial control of the look and feel of their MMOG.

Acknowledgment

The research leading to these results has received funding from the European Commission's Seventh Framework Programme (FP7, 2007-2013) under the grant agreement no. ICT-248175 (CNG project).

References

[1] [Online]. Available at: http://www.slideshare.net/ICOPartners/free-to-play-games-in-europe-2009. Last accessed: 7/2/2012.

[2] [Online]. Available at: http://www.cng-project.eu/. Last accessed: 7/2/2012.

[3] S. Ahmad, C. Bouras, R. Hamzaoui, A. Papazois, E. Perelman, A. Shani, G. Simon, and G. Tsichritzis, "The Community Network Game project: Enriching online gamers experience with user generated content," in Proc. 2nd Int. Conf. Creative Content Technologies (CONTENT 2010), Lisbon, Nov. 2010.

[4] [Online]. Available at:: http://eu.battle.net/wow/en/. Last accessed: 7/2/2012.

[5] Facebook API. [Online]. Available at: http://developers.facebook.com/docs/. Last accessed: 7/2/2012.

[6] [Online]. Available at: http://eu.spore.com/home.cfm?lang=en. Last accessed: 7/2/2012.

[7] [Online]. Available at: http://www.techsmith.com/camtasia/. Last accessed: 7/2/2012.

[8] [Online]. Available at: http://www.fraps.com. Last accessed: 7/2/2012.

[9] Youtube API. [Online]. Available at: http://code.google.com/apis/youtube/casestudies/ea.html. Last accessed: 7/2/2012.

[10] D. Takahashi, "YouTube game videos become a big channel for game marketers," Dec. 2008. [Online]. Available at: http://games.venturebeat.com/2008/12/18/youtube-game-videos-become-a-big-channel-for-game-marketers/. Last accessed: 7/2/2012.

[11] [Online]. Available at: http://xfire.com. Last accessed 7/2/2012.

[12] Adobe Flash. [Online]. Available at: http://www.adobe.com/support/flash/downloads.html. Last accessed: 7/2/2012.

[13] XMLHttpRequest, W3C Working Draft, 2009. [Online]. Available at: http://www.w3.org/TR/2009/WD-XMLHttpRequest-20091119/. Last accessed: 7/2/2012.

[14] SOAP Version 1.2 Part 1: Messaging Framework (Second Edition), W3C Recommendation, 2007. [Online]. Available at: http://www.w3.org/TR/soap12-part1. Last accessed: 7/2/2012.

[15] [Online] Available at: http://blueimp.net/ajax. Last accessed: 7/2/2012.

[16] [Online] Available at: http://www.gmail.com. Last accessed: 7/2/2012.

[17] [Online] Available at: https://addons.mozilla.org/en-US/firefox/addon/chatzilla/. Last accessed: 7/2/2012.

[18] [Online] Available at: http://www.mibbit.com. Last accessed: 7/2/2012.

[19] [Online] Available at: http://www.php-development.ru/software/chat-java.php. Last accessed: 7/2/2012.

[20] [Online] Available at: http://www.meebo.com/. Last accessed: 7/2/2012.

[21] [Online] Available at: http://www.google.com/ig. Last accessed: 7/2/2012.

[22] [Online] Available at: http://www.orkut.com/. Last accessed: 7/2/2012.

[23] [Online] Available at: http://www.flashcoms.com/products/community_video_chat/. Last accessed: 7/2/2012.

[24] [Online] Available at: http://avchat.net/. Last accessed: 7/2/2012.

[25] [Online] Available at: http://www.red5chat.com/. Last accessed: 7/2/2012.

[26] [Online] Available at: http://www.mebeam.com/. Last accessed: 7/2/2012.

[27] [Online] Available at: http://www.fileguru.com/Web-Voice-Chat/info. Last accessed: 7/2/2012.

[28] [Online] Available at: http://www.123flashchat.com. Last accessed: 7/2/2012.

[29] [Online] Available at: http://acidjs.wemakesites.net/voting-poll.html. Last accessed: 7/2/2012.

[30] [Online] Available at: http://ajaxian.com/archives/dpolls-an-ajax-pollster. Last accessed: 7/2/2012.

[31] [Online] Available at: http://www.flabell.com/flash/XML-Flash-Voting-Poll-39. Last accessed: 7/2/2012.

[32] [Online] Available at: http://wordpress.org/. Last accessed: 7/2/2012.

[33] [Online] Available at: http://www.movabletype.org/. Last accessed: 7/2/2012.

[34] [Online]. Available at: http://www.pptv.com/. Last accessed: 7/2/2012.

[35] [Online]. Available at: http://www.PPstream.com. Last accessed: 7/2/2012.

[36] Y.-H. C. Sanjay, S. G. Rao, S. Seshan, and H. Zhang, "A case for end system multicast," in Proc. ACM Sigmetrics, pp. 1–12, 2002.

[37] G. Yang, S. Kyoungwon, K. Jim, and T. Don, "P2cast: peer-to-peer patching scheme for VoD service," in Proc. 12th Int. Conf. World Wide Web, pp. 301–309, 2003.

[38] M. Castro, P. Druschel, A.-M. Kermarrec, A. Nandi, A. Rowstron, and A. Singh, "SplitStream: High-bandwidth multicast in a cooperative environment," in Proc. ACM SOSP, 2003.

[39] V. N. Padmanabhan, H. J. Wang, P. A. Chou, and K. Sripanidkulchai, "Distributing streaming media content using cooperative networking," in Proc. NOSSDAV '02, pp. 177–186, 2002.

[40] M. Jelasity, R. Guerraoui, A.-M. Kermarrec, and M. V. Steen, "The peer sampling service: experimental evaluation of unstructured gossip-based implementations," in Proc. ACM/IFIP/USENIX Int. Conf on Middleware, pp. 79–98, 2004.

[41] B. Li, G. Y. Keung, C. Lin, J. Liu, and X. Zhang, "Inside the new coolstreaming: Principles, measurements and performance implications," in Proc. INFOCOM'08, 2008.

[42] A.-M. Kermarrec, E. Le Merrer, Y. Liu, and G. Simon, "Surfing peer-to-peer IPTV system: distributed channel switching," in Proc. EuroPar, 2009.

[43] A.-T. Gai, F. Mathieu, F. de Montgolfier, and J. Reynier, "Stratification in P2P networks: application to BitTorrent," in Proc. 27th IEEE International Conference on Distributed Computing Systems (ICDCS), 2007.

[44] H. Yin, X. Liu, T. Zhan, V. Sekar, F. Qiu, C. Lin, H. Zhang, and B. Li, "Design and deployment of a hybrid CDN-P2P system for live video streaming: Experiences with livesky," in Proc. ACM Multimedia, 2009.

[45] H. Xie, Y. R. Yang, A. Krishnamurthy, Y. G. Liu, and A. Silberschatz, "P4p: Provider portal for applications," ACM SIGCOMM Computer Communication Review, Vol. 38, No. 4, Oct. 2008.

[46] M. Zhang, J.-G. Luo, L. Zhao, and S.-Q. Yang, "A peer-to-peer network for live media streaming using a push-pull approach," in Proc. 13th ACM International Conference on Multimedia, 2005.

[47] M. Röder, *Efficient Rate-Distortion Optimized Media Streaming*, PhD Thesis, University of Konstanz, 2007.

[48] P.A. Chou, "Streaming media on demand," in M. van der Schaar, P.A. Chou (editors), Multimedia over IP and Wireless Networks: Compression, Networking, and Systems, Academic Press, 2007.

[49] V. N. Padmanabhan, H. J. Wang, and P. A. Chou, "Resilient peer-to-peer streaming," in Proc. IEEE ICNP, pp. 16–27, Atlanta, GA, Nov. 2003.

[50] M. Hefeeda, A. Habib, B. Botev, D. Xu, and B. Bhargava, "PROMISE: peer-to-peer media streaming using CollectCast," in Proc. ACM Multimedia (MM'03), Berkeley, CA, Nov. 2003.

[51] X. Xu, Y. Wang, S.§. Panwar, and K. W. Ross, "A peer-to-peer video-on-demand system using multiple description coding and server diversity," in Proc. IEEE Int. Conf. Image Processing, Singapore, Oct. 2004.

[52] K. Suh, C. Diot, J. Kurosey, L. Massoulie, C. Neumann, D. Towsley, and M. Varvello, "Push-to-Peer video-on-demand system: design and evaluation," IEEE Journal on Selected Areas in Communications, vol. 25, no. 9, pp. 1706–1716, Dec. 2007.

[53] E. Setton, J. Noh, and B. Girod, "Rate-distortion optimized video peer-to-peer multicast streaming," in Proc. P2PMMS'05, pp. 39–48, Singapore, Nov. 2005.

[54] E. Akyol, A. M. Tekalp, and M. R. Civanlar, "A flexible multiple description coding framework for adaptive peer-to-peer video streaming," IEEE Journal of Selected Topics in Signal Processing, vol. 1, no. 2, pp. 231–245, Aug. 2007.

[55] M.-T. Lu, J.-C. Wu, K.-J. Peng, P. Huang, J. J. Yao, and H. H. Chen, "Design and evaluation of a P2P IPTV system for heterogeneous networks," IEEE Transactions on Multimedia, vol. 9, pp. 1568–1579, Dec. 2007.

[56] N. Thomos and P. Frossard, "Raptor network video coding," in Proc. 1st ACM International Workshop on Mobile video (in conjunction with ACM Multimedia 2007), Augsburg, Germany, Sep. 2007.

[57] A. Shokrollahi, "Raptor codes," IEEE Trans. Inf. Theory, vol. 52, pp. 2551–2567, June 2006.

[58] C. Wu and B. Li, "rStream: resilient and optimal peer-to-peer streaming with rateless codes," IEEE Transactions on Parallel and Distributed Systems, vol. 19, pp. 77–92, Jan. 2008.

[59] M. Grangetto, R. Gaeta, and M. Sereno, "Rateless codes network coding for simple and efficient P2P video streaming," in Proc. IEEE ICME 2009, Cancun, Mexico, 2009.

[60] M. Grangetto, R. Gaeta, and M. Sereno, "Reducing content distribution time in P2P based multicast using rateless codes," in Proc. Italian Networking Workshop, pp. 1–12, Cortina, 2009.

[61] E. Setton, J. Noh, and B. Girod, "Congestion-distortion optimized peer-to-peer video streaming," in Proc. IEEE ICIP-2006, Atlanta, GA, Oct. 2006.

[62] P. Baccichet, T. Schierl, T. Wiegand, and B. Girod, "Low-delay peer-to-peer streaming using scalable video coding," in Proc. International Packet Video Workshop, PV2007, Lausanne, Switzerland, Nov. 2007.

[63] [Online]. Available at: http://tools.ietf.org/html/rfc4340. Last accessed 7/2/2012.

[64] R. Diana and E. Lochin, "ECN verbose mode: A statistical method for network path congestion estimation," in Proc. INFOCOM, San Diego, CA, 2010.

[65] T. Silverston and O. Fourmaux, "Measuring P2P IPTV systems," in Proc. ACM NOSSDAV'07, June 2007.

[66] S. Ali, A. Mathur, and H. Zhang, "Measurement of commercial peer-to-peer live video streaming," in Proc. ICST Workshop Recent Adv. Peer-To-Peer Streaming, Waterloo, Aug. 2006.

[67] M. Alhaisoni and A. Liotta, "Characterization of signalling and traffic in Joost," Journal of Peer-to-Peer Networking and Applications, Vol. 2, issue 1, pp. 75–83, 2009.

[68] P. Baccichet, J. Noh, E. Setton, and B. Girod, "Content-aware P2P video streaming with low latency," in Proc. IEEE International Conference on Multimedia and Expo, ICME 2007, Beijing, China, Jul. 2007.

[69] M. Zhang, Q. Zhang, L. Sun, and S. Yang, "Understanding the power of pull-based streaming protocol: Can we do better?," IEEE Journal on Selected Areas in Communications, Vol. 25, pp. 1678–1694, Dec. 2007.

[70] M. Zhang, L. Sun, and S. Yang, "iGridMedia: Providing delay-guaranteed peer-to-peer streaming service on Internet," in Proc. IEEE GLOBECOM 2008, New Orleans, LO, Dec. 2008.

[71] F. Pianese, D. Perino, J. Keller, and E.W. Biersack, "PULSE: An adaptive, incentive-based, unstructured P2P Live streaming system," IEEE Transactions on Multimedia, Vol. 9, Number 8, pp. 1645–1660, Dec. 2007.

[72] M. Wang and B. Li, "R2: Random push with random network coding in live peer-to-peer streaming," IEEE Journal on Selected Areas in Communications, Vol. 25, Number 9, pp. 1655–1666, Dec. 2007.

[73] [Online]. Available at: http://www.playxpert.com/. Last accessed 7/2/2012.

[74] [Online]. Available at: http://www.freeridegames.com/. Last accessed 7/2/2012.

[75] [Online]. Available at: http://en.wikipedia.org/wiki/Massive_Incorporated. Last accessed 7/2/2012.

[76] [Online]. Available at: http://www.doublefusion.com/. Last accessed 7/2/2012.

[77] [Online]. Available at: http://store.steampowered.com/. Last accessed 7/2/2012.

[78] [Online]. Available at: http://www.overwolf.com/. Last accessed 7/2/2012.

[79] [Online]. Available at: http://raptr.com/. Last accessed 7/2/2012.

[80] [Online]. Available at: http://www.roma-victor.com. Last accessed 7/2/2012.

A Cognitive Handoff

Holistic Vision, Reference Framework, Model-driven Methodology and Taxonomy of Scenarios

Francisco A. González-Horta, Rogerio A. Enríquez-Caldera, Juan M. Ramírez-Cortés, Jorge Martínez-Carballido
Department of Electronics
INAOE
Tonantzintla, Puebla, México
e-mail: fglez@inaoep.mx, rogerio@inaoep.mx, jmram@inaoep.mx, jmc@inaoep.mx

Eldamira Buenfil-Alpuche
Faculty of Engineering
Polytechnic University of Guerrero State, UPEG
Taxco, Guerrero, Mexico
e-mail: eldamira@gmail.com

Abstract— **Current handoffs are not designed to achieve multiple desirable features simultaneously. This weakness has resulted in handoff schemes that are seamless but not adaptive, or adaptive but not secure, or secure but not autonomous, or autonomous but not correct, etc. To face this limitation, in this paper we envision and develop a new kind of handoff system, which is context-aware in the sense that uses information from its external and internal environment: a cognitive handoff. Thus, the resulting cognitive handoff can attain multiple purposes simultaneously through trading-off multi-criteria and based on a variety of policies. We also discuss the difficulties of developing cognitive handoffs and propose a new model-driven methodology for their systematic development. The theoretical framework of this methodology is the holistic approach, the functional decomposition method, the model-based design paradigm, and the theory of design as scientific problem-solving. We applied the proposed methodology and obtained the following results: (i) a correspondence between handoff purposes and quantitative environment information, (ii) a novel taxonomy of handoff mobility scenarios, and (iii) an original state-based model representing the functional behavior of the handoff process.**

Keywords- Cognitive handoff; handoff methodology; handoff scenarios

I. INTRODUCTION

A handoff is intended to preserve the user communications while different kinds of transitions occur in the network connection. Thus, a handoff is the process of transferring communications among radio channels, base stations, IP networks, service providers, mobile terminals, or any feasible combination of these elements [1]. Significant desirable handoff features mentioned in the review of the literature [2] are: seamless [3], adaptive [4], autonomous [5], secure [6], and correct [7]; however, many others can be found in the vast literature of handoffs: transparent, reliable, flexible, robust, balanced, immune, fast, soft, smooth, lossless, efficient, proactive, predictive, reactive, QoS-based, power-based, location-aided, time-adaptive, intelligent, generic, etc. Despite the rich variety of desirable handoff features the resulting handoffs they are not enough to face the challenges of the future Internet [8], [9], [10], [11] and two important problems remain unsolved: i) how can be combined different desirable features into a single handoff process so that it can achieve many purposes simultaneously? And ii) how to define every desirable feature so that ambiguity and subjectivity can be reduced?

This gap in knowledge about handoffs has produced a number of single-purpose schemes that successfully achieve one attractive feature but completely ignore others; e.g., seamless handoffs with poorly or null adaptation to handoff scenarios or technologies [12]; adaptive handoffs that do not consider any security goal [4]; secure handoffs that ignore user autonomy [6]; etc. Also, there is a growing confusion in literature about similar features; e.g., accurate-correct, fast-timely, smooth-seamless, robust-reliable, etc. In order to reduce misuse and ambiguity of these attributes is convenient to associate a qualitative property (purpose) and quantitative measures (objectives and goals) to each desirable feature. By doing so, we can qualify and quantify their performance individually or in comparison with others.

The development of handoffs achieving multiple desirable features has been "delayed" by the research community itself, despite it was advised since 1997 by Tripathi [8], because many authors preferred to focus on understanding and controlling very specific handoff scenarios (reductionist approach) instead of managing complex and generic handoff scenarios (holistic approach). However, recent handoff schemes, like the ones proposed by Altaf in 2008 [9] for secure-seamless-soft handovers, or Cardenas in 2008 [10] for fast-seamless handoffs, or Singhrova in 2009 [11] for seamless-adaptive handoffs, show a tendency towards cognitive handoffs.

Major contributions of this paper include:

1) A new holistic vision of handoffs.
Many handoff solutions follow a reductionist approach; i.e., they achieve one desirable feature, use a small amount of handoff criteria, and work only in very specific scenarios. Although these simplistic solutions provide understanding and control of particular situations, we have seen how they

quickly become special cases of more general models. Thus, we claim that the handoff problem for the future Internet requires holistic solutions, achieving multiple desirable features, using a diversity of context information, and adapting to any handoff scenario.

2) A reference framework for cognitive handoffs.
We propose a new kind of handoff that is multipurpose, multi-criteria, context-aware, self-aware, policy-based, and trades-off multiple conflicting objectives to reach its intended goals. This paper provides the conceptual model and its first level of functional decomposition.

3) A model-driven methodology to develop cognitive handoffs.
This methodology allows to systematically develop cognitive handoffs using a comprehensive model-based framework. The proposed methodology is founded on a synthesis of holism, reductionism, functional decomposition, model-based design, and scientific problem-solving theory. As a result of deploying our methodology, we present a clear correspondence among cognitive handoff purposes and handoff environment information.

Besides, in order to test the resulting cognitive handoff when applying such methodology with the parameters associated to, and for a given scenario, we develop the following contributions:

4) A taxonomy of handoff mobility scenario.
This taxonomy gives a classification of handoff scenarios by considering all feasible combinations of several communication dimensions involved.

5) An original state-based model of the handoff process.
This state based model is represented by five-state diagram, which describes the control handoff process and the way all different stages are being coordinated before, during, and after the handoff.

The paper is organized in sections that correspond to the previously described contributions and therefore it starts in Section II with a distinction between Single-purpose and multi-purpose handoffs. Section III presents the holistic vision for the conceptualization of cognitive handoffs. Section IV presents our cognitive handoff reference framework and its specific characteristics. Section V describes the ad hoc model-driven methodology that we are using for developing cognitive handoffs. This section discusses the difficulties for developing cognitive handoffs and provides an overview of theoretical framework setting the basis of our methodology. Section VI shows the first results we obtained from applying the methodology. These results include: (a) the correlation between context data and desirable handoff features through the definition of handoff purposes, objectives, and goals; (b) the taxonomy of handoff scenarios derived from combining all the possible transition elements involved in handoffs; and, (c) a cognitive handoff state-based model that describes a general behavior of the control handoff process. Section VII presents a basic discussion on the applicability of preliminary results. Finally, Section VIII concludes the paper with a summary of contributions and future work.

II. Single-Purpose Vs Multi-Purpose Handoffs

Dr. Nishith D. Tripathi in his outstanding thesis work published in 1997 [12] probably was the first author in considering a handoff that can simultaneously achieve many desirable features. His inspiring work served for many years as a basis for developing high performance handoffs; however, the complexities of handoff scenarios from 1997 to present days have changed significantly. For instance, the handoff concept changed from simple lower-layer transitions between base stations and channels to more elaborated cross-layer transitions among networks, providers, and terminals. The limited scope of Tripathi's handoff concept has brought in consequence that his algorithms and models become today special cases of more general models. Holism is relevant in this way to provide a long-term solution for the handoff problem. Another author who describes several desirable handoff features is Nasser et al. [13] in 2006. Both, Tripathy and Nasser, described various desirable features, but they did not make any difference among features, purposes, objectives, and goals. A handoff model needs a clear distinction to such former concepts.

The holistic vision to the handoff problem has also been studied by Dr. Mika Ylianttila in his exceptional thesis work [14] published in 2005. He presented a holistic system architecture based on issues involved in mobility management areas (e.g., mobility scenarios, handoff strategies, handoff control, handoff algorithms, handoff procedures, mobility protocols, mobility parameters, performance measures, and handoff metrics). The work of Ylianttila improved the architecture of handoff issues that Pahlavan [15] published in 2000. However, these architectures have some drawbacks: i) they did not include the context management problem in their models; ii) they did not mention the tradeoffs that handoffs should consider in a multi-objective scenario; and iii) their architectures are based on types of issues and not in the functionality aspects of the handoff process.

Besides the above related work, we use two criteria to classify handoff schemes that are approaching to cognitive handoffs: the number of desirable features they achieve and the amount of context information they use. Handoff schemes, like the ones proposed by So [16] and Zhang [17], achieve only one desirable feature using limited context information; they provide seamless handoffs between particular network technologies and specific mobility scenarios. The schemes proposed by Siddiqui [18] and Hasswa [19] use broad context information, but they are focused only in one feature (seamlessness).

Conversely, the solutions proposed by Sethom [6] and Tuladhar [20] provide seamless and secure handoffs on a variety of handoff scenarios because they use broad context. The schemes proposed by Singhrova [4] and Chen [21] achieve seamless and adaptive mobility, but they cannot adapt to any handoff scenario because they use limited context. Finally, the scheme proposed by Altaf [22] achieves seamless, secure, soft, and adaptive handoffs, but just between WiMAX and 3G networks because they use limited context.

The information the handoff process uses for making decisions, has been increasing as we want to deploy more intelligent handoff systems. Handoffs for the first generation wireless networks were called *single-criterion,* because they were mainly based on signal-strength or few criteria taken from the access network dimension. At the second generation, the need for improving the effectiveness of handoffs led to include more information to this process. Handoffs for 2G networks were known as *multi-criteria*. They included many criteria from distinct dimensions, but not from all dimensions; e.g., they might consider the battery load from the terminal dimension and the user speed from the mobility dimension, but ignore the fees from the service provider.

At the beginning of 3G networks, several handoff schemes were deployed using information or criteria from the entire external handoff environment, and they were called *context-aware* handoffs. In 2003, Prehofer et al. [23], defined a context-management architecture for addressing the problem of collecting, compiling, and distributing handoff context information. This remarkable work started a new stage in the development of handoffs. Part of this architecture was used by Pawar et al. [24] in 2008 for developing context-aware handoffs applied to mobile patient monitoring.

At the dawn of 4G networks, a new type of handoff solutions, called *self-aware*, started to use a variety of handoff performance parameters from its internal environment to self-adapt its behavior according to different performance goals. For the future networks, *environment-aware* handoffs, using information from both, the external and internal environment, will be deployed.

Despite these recent advances in context-management architectures and applications, the lack of a clear relationship between handoff context information and handoff desirable features is adding unnecessary complexity to the process of handoff. The handoff decision making process should be oriented to accomplish more than just one desirable feature. Therefore, we consider that in difference with current handoff schemes, a *cognitive* handoff is aware of its external and internal environment and optimally achieves multiple desirable features simultaneously.

Considering the tendency on handoff research, it will be common to observe in the near future a new generation of handoffs that can achieve many desirable features using broad handoff context information. In current literature, none architecture, model, or algorithm is reported to have this property.

Regarding the related work of standardization bodies, like the IEEE 802.21 and the IETF MIPSHOP, we observed that they are focusing in seamless heterogeneous handoffs; they are not taking into account the vast diversity of desirable features that handoffs could have. The IEEE 802.21 workgroup has approved three task groups to face very particular handoff scenarios: the IEEE 802.21a for security extensions to media independent handovers, the IEEE 802.21b for handovers with downlink only technologies, and the IEEE 802.21c for optimized single radio handovers. We believe they are following a reductionist approach, but they lack the holistic vision of cognitive handoffs. Emmelman [25] discusses ongoing activities and scopes of these standardization bodies.

III. THE COGNITIVE HANDOFF HOLISTIC VISION

First of all, we will describe in a holistic manner the vision of cognitive handoffs.

A. *Origin of Single-Purpose Handoffs*

The thoughtful study of handoffs started in the early 1990s with the first generation (1G) cellular networks (e.g., AMPS [26]). These networks provided seamless conversations while the mobile phone switched between channels and base stations. The decision to perform a handoff was made only on a signal strength basis, but the handoff execution should be imperceptible to users. For this reason, the AMPS system required that the handoff gap be no more than 100 ms to avoid the possibility of dropping a syllable of speech [26]. These traditional handoffs are single-purpose/single-criterion or seamless/signal strength.

B. *Major Challenges in the Future Internet*

1) Multidimensional Heterogeneity: A major trend in future communication systems is the coexistence of multiple dimensions of heterogeneity integrated into a seamless, universal, uniform, ubiquitous, and general-purpose network. This future Internet will be seamless if it hides heterogeneity to users, universal if it can be used by anyone with any terminal, uniform if it is an all-IP network, ubiquitous if it is available anywhere and anytime, and general-purpose if it can provide any service. We divide heterogeneity into five dimensions as illustrated in Fig. 1 and explained in the next paragraphs. The arrows going down from the service provider dimension to the user mobility dimension depict two different handoff scenarios created by instantiating objects in each dimension.

a) Diversity on service providers and operators: Offer different classes of services, billing models, security policies, and connection prices. They deploy different wireless technologies around the world and make roaming agreements and alliances with other providers and operators.

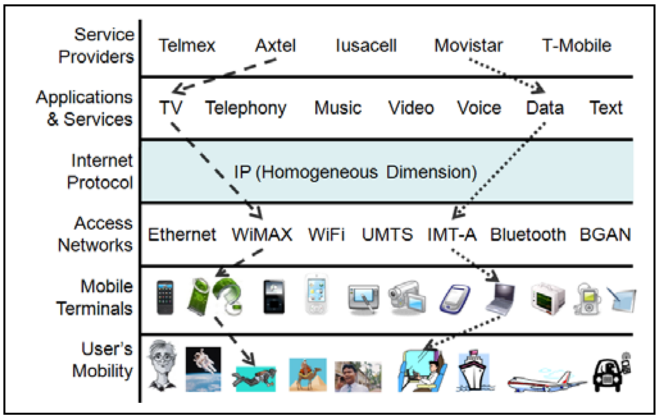


Figure 1. Multidimensional heterogeneity in the future Internet.

b) Diversity on service providers and operators: Offer different classes of services, billing models, security policies, and connection prices. They deploy different wireless technologies around the world and make roaming agreements and alliances with other providers and operators.

c) Variety of applications and services: Intend to fulfill the distinct ways of human communication; e.g., voice, video, data, images, text, music, TV, telephony, etc.

d) Several access network technologies: Include wired and wireless access technologies [21]; e.g., Ethernet, Bluetooth, WiMAX, WiFi, UMTS, MBWA, IMT-2000, GPRS, GSM, EDGE, LTE/SAE, DVB-HS, etc. They differ in terms of electrical properties, signaling, coding, frequencies, coverage, bandwidth, QoS guarantees, mobility management, media access methods, packet formats, etc.

e) Plethora of mobile user terminals: Users can be humans, machines, or sensors. Terminals for machines are integrated parts of machines. Sensor terminals collect information from networked sensors [27]. Terminals for humans are mobile and multimode, equipped with telecommunication capabilities and different saving energy characteristics; they change its factor form from those looked like computers (laptops, netbooks) to those looked like cell phones (PDAs, smartphones).

f) Numerous user mobility states: Network terminals can be located anywhere – in space, on the ground, under the ground, above water, underwater, and they can be fixed in a geographic position or moving at any speed – pedestrian, vehicular, ultrasonic [27].

Nowadays, no handoff solution exists, which comprehensively addresses the entire scale of heterogeneity. Moreover, multidimensional heterogeneity has three main attributes: is inevitable, is the source of great amounts of context information, and produces an infinite number of handoff scenarios.

2) Ubiquitous Connectivity: It enables connectivity for anyone or anything, at any time, from anywhere. A myriad of wireless access technologies are spread across the entire world overlapping one another but avoiding interferences among them. Two requirements for ubiquitous connectivity are:

a) to develop scalable architectures to integrate any number of wireless systems from different service providers [28] and

b) to develop smart multimode mobile terminals able to access any wireless technology [29].

3) Cognitive Mobility: It allows roaming mechanisms where the user is always connected to the best available network, with the smaller number of handoffs, service disruptions, user interventions, security threats, and the greater number of handoff scenarios.

C. External and Internal Handoff Environment

We envision a cognitive handoff as a process that is both context-aware and self-aware. This implicates to make the handoff process aware of its external and internal environment. We borrowed the term ‘cognitive’ from Dr. Dixit vision of cognitive networking [30]. He defines *cognitive networking* as an intelligent communication system that is aware of its environment, both external and internal, and acts adaptively and autonomously to attain its intended goals. We believe cognitive handoffs not only should behave adaptively or autonomously to attain its intended goals, but also seamlessly, securely, and correctly.

On one hand, the external environment is directly related with all the external entities that provide a source of context information to the handoff process. These entities are users, terminals, applications, networks, and providers; a cognitive handoff should adapt to any kind of these entities. These entities maintain a strong cyclic relationship as follows: users interact with terminals, terminals run applications, applications exchange data through networks, networks are managed by providers, and providers subscribe users. The cyclic relationship of external entities suggests that all external context information emanates just from these five basic entities and no more; hence, if we ignore information of any of these entities, the handoff process will not adapt properly to all the scenarios. Therefore, a cognitive handoff should consider all the five entities.

On the other hand, the internal environment is another source of context and it is directly related with the behavior or performance of handoffs. This behavior directly depends on the desirable features of handoff. Next, we identified and describe five major desirable features, which are considered highly significant for the current and future scenarios.

D. Multiple Desirable Features of Handoff

1) Seamlessness: It means to preserve the user communications before, during, and after the handoff thus reducing service degradation or interruption. Service degradation may be due to a continuous reduction in link quality, network quality, handoff quality, QoS guarantees, and energy savings. Service interruption may be due to excessive degradations or a "break before make" approach.

2) Autonomy: This desirable feature is closely related to seamlessness. A handoff is autonomous, automatic, or autonomic when no user interventions are required during a handoff in progress. However, this does not mean that user interventions are not required in handoffs. It is good that users participate in the handoff configuration process by defining their preferences, priorities, or necessities; but, it is convenient that users can perform this activity offline to prevent any distraction during online communications.

3) Security: We say a handoff is secure if not new threats appear along the handoff process and security signaling traffic does not overload the network and degrades the communication services. This is a very challenging task, but if optimization techniques are used together with our model it could be shown that by minimizing handoff latency, authentication latency, and signaling overload, the risk of new threats appearance may be reduced.

4) Correctness: A handoff is correct if it keeps the user always connected to the best available network with the smaller number of handoffs; this is similar to the Gustaffson's vision of ABC defined in [31]. We consider that the best network is the one that is sufficiently better and consistently better. Furthermore, correctness can bring other additional features to the handoff process:

- *Beneficial*: if quality of communications, user expectations, or terminal power conditions get improved after handoff.
- *Timely*: if handoff is executed just in time; i.e., right after target is properly selected and before degradations or interruptions occur.
- *Selective*: if it properly chooses the best network among all the available networks.
- *Necessary*: if it is initiated because of one imperative or opportunist reason.
- *Efficient*: if it selects the most appropriate method, protocol, or handoff strategy, according to the types of: handoff in progress, user mobility, and application.

These handoff attributes derived from correctness, take special relevance during the decision-making phase, where it must be decided why, where, how, who, and when to trigger a handoff.

5) Adaptability: An adaptable handoff should be successful across any handoff scenario. A handoff is successful if it achieves a balance of every desirable feature at a minimum level of user satisfaction.

E. Structure of Handoff Context Information

The handoff context information is extensive, heterogeneous, distributed, and dynamic. It supports the whole operation of the handoff process and the achievement of multiple desirable features. Therefore, such context information should be arranged in a clear structure. Table I and Table II show the structure of handoff context information according to a pair of criteria: the source of context and the class of information respectively. The sources of context originated in the external handoff environment support context-awareness while the one originated in the internal environment (the handoff process itself) will provide self-awareness.

TABLE I. STRUCTURE FOR SOURCE OF CONTEXT INFORMATION

User context: This context allows users to customize the handoff according to their own needs, habits, and preferences. It includes: user preferences, user priorities, user profiles, user history, etc.
Terminal context: Allows the deployment of QoS-aware handoffs, power-based handoffs, and location-aided handoffs: (a) *Link quality:* Received signal strength (RSS), signal to noise ratio (SNR), signal to interference ratio (SIR), signal to noise and interference ratio (SNIR), bit error rate (BER), block error rate (BLER), co-channel interference (CCI), carrier to interference rario (CIR), etc. (b) *Power management:* Battery type (BT), battery load (BL), energy-consumption rate (ECR), transmit power in current (TPC), transmit power in target (TPT), power budget (PB), etc. (c) *Geographic mobility:* Velocity (Vel), distance to a base station (Dist), location (Loc), direction (MDir), coverage area (GCA), etc.
Application context: This context includes the QoS requirements of active applications: Lost packets (LP), delayed packets (DP), corrupted packets (CP), duplicated packets (DuP), data transfer rate (DTR-goodput), packet jitter (PJ), out-of-order delivery (OOD), application type (AppT), etc. The consideration of these QoS parameters makes provisions for application-aware handoffs.
Network context: This context is needed to avoid selecting congested networks (befor handoff), to monitor service continuity (during handoff), and to assess the handoff success by measuring network conditions (after handoff): Network bandwidth (NBW), network load (NL), network delay (ND), network jitter (NJ), network throughput (NT), network maximum transmission unit (NMTU), etc.
Provider context: Connection fees, billing models, roaming agreements, coverage area maps, security management (AAA), types of services (data, voice, video), provider preferences, and provider priorities. A negotiation model may be required to equate the differences between service providers, network operators, and mobile users.
Handoff performance context: Call blocking (CB), call dropping (CD), handoff blocking (HOB), handoff rate (HOR), handoff latency (HOL), decision latency (DLat), execution latency (ExLat), evaluation latency (EvLat), handoff type (HOType), elapsed time since last handoff (ETSLH), interruptions rate (IR), interruption latency (IL), degradations rate (DR), degradation latency (DL), degradation intensity (DI), utility function (UF), signaling overload (SO), security signaling overload (SSO), improvement rate (ImpR), application improvement rate (AppImpR), user improvement rate (UsrImpR), terminal improvement rate (TermImpR), successful handoff rate (SHOR), imperative handoff rate (IHOR), opportunist handoff rate (OHOR), dwell time in the best (DTIB), authentication latency (AL), detected attacks rate (DAR), online user interventions rate (OUIR), tardy handoff rate (THOR), premature handoff rate (PHOR), etc.his context allows users to customize the handoff according to their own needs, habits, and preferences. It includes: user preferences, user priorities, user profiles, user history, etc.

TABLE II. STRUCTURE FOR CLASS OF INFORMATION

Handoff criteria: Network discovery, decision-making, and performance evaluation. Some examples of handoff criteria include variables or parameters from the external/internal environment such as RSS, NL, BL, LP, HOL, Vel, connection price, etc.
Handoff metrics: Mathematical models used to measure several significant tasks of the handoff process; for instance, the quality of links, the quality of communications, the quality of different networks, the quality and quantity of handoffs, the quality of different providers, the achievement of user preferences, the power budget of a mobile terminal, the geographic mobility of a user, etc. Handoff metrics may combine a variety of handoff criteria and help any specific handoff algorithm to make optimal decisions.
Performance measures: Set of handoff metrics that are used to quantify performance of communications, performance of networks, performance of handoffs, and to evaluate the degree of achieving a handoff objective.
Handoff policies: Users and providers define a series of policies to the handoff operation. Policies define and specify rules for making handoff decisions in any particular situation; for instance, what to do if the link quality drops below a level required for an acceptable service. User and provider may have different views of the handoff process; provider may be interested in QoS while user in connection charges. Both points of view must be consistently integrated into a single handoff policy management database.
Handoff constraints: Conditions that must be satisfied in a particular handoff scenario and used to control the handoff operation by keeping performance parameters within specific limits. For instance, for a seamless handoff process, the delay has to be kept within certain boundaries; for real-time applications a delay of 50 ms could be acceptable, whereas non-real-time applications might accept delays as long as 3-10 sec [11].
Handoff configuration: Defines preferences, priorities, and other configuration parameters required to customize the handoff operation. Typically, the configuration information is organized in a handoff profile linked to a particular user, provider, and terminal and should be initially performed offline either by the user, the provider, both or an auto-configuration setup. But, depending on the type of handoff algorithm, different configuration parameters may be required to be initialized, e.g., thresholds, timers, hysteresis, weights, etc.

F. *Cognitive Handoff Reference Framework*

Once we have established and justified the necessity for developing a new handoff system, we present our reference framework based on the statement that “a cognitive handoff should intend to achieve multiple desirable features and be aware of its entire environment by using information coming from multiple context domains”. Fig. 2 depicts this basic idea by interconnecting multiple desirable features with multiple context domains that we already explained separately in III.D and III.E.

The purpose of this model is to help people debate and discuss about the complexity of cognitive handoffs. Thus, topics of discussion would be related to level of complexity, correlation among desired features and context data, and the possibility of establishing handoffs as a multi-objective optimization problem as well as to give specifications for practical implementations. Used in this way this model is not intended for predicting, designing, or implementing cognitive handoffs, but for understanding and explaining such difficult and complex process.

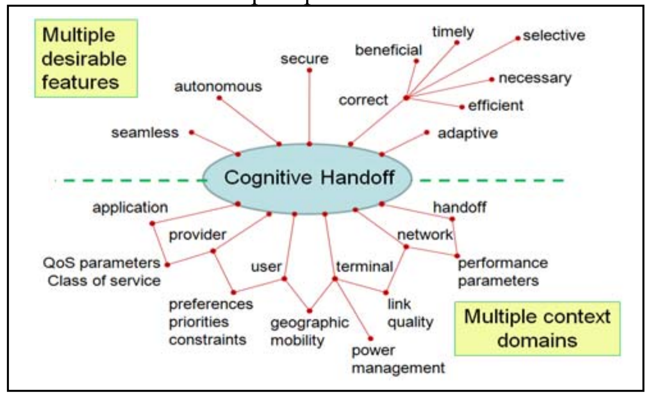


Figure 2. Cognitive handoff conceptual model. The desired features to achieve determine the context data to use and vice versa.

All the above issues have not been addressed in the handoff literature; therefore, in effect, the purpose of this conceptual model is being achieved. Models like the one we present here are validated by credibility, and credibility comes from the way in, which the cognitive maps are built and the clarity it represents most of the opinion’s experts [32]. In the next section we provide some advances towards the development of cognitive handoffs.

IV. COGNITIVE HANDOFF AT WORK

A. *Cognitive Handoff and Complex Systems*

Cognitive handoffs are complex adaptive systems because: (1) they exhibit a complicated hierarchical structure (e.g., a power saving system is part of a network discovery system, which is part of a handoff system, which is part of a mobility system, which is part of a wireless communication system, and so on, but also a power saving system is part of the decision system, which is part of the handoff system, and so on); (2) the whole cognitive handoff system achieves purposes that are not purposes of the parts (e.g., a cognitive handoff purpose is to maintain the continuity of services, but this purpose is not defined in any of the parts or subsystems of the cognitive handoff system); and, (3) the handoff environment is dynamic and therefore adaptability is a desired handoff feature.

B. *Correlating Desired Features and Context Data*

With respect on whether all previously described context data are necessary to describe limitations on the model; one has to realize that the usage of certain context parameters depends on the desirable features being implemented and the context data available in a moment will allow to accomplish or not a particular desired feature. Thus, we need to state a correct relationship or dependence between each desirable handoff feature and the subset of context data necessary to be accomplished. We made a correlation between desired features and context data by transforming desired features into purposes, purposes into objectives,

objectives into goals, and goals into context data. This correlation will be shown in Section VI.

C. *Advances for a Practical Implementation*

The cognitive handoff system, represented in Fig. 2 by the oval in the middle, can be expanded into several sub-systems by using a functional decomposition approach [33].

Fig. 3 shows the main functional sub-systems for cognitive handoffs represented in ovals: handoff control algorithm, network discovery, handoff decisions, handoff execution, handoff evaluation, and handoff context information management. We briefly describe them:

- ***Handoff Control Algorithm***: This is the main director of the handoff procedure. The entity, which implements the control algorithm is called Handoff Control Entity (HCE). There should be one HCE in every user terminal and also there may be many others distributed across the network infrastructure. HCEs are agents that cooperate and compete to take a particular handoff to succeed.
- ***Network Discovery***: This is the system for detecting and discovering available access networks. An available network is a reachable and authorized network considered for an eventual handoff.
- ***Handoff Decisions***: The handoff decisions system is intended to answer the questions of why, when, where, how, and who should trigger the handoff. Typically, this system has focused only in where and when to handoff [34]. The holistic vision extends the scope of handoff decisions.
- ***Handoff Execution***: This system is intended to change the physical and logical connection from one network to another, from one provider to another, or from one terminal to another. This change requires the most effective method, protocol, or strategy according to the current handoff scenario. The MIPSHOP group at IETF and the IEEE 802.21 standard are creating tools for implementing media-independent handoffs since 2003.

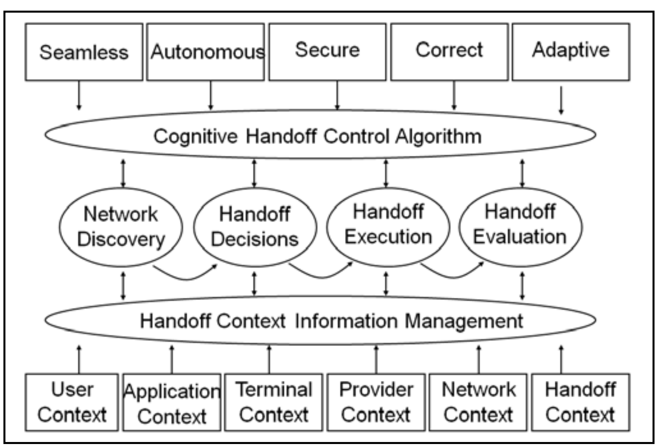


Figure 3. Functional decomposition model. The desired features provide purposes, objectives, and goals to achieve, while context domains provide the information needed to attain such goals.

- ***Handoff Evaluation***: This system measures the achievement of every desirable handoff feature and decides whether the executed handoff was successful or not. The evaluation results should be delivered after the handoff execution but within strict time constraints, thus this task is proactively distributed along the handoff process.
- ***Handoff Context Information Management***: This system is intended to collect the distributed handoff context data, transform the data in information, and redistribute this information to the HCEs, which are responsible for making handoff decisions and control.

Discovery, decisions, execution, and evaluation systems can be viewed as sequential stages of the handoff process; however, the context manager is a background process, which permanently supplies the handoff control entities with fresh information about the handoff environment.

D. *Cognitive Handoff Performance Measures*

The performance evaluation of cognitive handoffs requires a performance metric for each handoff purpose and a graphical representation to visualize multivariate data [35]. These metrics combine mathematically several performance measures that are associated to every handoff purpose. It is possible that metrics can normalize heterogeneous data into a single value representing the performance of each handoff purpose. Moreover, metrics can also be designed as utility functions so that greater values are better and all values are on the same scale.

Fig. 4 exemplifies a radar graph comparing the performance of multiple handoff purposes simultaneously. We say that if all measures are within a boundary circle of acceptable quality, then the cognitive handoff is successful, otherwise the handoff is defective and outliers should be corrected.

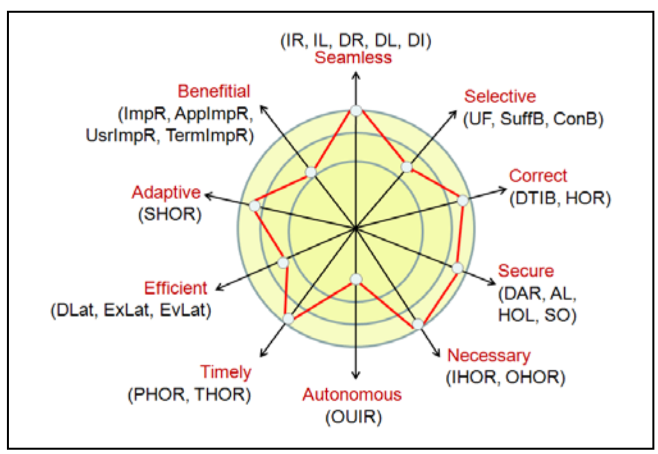


Figure 4. Radar graph comparing the objective functions of multiple handoff purposes simultaneously.

E. Formulating the Cognitive Handoff as a MOP

Let F be the set of desirable handoff features and C be the set of context data. We say that a context variable $v_i \in C$ is *correlated* with a desired feature $f \in F$ if and only if a change on the value of v_i impacts on the purpose of f. For instance, some changes on the value of SNR may degrade or improve the link quality and impact on the purpose of seamlessness that is to maintain the continuity of services; thus, we say that SNR is correlated with seamlessness.

Let V_f be the set of correlated variables with f, where $v_i \in V_f \subseteq C$. We say that v_i is *positively correlated* with f if and only if increments on the value of v_i produce improvements on the purpose of f and, decrements on v_i produce degradations on the purpose of f. For instance, increments on SNR improve the link quality, which improves the service continuity of seamlessness, and conversely, decrements on SNR degrade the link quality, which degrades the service continuity of seamlessness. Therefore, SNR is positively correlated with seamlessness.

↑SNR → ↑LINKQUALITY → ↑SEAMLESSNESS
↓SNR → ↓LINKQUALITY → ↓SEAMLESSNESS

We say that v_i is *negatively correlated* with f if and only if increments on the value of v_i produce degradations on the purpose of f and, decrements on v_i produce improvements on the purpose of f. For example, increments on BER degrade the link quality, which degrades the service continuity of seamlessness, and conversely, decrements on BER improve the link quality, which improves the service continuity of seamlessness. Therefore, BER is negatively correlated with seamlessness.

↑BER → ↓LINKQUALITY → ↓SEAMLESSNESS
↓BER → ↑LINKQUALITY → ↑SEAMLESSNESS

The set V_f is partitioned in two subsets V_f^+ and V_f^- where V_f^+ is the set of variables positively correlated with f and, V_f^- is the set of variables negatively correlated with f.

Furthermore, every v_i may have associated a weight w_i depending of its priority where $w_i \in \Re[0,1]$ and $\sum w_i = 1$. Let $\mathbf{V}$ represent the vector of variables $\mathbf{V} = (v_1, v_2, ..., v_m)$, then the *objective function* for the desired handoff feature f is defined by

$$D(\mathbf{V}) = \sum (k + w_i) \log(v_i^+) - \sum (k + w_i) \log(v_i^-) \quad (1)$$

where k is a scaling factor so that small changes on the context variables reflect big changes on $D(\mathbf{V})$, v_i^+ and v_i^- are positively and negatively correlated variables of f.

In general, the objective function is such that $D(\mathbf{V}) : \Re^m \to \Re$ and is a utility function that we want to maximize because, when desirable features get higher, they represent that they get at the best.

Thus, considering K different objective functions $D_i(\mathbf{V})$ that we want to maximize simultaneously where some of them may be in conflict, the multi-objective optimization problem (MOP) can be stated as the problem of

$$\text{Maximize } \{D_1(\mathbf{V}), D_2(\mathbf{V}), ..., D_k(\mathbf{V})\} \quad (2)$$
$$\text{constrain to } \mathbf{V}_L \le \mathbf{V} \le \mathbf{V}_U,$$

where $\mathbf{V}_L$ and $\mathbf{V}_U$ represent the vectors of lower and upper values of the tolerance range for each variable.

F. Tradeoffs between Conflicting Objectives

A cognitive handoff is designed to achieve multiple purposes, objectives, and goals simultaneously. In the space of handoff objectives, we can distinguish between those with complementary nature and those with competitive nature. Complementary objectives can be simultaneously optimized without any conflict between them, but competing objectives cannot be simultaneously optimized, unless we find compromised solutions, largely known as the tradeoff surface, Pareto-optimal solutions, or non-dominated solutions [36]. We describe several tradeoffs to consider in a multi-objective handoff scheme:

a) (Max. DTIB and Min. HOR): There is a tradeoff between maximizing the time to stay always best connected (DTIB) and minimizing the number of handoffs (HOR). The conflict arises because in a dynamic environment the best network is changing frequently and stochastically; thus, to maximize DTIB is necessary to make frequent handoffs as soon as a new best is available. This increase in the number of handoffs creates a conflict with minimizing HOR.

b) (Min. DLat and Max. SHOR): This tradeoff is between minimizing the handoff decisions latency (DLat) and maximizing the number of successful handoffs (SHOR). The conflict emerges because the less time elapsed to make decisions will necessary lead to reduce the number of successful handoffs. For example, in case of imperative handoffs, DLat is reduced but this may lead to select an incorrect target because the selection time is also reduced.

c) (Max. Sizeof-ContextInfo and Min SO): This is a tradeoff between minimizing the handoff signaling overload (SO) and maximizing the amount of handoff context information to be managed by the handoff control entities. The conflict arises because broad handoff information is required to attain multiple desirable features, but this will increase the amount of signaling traffic in the network.

d) (User and Provider Preferences): Several conflicts may appear due to differences between provider and user preferences. For instance, providers may prefer networks within its own administrative domain while users may prefer networks with lower charges even if they are owned by other service providers; users may prefer a Mobile Controlled Handoff (MCHO) while providers may prefer Network Controlled Handoffs (NCHO). Conflicts like these require a balance between different interests. Handoff protocols like Mobile Assisted Handoff (MAHO) and Network Assisted Handoff (NAHO) try to balance the handoff control [9].

V. Model-Driven Methodology for Developing Cognitive Handoffs

Next, we are going to describe the methodology to develop cognitive handoffs.

A. Difficulties for Developing Cognitive Handoff

The simple idea of achieving multiple purposes simultaneously is challenging even for humans. Moreover, if the intended purposes represent opposing situations, which all of them are desired, then even humans need a way to balance the different purposes in conflict; e.g., the conflict between doing the job accurately and doing it quickly. In optimization theory, multi-objective optimization states that improvements to a single purpose can be made as long as the change that made that purpose better off does not make any other purpose worse off. This is called a Pareto improvement. When no further Pareto improvements can be made, then the solution is called Pareto optimal [36].

Typically, a decision-maker chooses one optimal solution according to his preference. Therefore, the first difficulty in developing cognitive handoffs arises because there are many purposes, objectives, and goals all of them in conflict that need to be tradeoff.

A second significant difficulty emerges when numerous sources of environment information need to be considered to achieve the desired multiple purposes. Six sources of context we consider include: user, terminal, network, provider, application, and handoff process. Such sources produce context data that need to be collected, transformed, and distributed at the different handoff control entities (HCEs). The challenge is how to manage large amounts of unsorted high-dimensional data that have very complicated structures and at the same time reducing the signaling traffic overload produced by this task.

The last significant difficulty is originated by the different transition elements involved in the handoff process. These elements include radio channels, base stations, IP networks, service providers, user terminals, and all the feasible combinations. This variety of elements produces a large amount of scenarios that need to be considered for an adaptive handoff scheme.

B. Theoretical Background

First, we state the basis for establishing our methodology.

1) Holism and Reductionism: Holism and reductionism are two complementary and opposing approaches for analyzing complex systems [37]. They represent different views of the relationship between the whole and the parts. Holism states that parts cannot explain the whole, the whole states the behavior of parts; i.e., it is necessary to understand how the entire handoff system determines the behavior of its components. Conversely, reductionism states that parts can explain the whole, then the behavior of parts determine the behavior of the whole. We have seen how reductionist handoff schemes achieve its goals in specific scenarios but they quickly become special cases of more general models. Holistic models are more complex models that pretend to consider all the individual parts and to understand the purposes of the whole.

2) Model-based Design: The model-driven paradigm has emerged as one of the best ways to confront complex systems. As it was clearly expressed by Dr. Hoffman [38], models can capture both the structure of the system (architecture) and behavior (dynamism). Model-based systems engineering [39] helps to address complexity by raising the level of abstraction, enabling developers to view system models from many perspectives and different levels of detail while ensuring that the system is consistent. The Systems Modeling Language (SysML) [38,39] is becoming an accepted standard for modeling in the systems engineering domain. Using SysML for modeling helps to reduce ambiguity in models. In fact, models can now show the dynamic behavior of systems, including how they transition between states and how the system behaves overall.

3) Functional Decomposition: refers to the process of resolving a functional relationship into its constituent parts in such a way that the original function can be reconstructed from those parts by function composition. The process of decomposition [40] is undertaken for the purpose of gaining insight into the constituent components.

4) Design as Scientific Problem-Solving: In his inspiring paper, Braha [41] showed the similitude between the systems design process and the solving-problem process.

Therefore, we developed his foundation and proposed a methodology establishing a general procedure that starts with a problem statement and ends up with the solution deployment. This theory views the problem statement as the initial state and then, by searching through a state-space, reaches a goal state representing the solution.

C. Design and Development Procedure

The steps involved in a form of top-down procedure are:

1) Stating the problem: Develop a handoff procedure that can optimally achieve multiple desirable features simultaneously. The handoff procedure should be implemented for operating in real scenarios with multiple dimensions of heterogeneity. Then, as part of the problem:

a) Identify and analyze the required system functions: Study the desirable handoff features that need to be implemented and determine the purpose, objectives, and goals associated to every feature. Associate a clear and single purpose to every desirable feature. Decompose each purpose into one or more objectives by identifying the performance parameters that help to quantify the achievement of every purpose. In the same way, divide every objective into one or more specific handoff goals, using optimization values and handoff context data and

b) Determine the needed handoff context information: Establish what handoff criteria, handoff metrics, performance measures, handoff policies, handoff constraints, and handoff scenarios are needed to achieve every desired purpose. Study the availability, locality, dynamicity, structure, and complexity of the variables, policies, and constraints to use.

2) Design a subsystem structure or model-based framework: State a cognitive handoff conceptual model, i.e., identify all external context information as well as all internal context information with the highest abstraction level. Whilst internal data constitutes self-awareness, external data constitutes context-awareness of the handoff process. Then, using functional decomposition divide up the conceptual model into a number of sub-models. Every sub-model corresponds to a particular sub-problem that functionally is part of the whole handoff problem. The structure of the system may be represented with a hierarchy of models or framework enclosing the parts of the whole system organized through functional relations. Models in this framework describe the system behavior in an accurate and unambiguous way if one uses a finite set of states and a set of transition functions, thus to ease this part: Identify the associated system states and phases. These dynamic models can be formally represented using finite automata, Petri nets, timed automata, etc. [42]. The states or phases of the handoff process should describe a general behaviour rather than specific details of particular sub-models.

3) Execute the models: Execution of models allows verification and validation of such models. This is the difference between just drawing pictures and making pictures "live" as it was pointed out by Hoffmann in [38]. However, verification and validation should not be confused. Model verification means to test if the model satisfies its intended purposes or specifications. Model validation tests if the model provides consistent outcomes that are accurate representations of the real world. We use three strategies for these tasks: simulation, prototyping, and analysis. Whatever the strategy we choose, model testing or model checking [43] requires the use of a formal notation; e.g., modelling languages for simulation, mathematic and logic for analysis, and programming languages or middleware for model prototype implementation. If a model cannot be properly validated or verified, then it must be redesigned within the framework.

4) Implementation stages: Once all the models in the framework have been individually tested, the design problem now reflects a well-structured solution. A detailed design can now be generated considering the entire framework of models. This whole system design should be implemented in a whole system prototype. The final prototype is ready to be tested in-situ; should any failure occur during testing, then a review of the conceptual model or any sub-model in the framework should be performed.

5) Solution deployment: The cognitive handoff solution is ready to operate on a real handoff environment. The solution system (cognitive handoff) provides a simultaneous acomplishment of the multiple purposes defined by the handoff problem. Each purpose should be associated to quantitative objective functions to measure the degree in, which every handoff purpose was achieved.

VI. Applying the Model-Driven Methodology

Now, we are going to apply the previously proposed methodology to develop cognitive handoffs.

A. Purposes, Objectives, Goals, and Context Data

The handoff context information is extensive, heterogeneous, distributed, and dynamic. It supports the whole operation of the handoff process and the achievement of multiple desirable features. From the external and internal vision of the handoff environment, we have identified five external sources of context information (creating context-awareness) and one internal source, which is the handoff process itself (creating self-awareness):

1) User context: This context includes the user preferences, user priorities, user profiles, and user history and it is used to respond to user needs, habits, and preferences.

2) Terminal context: This context domain includes the following evaluating parameters: (i) Link quality: Received Signal Strength (RSS), Signal-to-Noise Ratio (SNR), Signal-to-Noise-and-Interference Ratio (SNIR), Bit Error Rate (BER), Block Error Rate (BLER), Signal-to-Interference Ratio (SIR), Co-Channel Interference (CCI), Carrier-to-Interference Ratio (CIR), etc.; (ii) Power management: Battery Type (BT), Battery Load (BL), Energy-Consumption Rate (ECR), Transmit Power in Current (TPC), Transmit Power in Target (TPT), and Power Budget (PB); (iii) Geographic mobility: Terminal Velocity (Vel), Distance from a Base Station (Dist), Geographic Location (Loc), Moving Direction (MDir), and Geographic Coverage Area (GCA). All these evaluating parameters allow the deployment of QoS-aware handoffs, power-based handoffs, and location-aided handoffs.

3) Application context: It includes the QoS requirements of running applications; Lost Packets (LP), Delayed Packets (DP), Corrupted Packets (CP), Duplicated Packets (DuP), Data Transfer Rate (DTR- goodput), Packet Jitter (PJ), Out-of-Order Delivery (OOD), Application Type (AppT).

4) Network context: This information is necessary to select among networks (before handoff), to monitor service continuity (during handoff), and to measure network conditions (after handoff) thus they are: Network Bandwidth (NBW), Network Load (NL), Network Delay (ND), Network Jitter (NJ), Network Throughput (NT), Network Maximum Transmission Unit (NMTU).

5) Provider context: Information about connection fees, billing models, roaming agreements, coverage area maps, security management (AAA), types of services (data, voice, video), provider preferences, and provider priorities.

6) Handoff performance context: This information forms the self-aware part of our cognitive model and allowing evaluation of its performance. Call Blocking (CB), Call Dropping (CD), Handoff Blocking (HOB), Handoff Rate (HOR), Handoff Latency (HOL), Decisions Latency (DLat), Execution Latency (ExLat), Evaluation Latency (EvLat), Handoff Type (HOType), Elapsed Time Since Last Handoff (ETSLH), Interruptions Rate (IR), Interruption Latency (IL), Degradations Rate (DR), Degradations Latency (DL), Degradations Intensity (DI), Utility Function (UF), Signaling Overload (SO), Security Signaling Overload (SSO), Improvement Rate (ImpR), Application Improvement Rate (AppImpR), User Improvement Rate (UsrImpR), Terminal Improvement Rate (TermImpR), Successful Handoff Rate (SHOR), Imperative Handoff Rate (IHOR), Opportunist Handoff Rate (OHOR), Dwell Time In the Best (DTIB), Authentication Latency (AL), Detected Attacks Rate (DAR), Online User Interventions Rate (OUIR), Tardy Handoff Rate (THOR), and Premature Handoff Rate (PHOR).

Once we have identified the context data from all the context sources and the desired handoff features that we wish to implement, then, we assign a qualitative purpose to every desired feature and, a set of quantitative objectives and goals to every handoff purpose. Tables III and IV summarize such previous description.

TABLE III. DESIRED FEATURES, PURPOSES, OBJECTIVES, AND GOALS

Desired Handoff Features	Qualitative	Quantitative	
	Purposes	*Objectives*	*Goals*
Seamlessness	Maintain continuity of services or preserve user communications	Reduce DR, DL, DI, IR, IL	Minimize (BER, BLER, CCI, NL,ND, NJ, LP, DP, CP, DuP, PJ, TPC, TPT, ECR, CB, CD, HOB, HOL) Maximize (RSS, SNR, SNIR, SIR, CIR, NBW, NT, NMTU, DTR, BL, ETSLH)
Autonomy	Preserve handoff operation independent of users	Reduce OUIR	Maintain (IL < app.Timeout)
Security	Maintain a constant level of security along the handoff	Reduce SSO, DAR	Minimize (AL, SO, HOL) Maintain (High Encryption)
Correctness	Keep user always connected to the best network with minimal handoffs	Reduce HOR Increase DTIB	Minimize (HOR) Maximize (DTIB)
Adaptability	Keep success of all handoff objectives across any scenario	Multi-objective optimal balance Increase SHOR	Keep every desirable feature within its success range. Maximize (SHOR)

TABLE IV. OTHER DESIRED PROPERTIES OF COGNITIVE HANDOFFS

Desired Handoff Features	Qualitative	Quantitative	
	Purposes	*Objectives*	*Goals*
Necessary	Prevent unnecessary handoffs	Start HO only if it is imperative or opportunist Maint. HOR = IHOR + OHOR	Imperative if (UFcurr<Thinf) Opportunist if (UFcurr>Thsup) UFtarget is SuffB & ConB
Selective	Avoid selecting the wrong target	Verify target is consistently better (ConB) and sufficiently better (SuffB)	SuffB: UFtarget > (UFcurr + Δ) ConB: SuffB is maintained for SP time
Efficient	Operate quickly and well-organized to decide how to perform the handoff (HO)	Select the best method, protocol, or strategy according to the HOType, AppType, and Mobility state. Reduce DLat, ExLat, EvLat	Define HO policies or conditions for choosing MIP, SIP, MAHO, NAHO, or other protocols
Beneficial	Augment benefits to applications, users, and terminals after handoff	Have a better UF after HO or a maximum improvement rate (UFnew/UFold)	ImpR >> 1 Maximize (AppImpR, UsrImpR, TermImpR)
Timely	Initiate a HO not tardy and not prematurely	Reduce THOR and PHOR	Maintain (DLat within its tolerance range)

These tables represent a relevant preliminary result of the applicability of cognitive handoff methodology. On one hand, they help to reduce the ambiguity and confusion on the usability of similar handoff features because every desirable handoff feature is defined in qualitative terms (purpose) and quantitative terms (objectives and goals). On the other hand, they help to correlate context data with desirable features. For instance, from Table III, we observe that RSS is correlated with seamlessness, IL with autonomy, AL with security, etc. This correlation is intended to select the context data that is needed to support every handoff purpose.

B. *Taxonomy of Handoff Mobility Scenarios*

A second significant result obtained from the proposed model-driven methodology is a new taxonomy of handoff mobility scenarios derived from combining all the possible transition elements involved in handoffs; i.e., channels, cells, networks, providers, and terminals. This taxonomy depicts all different kinds of handoffs that are possible in real networks.

Nowadays, no handoff solution exists, which comprehensively addresses the entire scale of heterogeneity. Multidimensional heterogeneity is the reason for the large number of handoff scenarios. If we define a handoff scenario as an array $(d_1, d_2 \ldots, d_n)$ where d_i is an instance of D_i the *i*th dimension of heterogeneity and there are $|D_i|$ different ways to instantiate the *i*th dimension, then by the multiplication principle there will be $|D_1|\times|D_2|\times\ldots\times|D_n|$ possible handoff scenarios. However, for the user mobility dimension, the array (location, velocity, direction) may have distinct values at any instant along the path with infinite paths crossing the network; therefore, the number of possible mobility scenarios is infinite. Despite of such infinite scenarios, it is important to make a classification of handoffs according to the elements involved during the transition.

The complexity and treatment for a handoff depend on the type of transition that is occurring. A handoff will require of services from distinct OSI model layers depending on the elements involved in the transition. For example, a handoff between channels of the same cell is a layer 1 handoff; a handoff between cells (base stations) is a layer 2 handoff, it is homogeneous if cells use the same wireless technology, otherwise is heterogeneous; a handoff between IP networks is a layer 3 handoff; a handoff from one provider to another or between user terminals will demand the services of layers 4-7.

Fig. 5 depicts the hierarchical structure of a mobile Internet in a four-layer design (core, distribution, access, and mobile). We will use this figure to explain a handoff hierarchy that involves channels, cells, networks, providers, and terminals.

The mobile Internet is divided into independent administrative units called Autonomous Systems (AS). An AS is a network administrated by a single organization or person. The Internet is a network of autonomous systems.

Fig. 5 presents two autonomous systems called ISP1 and ISP2 for two distinct service providers. Every ISP uses a very high-speed core network where main servers are located. Providers divide their distribution networks, physically and logically, into a number of IP networks, subnets, or VLANs (Virtual LANs), where the types of services and users are separated. Each IP Net includes a group of base stations (BS) or access points with the same or different wireless access technology. Base stations get distributed across a geographic area to offer mobile communication services. Each base station controls a cell that may have a group of channels to distribute among the associated terminals or a single channel that is shared among several associated terminals.

In Fig. 5, BS2 illustrates a layer 1 handoff when the mobile terminal (MT) changes its connection between channels ch1 and ch2 without changing of BS, IP Net, ISP, or MT. A layer 2 handoff is illustrated between BS1-BS2, BS3-BS4, BS5-BS6, and BS7-BS8. Note that layer 2 handoff changes from one channel to another and from one base station to another, but keeps the same IP Net, ISP, and MT; however, if the cells involved are heterogeneous, then the handoff is *vertical*, otherwise is *horizontal*. A layer 3 handoff is depicted in BS2-BS3 and BS6-BS7. Note that layer 3 handoff changes from one channel to another, from one cell to another, and from one IP network to another, but preserves the same provider and the same terminal; the layer 3 handoff may be heterogeneous, like in BS2-BS3, or homogeneous, like in BS6-BS7. We represent a layer 4-7 handoff, in BS4-BS5, when MT changes its communications from on channel to another, from one cell to another, from one IP Net to another, and from one ISP to another, but the user keeps the same terminal.

The encryption schemes and data representation formats change from one provider to another, thus higher layer services are required. Inside the cell for BS5 we depict a handoff between terminals where the user transfers the whole session (current state of running applications) from terminal MT-A to terminal MT-B. Handoffs between terminals can be done for terminals within the same cell or different cells, within the same IP network or different IP networks, within the same provider or different providers. The terminal handoff depicted in BS5 keeps the same cell, same IP Net, and same ISP.

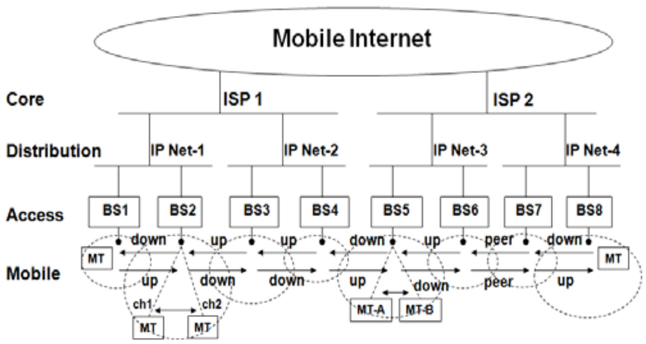


Figure 5. Hierarchy of handoff mobility scenarios. Different overlay sizes for macro, micro, pico, and femto cells.

Fig. 6 presents a process diagram that generates the complete taxonomy of handoffs by following the different paths from the upper node to the lower nodes.

Every handoff type in this taxonomy should be complemented or further classified according to many other criteria by using the handoff classification tree of *Nasser et al* in [44].

C. *Cognitive Handoff State-Based Model*

By applying the second step of the model-driven methodology, design a subsystem structure, we created a cognitive handoff conceptual model and its first decomposition model both illustrated and discussed in Section III. Following the reductionist approach, we now focus on a major component of the handoff system, the cognitive handoff control system. At this stage, we designed a state-based model whose purpose is to understand the general behavior that should have the handoff control system. Thus, this model represents our third main result obtained from following the methodology.

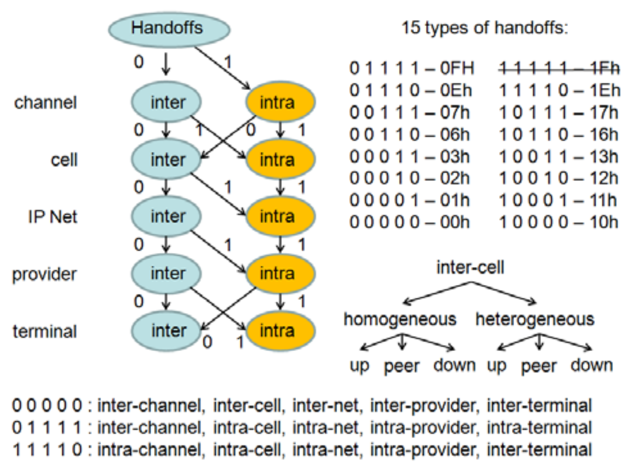


Figure 6. Generation process for handoff taxonomy. There are 15 types of feasible handoffs that can be implemented in real wireless overlay networks. The 1Fh is not a handoff.

Fig. 7 shows a five-state diagram modeling a general control handoff process. The states are: (1) Disconnection, (2) Initiation, (3) Preparation, (4) Execution, and (5) Evaluation. This model describes a generic control handoff system coordinating the stages before, during, and after the handoff.

We describe each state briefly:

1) ***Disconnection:*** is the initial state and one of the two final states. Here, the terminal is disconnected but discovering available networks. The process will stay here while there are no available networks.

2) ***Initiation:*** in this state the terminal is connected to the best available network and communications flow normally. This is another final state. The process stays here while there are no reasons (imperative or opportunistic [45]) to prepare for a handoff. If current connection breaks and no other network is available, then the process goes back to the disconnection state.

3) ***Preparation****:* as soon as a better network appears, the process changes to the preparation state. Here is where properly the handoff begins. This state decides why, where, how, who, and when to trigger the handoff. The handoff in progress can be rolled back to initiation if current link becomes again the best one.

4) ***Execution****:* once a control entity decides to trigger a handoff, there is no way to rollback; the handoff will be performed. This state knows the current and destination networks, the active application to be affected, and the strategy or method to use.

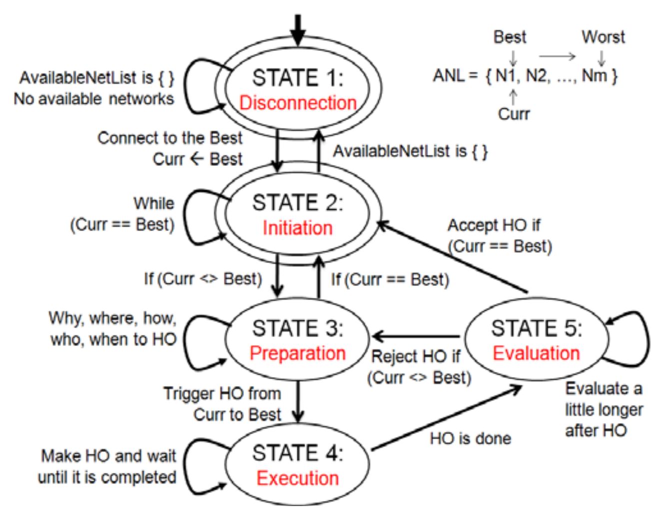


Figure 7. A handoff control model. This state diagram shows a reactive and deterministic behavior of cognitive handoffs.

5) ***Evaluation****:* once the link switch is made, the control entity enters the evaluation state. This state recombines the measures for every objective function taken before and during the handoff, with new samples taken after the handoff to determine its successfulness. The evaluation latency is adjusted to a stabilization period [46].text edit has been completed, the paper is ready for the template. Duplicate the template file by using the Save As command, and use the naming convention prescribed by your conference for the name of your paper. In this newly created file, highlight all of the contents and import your prepared text file. You are now ready to style your paper.

VII. Discussion

In this research, we have shown a new methodology to systematically develop cognitive handoffs, which are expected to be in operation in the mobility scenarios of the future Internet. Such methodology is based on a sound theoretical framework including: methods for analyzing complex systems, the model-based systems engineering, the functional decomposition approach, and the scientific problem-solving theory. There are five stages in the proposed methodology: 1) state the problem, 2) design a model-based framework, 3) execute the models, 4) implement a prototype, and 5) deploy the solution. Thus, we have presented three main results obtained from applying the first two stages of the methodology: i) a cascade relationship of desired features, purposes, objectives, goals, and context data; ii) a taxonomy of handoff mobility scenarios; and iii) a generic state-based model for a cognitive handoff control system.

Furthermore, there are some other issues that require detailed discussion: (a) the complexity of a cognitive handoff system, (b) the evaluation of cognitive handoff models, and (c) the implementation of cognitive handoffs.

A. Cognitive Handoff Complexity

In Section III we showed two main properties of complex systems that are also present in cognitive handoffs: the hierarchic structure of systems and the property of emergence. Now, in this section we provide other reasons of why cognitive handoffs are complex software systems: (1) Cognitive handoffs exhibit a rich set of behaviors: reactive, proactive, deterministic, non-deterministic, context-aware, self-aware, etc.; behavior is determined by the particular desirable features associated to handoffs. (2) Cognitive handoffs can be stated as multi-objective optimization problems. (3) Cognitive handoffs are driven by events in the physical world; e.g., the user mobility, the user preferences, the provider services, the coverage areas, etc. (4) Cognitive handoffs maintain the integrity of hundreds or thousands of records of information while allowing concurrent updates and queries. (5) Context information is extensive, heterogeneous, dynamic, and distributed. (6) Cognitive handoffs control real-world entities, such as the switching of data flows through a large set of available networks, providers, and terminals. (7) Handoff management has a long-life span; handoffs will exist in all future wireless networks. (8) Handoff management is a key issue for wireless industry and standardization bodies. Grady Booch in [46] provides further discussion on the attributes of complex software systems.

B. Evaluation of Cognitive Handoff Methodology and Models

Now, as a result of applying our proposed methodology, one gets a set of models that are different in purpose (intentions), usability (applicability), notation (language), and abstraction (hierarchy).

Methodology and each model must be evaluated, either by quantitative evaluation, which comprises the definition of criteria and metrics intended to measure one specific property or, conversely by a qualitative evaluation, which is related to credibility that comes from the way in, which the cognitive maps are built and the clarity it represents the opinion's of most experts [48].

In relation to a qualitative evaluation of the methodology, one requires to think on the stages proposed by the development process, the kind of activities to accomplish in each stage, the strength of its theoretical basis, the kind of lifecycle in the development process, etc. Meanwhile, corresponding quantitative evaluation, metrics should be applied to all associated parametres in the stages of the process.

With respect to evaluate models, we made a clear distinction in Section II.C between verification and validation. The verification tests if the model satisfies its purpose, whilst validation tests if the model outcomes are representations of reality. During the development process of a new system, special purpose models are built to support the understanding that goes on during the development and no hard data emerge from such models, thus, they can only be verified, but not validated.

It is worth to notice that in this paper, we deal with a specific kind of model belonging to those known as soft models [48]. Soft models are intended to understand rather than to predict and therefore verification is the way to qualitatively evaluate such models. Specifically, the theoretical framework in Section IIB has solid and proven bases.

C. Cognitive Handoff Implementation

We envision the implementation of cognitive handoffs as a network of distributed agents cooperating and competing to take any type of handoff to success. We distinguish between agents for controlling the handoff process (HCEs) and agents for managing the handoff context data (CMAs). The CMAs are responsible for recollecting the context data and updating the handoff information base at the HCEs. CMAs are located in user terminals and distributed in different layers of the network infrastructure. HCEs are located also in every user terminal and at the network access layer; HCEs perform a handoff control process like the one depicted in Fig. 3. Thus, let us develop the state-based model as follows.

A dynamic ordered list of available networks (ANL) is organized from best to worst, according to the value of desirability calculated for every network. The desirability metric is a utility function combining a broad set of network selection criteria. The best network is the one with highest desirability. The value of desirability for the *n*th network, named $D_n(\mathbf{V})$, may have a geometric or stochastic distribution depending on the dynamic nature of context variables used as selection criteria, and arranged in a criteria vector $\mathbf{V} = (v_1, v_2, ..., v_m)$. We use Equation (1) to represent a general mathematical model for the desirability function:

$$D_n(\mathbf{V}) = \sum (k + w_i)\log(v_i^+) - \sum (k + w_j)\log(v_j^-) \quad (3)$$

The set of decision variables $(v_1, v_2, .., v_m)$ fetched for the *n*th available network is partitioned in two subsets: V_i^+ and V_j^-; where V_i^+ is the set of criteria that contribute to the desirability (e.g., NBW and NT) and V_j^- is the set of variables that contribute to the undesirability (e.g., NL and ND). w_i and w_j are weights corresponding to each variable such that $w_l \in \Re[0,1]$, $\sum w_l = 1$ and k .is a scaling factor so that small changes in the context variables reflect big changes in $D_n(\mathbf{V})$.

For geometric distributions, a proactive handoff strategy may anticipate handoff decisions and for stochastic distributions a reactive handoff strategy with thresholds, hysteresis margins, and dwell-timers may prevent unnecessary handoffs. The control handoff process illustrated in Fig. 3 shows a reactive and deterministic procedure; reactive, because the process starts the preparation for a handoff until another network with higher desirability is present and, deterministic, because it is always possible to determine the current state of the process within one of five states.

Fig. 8 and Fig. 9 depict geometric distributions of desirability with different handoff strategies. Fig. 8 shows a proactive strategy where the handoff preparation starts before the target network improves the current connection. Fig. 9 shows a reactive strategy where handoff preparation starts after the target network has improved the current connection.

The darken line over the desirability functions illustrate the current connection. The performance parameters ΛPREP, ΛEXEC, ΛEVAL, and ΛVHO depict the latencies for the different stages: preparation, execution, and evaluation. Configuration parameters include Δ (hysteresis margin), desirability threshold (Thsup, Thinf), and dwell-timer (SP). Relative Desirability measures are (ΔRs), which are equal to |Dcurr – Dbest|.

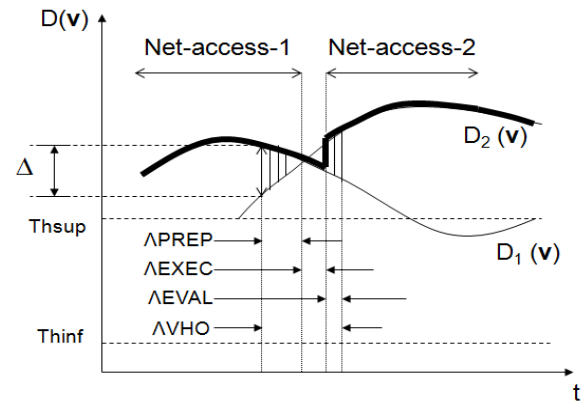


Figure 8. A proactive handoff strategy.

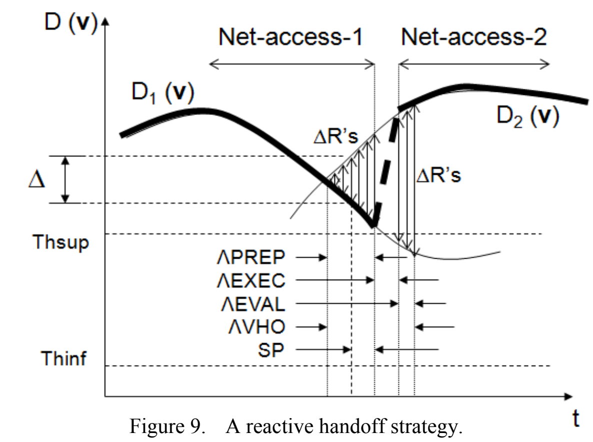


Figure 9. A reactive handoff strategy.

The available network list (ANL) is a data structure located at the HCEs, but continuously updated by the CMAs. When the ANL is empty, the terminal goes to the disconnection state (State 1) and stays there while such list is empty. CMAs are continuously discovering new networks and ordering the list from the highest desirable networks to the lowest desired networks.

The change from disconnection state to initiation state (State 2) occurs as soon as new networks are available. The HCE selects the best available network from the list and connects the terminal to it. The State 2 is the Always Best Connected state because the terminal will stay connected to the best network as long as no other available network improves the current connection.

The change from initiation to preparation (State 3) occurs when a new network is improving or has improved the current network. Handoff decisions, in State 3, start by identifying a reason to begin the preparation for a handoff (why). Next, selecting the target network (where). Then, deciding what strategy, method, or protocol to choose (how). Then, deciding what HCE will be responsible to trigger the handoff (who), and finally, deciding the best moment to trigger the handoff (when). The chosen handoff strategy, method, or protocol depends on the current handoff scenario (as those depicted in Fig. 5) and the type of handoff in progress (as those illustrated in Fig. 6).

The decision to trigger a handoff in one terminal changes the control process from preparation to execution (State 4). The trigger handoff decision activates a procedure to change the data flows of an application from one access network to another, within specific handoff and time constraints. The switching mechanism takes a time ΛEXEC to complete.

Once the switching process is completed, the HCE enters to the evaluation state (State 5). This is an important stage of feedback to the handoff control process. At this stage, the HCE has a constrained period of time to decide to accept or reject the recently executed handoff. One condition for handoff success occurs if the new current connection is the best available connection, but others include measuring the objective functions, associated to every handoff purpose, and if all these measures are within a boundary region of acceptable quality, then the cognitive handoff is successful, otherwise it is defective and outliers should be corrected.

VIII. Model Results

So far we have described a challenging handoff optimization problem and we have created a series of models to study the problem. Moreover, we proposed both: a computational model that offers a heuristic solution to the problem and a methodology to implement cognitive handoffs. Therefore, now we are interested in a simulation instrument that can help us to validate the behavior of a given specific handoff algorithm over a variety of handoff scenarios based on time, space, or both and measure particular performance quantities. To this end, we created a Relative Desirability Handoff Algorithm with hysteresis, dwell-timers, and two thresholds in order to make a terminal stay most of the time on the best network, while it performs the fewer number of handoffs on most handoff scenarios.

Fig. 10 shows an example that considers our particular cognitive handoff algorithm and a user defined valid handoff scenario. The handoff scenario consists of two networks, one that changes abruptly and rapidly and another that changes smoothly and slowly. Lower and upper thresholds are defined within the visual area, $L = -1$ and $U = 4$, separating the graphics into three handoff regions.

The bottom thick line depicts the current network passing through different handoff states: initiation (black), preparation (blue), evaluation (pink), disconnection and execution (red).

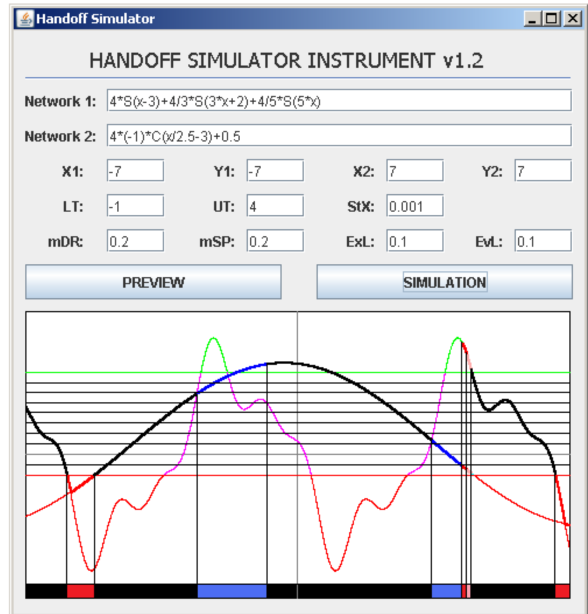


Figure 10. Visual outputs of handoff simulator with additional visual aids

Each test in the virtual .instrument displays graphically the behavior of the handoff algorithm and yields handoff performance data which, are collected in a structured file. The handoff collected data include handoff performance measures and the handoff scenario.

We design a nondeterministic experiment for collecting representative samples of input handoff scenarios which, will be used to test our proposed algorithm. The algorithm performs a cognitive handoff from the current network to the best candidate network in order to stay in the best available connection most of the time; i.e., increase DTiB. Simultaneously, this algorithm tries to perform the fewer number of handoffs because each handoff entails some overload to communications; i.e., decrease nEHO. However, these tasks are in conflict, they cannot be improved simultaneously. As a result, this algorithm makes a balance between increasing DTiB and decreasing nEHO. The way of doing this balance is by delaying the execution of a handoff until it becomes really necessary, i.e., until the candidate network becomes sufficiently and consistently better.

This algorithm obtains three performance measures (rTiB, rEHO, rBHO) which, are associated to the particular handoff scenario under analysis. Values for rTiB ≥ 50% or rEHO ≤ 50% are considered good or acceptable results.

In this experiment, we ask three users to define at their own will several statistically valid scenarios User “A” made 32 trials, user “B” 84, and user “C” 133, which, gives a total sample size of 249 tested scenarios.

Since for each input scenario, the instrument records and measures three handoff performance parameters: rTiB, rEHO, and rBHO, then the space of handoff results will be composed of data obtained from each test. By observing the distribution of sample data within the space of results, we may compute the degree of achievement of each performance goal.

Fig. 11 shows a scatter diagram of 133 random sample points obtained by user "C". The graphic presents 17 samples in the space for very good results and very good balance which, represent a percentage of 12.78%. It includes 95 (17+78) samples in the space for good results and good balance, which represent the 71.43% of the sample size; and, 121 (17+78+26) samples in the space for good results which, represent a hit rate of 90.98%. The diagram also illustrates 12 sample points located in the space for bad results, representing a 9.02%. The random experiment of 133 samples meets all the percentage goals: for good results (90.98% > 90%), for good results and good balance (71.43% > 50%), and for very good results and very good balance (12.78% > 10%).

Table V presents a summarization of results taken from the testing experiment of the handoff instrument. This table compares the percentages of sample points falling in each region of handoff results with the different random samples obtained from the experiments. It can be seen that the hit rates in all testing cases meet the handoff performance goals.

The handoff simulation instrument produced, in average, a rate of "good" results above 90% or a rate of "bad" results below 10%, a rate of "good" results and "good" balance above 50%, and a rate of "very good" results and "very good" balance above 10%.

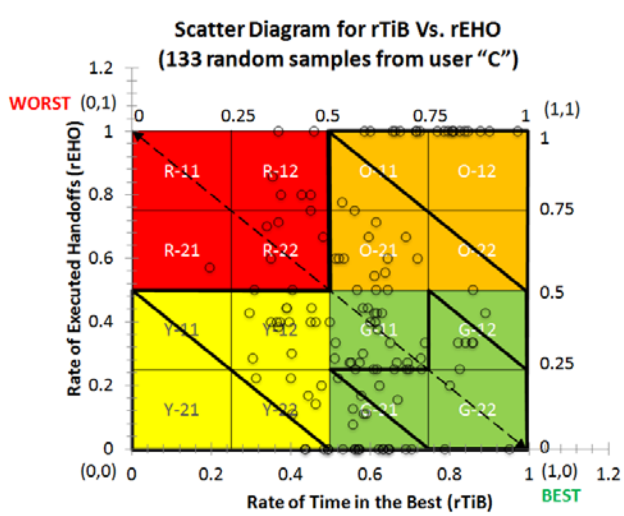


Figure 11. Scatter diagram for rTiB vs. rEHO 133 observations made by user "C".

TABLE V SUMMARY OF TEST RESULTS USING THE HANDOFF INSTRUMENT

Scenarios / Results	user A 32	user B 84	user C 133	All users 249	Perfor-mance Goals
Very good results & very good balance	34.48%	14.29%	12.78%	16.06%	> 10%
Good results & good balance	87.5%	54.76%	71.43%	71.43%	> 50%
Good results	90.63%	92.86%	90.98%	91.57%	> 90%
Bad results	9.37%	7.14%	9.02%	8.43%	< 10%

Therefore, all these results provide evidence that support the correctness of our proposed algorithm based on our cognitive handoff model and methodology as well the usefulness of the taxonomy to properly define scenarios.

IX. CONCLUSION AND FUTURE WORK

Handoffs are an integral component of any mobile-wireless network from past, present, and future. Handoffs are transitions that change the data flows from one entity to another, where these entities may be radio channels, base stations, IP networks, service providers, and user terminals. The handoff process should exhibit several desirable features beyond seamlessness and should consider more context information beyond the signal strength. This is a common requirement to face the handoff scenarios of the future Internet.

The existing handoff schemes are not able to achieve a variety of attractive features and managing arbitrary amounts of context information. Therefore, we proposed a conceptual model to create handoffs of this kind. We characterized a cognitive handoff to be multipurpose, multi-criteria, context-aware, self-aware, and policy-based.

We claimed that our cognitive handoff model is holistic because it considers all the transition entities that may be involved in handoffs, all the external and internal sources of context, and considers many significant desirable features.

Using a functional decomposition approach, we divided the functional behavior of a cognitive handoff into six general modules: control algorithm, network discovery, handoff decisions, handoff execution, handoff evaluation, and context management. Each module has assigned a purpose to every feature and decomposed each purpose into objectives and goals. We applied the cognitive handoff model to define its performance parameters and significant tradeoffs between conflicting objectives.

We proposed a new model-driven methodology for developing cognitive handoffs. We applied the proposed methodology and obtained a clear relationship between handoff purposes and handoff context information, a new taxonomy of handoff scenarios, and an original state-based model of a generic control handoff process.

We continue developing and integrating the models generated by the cognitive handoff methodology. A future work is to organize such models in a comprehensive framework of models representing the functional issues for the whole cognitive handoff process. Further work is needed to study the availability, locality, dynamicity, structure, and complexity of variables, metrics, polices, and constraints involved in cognitive handoffs.

ACKNOWLEDGMENT

F.A. González-Horta is a PhD candidate at INAOE Puebla, Mexico, and thanks the financial support received from CONACYT Mexico through the doctoral scholarship 58024.

REFERENCES

[1] F. A. González-Horta, R.A. Enríquez-Caldera, J. M. Ramírez-Cortés, J. Martínez-Carballido and E. Buenfil-Alpuche, “Towards a cognitive handoff for the future Internet: Model-driven methodology and taxonomy of scenarios” *2nd International Conference on Advanced Cognitive Technologies and Applications,* COGNITIVE 2010, Lisbon, Portugal, Nov. 2010.

[2] F. A. González-Horta, R. A. Enríquez-Caldera, J. M. Ramírez-Cortés, J. Martínez-Carballido, and E. Buenfil-Alpuche, “Towards a cognitive handoff for the future Internet: a holistic vision,” *2nd International Conference on Advanced Cognitive Technologies and Applications,* COGNITIVE 2010, Lisbon, Portugal, Nov. 2010

[3] M. Satyanarayanan, M. A. Kozuch, C. J. Helfrich, and D. R. O’Hallaron, “Towards seamless mobility on pervasive hardware” *Pervasive and Mobile Computing 1*, Elsevier, pp. 157-189, 2005.

[4] A. Singhrova and N. Prakash, “Adaptive Vertical Handoff Decision Algorithm for Wireless Heterogeneous Networks” *11th IEEE Intl. Conf. on High Performance Computing and Communications*, pp. 476-481, 2009.

[5] J. M. Kang, H. T. Ju, and J. W. K. Hong, “Towards Autonomic Handover Decisions Management in 4G Networks” A. Helmy et al. (Eds.): MMNS 2006, LNCS 4267, IFIP, pp. 145-157, 2006.

[6] K. Sethom, H. Afifi, and G. Pujolle, “Secure and Seamless Mobility Support in Heterogeneous Wireless Networks” *Proc. IEEE Globecom*, pp. 3403-3407, 2005.

[7] K.D. Wong and D.C. Cox, “A Pattern Recognition System for Handoff Algorithms” *IEEE Journal on Selected Areas in Communications* 18 (7), pp. 1301-1312, July 2000.

[8] N. D. Tripathi, “Generic adaptive handoff algorithms using fuzzy logic and neural networks,” Ph.D. dissertation, Virginia Polytechnic Institute and State University, August 21, 1997.

[9] A. Altaf, F. Iqbal, and M. Y. Javed, “S3H: A secure, seamless and soft handover between WiMAX and 3G networks,” *Intl. Conf. on Convergence and Hybrid Information Technology*, pp. 530-534, 2008.

[10] L. R. Cardenas, M. Boutabia, and H. Afifi, “An infrastructure-based approach for fast and seamless handover,” *The Third International Conference on Digital Telecommunications*, pp. 105-109, 2008.

[11] A. Singhrova and N. Prakash, “Adaptive vertical handoff decision algorithm for wireless heterogeneous networks,” *11th IEEE Intl. Conf. on High Performance Computing and Communications*, pp. 476-481, 2009.

[12] N. D. Tripathi, “Generic Adaptive Handoff Algorithms Using Fuzzy Logic and Neural Networks” Ph.D. dissertation, Virginia Polytechnic Institute and State University, August 21, 1997.

[13] N. Nasser, A. Hasswa, and H. Hassanein, “Handoffs in Fourth Generation Heterogeneous Networks” *IEEE Communications Magazine*, pp. 96-103, October 2006.

[14] M. Ylianttila, “Vertical Handoff and Mobility – System Architecture and Transition Analysis” Ph.D. dissertation, Faculty of Technology, Dept. of Electrical and Information Engineering, University of Oulu, Finland, May 6, 2005.

[15] K. Pahlavan, P. Krishnamurthy, A. Hatami, M. Ylianttila, J. P. Makela, R. Pichna, and J. Vallström, “Handoff in Hybrid Mobile Data Networks” *IEEE Personal Communications*, pp. 34-47, April 2000.

[16] J. W. So, “Vertical Handoff in Integrated CDMA and WLAN Systems” *International Journal of Electronics and Communications* (AEÜ) 62, Elsevier, pp. 478-482, 2008.

[17] Z. Zhang and A. Boukerche, “A Novel Mobility Management Scheme for IEEE 802.11-based Wireless Mesh Networks” *Intl. Conf. on Parallel Processing*, pp. 73-78, 2008.

[18] F. Siddiqui and S. Zeadally, “Mobility Management across Hybrid Wireless Networks: Trends and Challenges” *Computer Communications* 29, Elsevier, pp. 1363-1385, 2006.

[19] A. Hasswa, N. Nasser, and H. Hassanein, “Generic Vertical Handoff Decision Function for Heterogeneous Wireless Networks” *IEEE*, 2005.

[20] S. R. Tuladhar, C. E. Caicedo, and J. B. D. Joshi, “Inter-Domain Authentication for Seamless Roaming in Heterogeneous Wireless Networks” *IEEE Computer Society*, pp. 249-255, 2008.

[21] W. T. Chen, J. C. Liu, and H. K. Huang, “An Adaptive Scheme for Vertical Handoff in Wireless Overlay Networks” *Proc. 10th Intl. Conf. on Parallel and Distribution Systems*, 2004.

[22] A. Altaf, F. Iqbal, and M. Y. Javed, “S3H: A Secure, Seamless and Soft Handover between WiMAX and 3G Networks” *Intl. Conf. on Convergence and Hybrid Information Technology*, pp. 530-534, 2008.

[23] C. Prehofer, N. Nafisi, and Q. Wei, “A framework for context-aware handover decisions,” The 14th IEEE 2003 International Symposium on Personal, Indoor and Mobile Radio Communication Proceedings, PIMRC 2003, pp. 2794-2798, vol. 3, 7-10 Sept. 2003.

[24] P. Pawar, B. J. van Beijnum, M. van Sinderen, A. Aggarwal, P. Maret, and F. De Clercq, “Performance evaluation of the context-aware handover mechanism for the nomadic mobile services in remote patient monitoring,” Computer

Communications, Volume 31, Issue 16, pp. 3831-3842, Elsevier, October 25,2008.

[25] M. Emmelmann, S. Wiethoelter, A. Koepsel, C. Kappler, and A. Wolisz, “Moving toward seamless mobility: state of the art and emerging aspects in standardization bodies” *Wireless Pers Commun* 43, Springer, pp. 803-816, 2007.

[26] B. A. Black, P. S. Dipiazza, B. A. Ferguson, D. R. Voltmer, F. C. Berry, “Introduction to Wireless Systems” Ch. 4, Prentice Hall, 1st Edition, pp. 125-140, May 2008.

[27] J. L. Salina and P. Salina, “Next Generation Networks: Perspectives and Potentials” John Wiley & Sons, England, 2007.

[28] S. Mohanty and J. Xie, “Performance Analysis of a Novel Architecture to Integrate Heterogeneous Wireless Systems” *Computer Networks* 51, Elsevier, pp. 1095-1105, 2007.

[29] P. Koch and R. Prasad, “The Universal Handset” *IEEE Spectrum*, vol. 6 num. 4 International, pp. 32-37, April 2009.

[30] Q. H. Mahmoud (Edt), “Cognitive Networks: Towards Self-Aware Networks” Foreword 2: S. Dixit, J. Wiley & Sons, 2007.

[31] E. Gustaffson and A. Jonsson, “Always Best Connected: 3G Mobile Network Technologies and Experiences” *IEEE Wireless Communications*, pp. 49-55, February 2003.

[32] M. Pidd (Edt), “Systems Modelling: Theory and Practice,” John Wiley & Sons, England, pp. 1-42, 2004.

[33] O. Maimon and L. Rokach, “Decomposition Methodology for Knowledge Discovery and Data Mining: Theory and Applications” Series in Machine Perception Artificial Intelligence, Vol. 61, World Scientific Pub., London, 2005.

[34] Q. Song and A. Jamalipour, “A Time-Adaptive Vertical Handoff Decision Scheme in Wireless Overlay Networks” *17th Annual Intl. Symposium on Personal, Indoor, and Mobile Radio Commun*, PIMRC’06, 2006.

[35] J. Heer, M. Bostock, and V. Ogievetsky, “A Tour Through the Visualization Zoo” *Communications of the ACM*, vol. 53, no. 6, pp. 59-67, June 2010.

[36] J. Branke, K. Deb, K. Miettinen, and R. Slowinski (Eds.), “Multiobjective Optimization: Interactive and Evolutionary Approaches” Springer, Germany, 2008.

[37] V. V. Raman, “Reductionism and holism: two sides of the perception of reality,” The Global Spiral, an e-publication of Metanexus Institute, published on July 15, 2005. URL: http://www.metanexus.net/magazine/tabid/68/id/9338/Default.aspx, retrieved on September 4, 2010.

[38] H. P. Hoffman, C. Sibbald, and J. Chard, “Systems engineering: the foundation for success in complex systems development,” IBM Corporation (white paper), Software Group, December 2009, pp. 1-11.

[39] D. M. Buede, The Engineering Design of Systems: Models and Methods, 2nd Edition, John Wiley & Sons, USA, 2009.

[40] O. Maimon, and L. Rokach, Decomposition Methodology for Knowledge Discovery and Data Mining: Theory and Applications, Series in Machine Perception Artificial Intelligence, Vol. 61, World Scientific Publishing Co., London, 2005.

[41] D. Braha, and O. Maimon, “The design process: properties, paradigms, and structure,” *IEEE Trans. On System, Man, and Cybernetics – Part A: Systems and Humans*, 27(2), March 1997, pp. 146-166.

[42] P. A. Fishwick (Ed.), Handbook of Dynamic System Modeling, Chapman & Hall /CRC, USA, 2007.

[43] C. Baier and J.P. Katoen, Principles of Model Checking, The MIT Press, USA, 2008.

[44] N. Nasser, A. Hasswa, and H. Hassanein, “Handoffs in fourth generation heterogeneous networks” *IEEE Communications Magazine*, pp. 96-103, October 2006.

[45] W. Zhang, J. Jaehnert, and K. Dolzer, “Design and evaluation of of a handover decision strategy for 4th generation mobile networks,” The 57th Semiannual Vehicular Technology Conference, VTC 2003, Jeju, Korea, 2003.

[46] H. J. Wang, R. H. Katz, and J. Giese, “Policy-enabled handoffs across heterogeneous wireless networks,” *WMCSA 99*, New Orleans, Louisiana, USA, 1999.

[47] G. Booch, R. A. Maksimchuk, M. W. Engle, B. J. Young, J. Conallen, and K. A. Houston. Object-Oriented Analysis and Design with Applications, Third Edition, Addison-Wesley, Chap. 1, 2007.

[48] M. Pidd (Ed.), Systems Modelling: Theory and Practice, John Wiley & Sons, England, pp. 1-42, 2004.

Evaluation of Middleware for Bandwidth Aggregation using Multiple Interface in Wireless Communication

Etsuko Miyazaki
Ochamonizu University
2-1-1 Ohtsuka, Bunkyo-ku 112-8610
Tokyo, Japan
(Currently with Microsoft Corp.)

Masato Oguchi
Ochamonizu University
2-1-1 Ohtsuka, Bunkyo-ku 112-8610
Tokyo, Japan
Email: oguchi@computer.org

***Abstract*—Although a variety of wireless interfaces are available on mobile devices, they still provide only low throughput so far. When coverage areas of those different technologies overlap, mobile devices with multiple interfaces can use them simultaneously by mechanism of Bandwidth Aggregation. However, there are some performance problems for Bandwidth Aggregation on Network Layer and lower Layer which derive from TCP congestion control mechanism. If Bandwidth Aggregation is performed at a layer lower than Transport layer, a packet loss happend in one route should decrease performance of all routes because reducing TCP congestion window on such a case affects communications of all routes. Thus we have proposed advanced Bandwidth Aggregation on Middleware for the purpose of avoiding there problems. If Bandwidth Aggregation is performed at Middleware that locates between Transport layer and applications, TCP congestion windows can be managed separately as route by route. In this paper, we have evaluated Middleware for Bandwidth Aggregation, which includes throughput and buffer size of receiver Middleware. According to the evaluation, it is possible to prevent from performance degradation when a packet loss happens using Middleware with an appropriate size of buffer at receiver-side.**

***Keywords*-component; Bandwidth Aggregation; Multiple Interface; Middleware; Buffer Size; IEEE 802.11; TCP Congestion Window**

I. INTRODUCTION

The growth of mobile Internet communication stimulate developments of a variety of wireless technologies: for example, IEEE 802.11, Bluetooth, and Worldwide Interoperability for Microwave Access (WiMAX). Although some of them have relatively broad bandwidth, they still have lower throughput than wired connection such as Ethernet, and are able to be accessed only in limited areas. It is possible to realize more efficient mobile Internet service using multiple interfaces simultaneously, when we are in areas covered by several services of wireless technologies. Bandwidth Aggregation which uses multiple interface simultaneously is proposed as advanced way to access Internet from mobile node.

Among several research works, seamless vertical handoff from one interface to another has been addressed [2]. One of the advanced form of this technology is known as cognitive radio. In such a system, it is possible to change a wireless connection from one radio wave frequency band to another, depending on the condition of radio wave. There are two types in cognitive radio; one is shared frequency type in which available frequency is chosen and used dynamically, and the other is heterogeneous type in which different kinds of wireless systems such as IEEE802.11 WiFi, WiMAX, and Long Term Evolution (LTE) have been chosen and used dynamically. These technologies have already been practical and expected to begin its commercial service in the near future. For example, NEC Corp. has demonstrated that, based on OpenFlow technology [3], it is possible to change a connection from WiFi to WiMAX dynamically based on the load of each connection [4].

Although we are able to change from one wireless connection to another, we have not achieved Bandwidth Aggregation in practical use. This is a little more complicated than vertical handoff, in which cognitive radio technologies and aggregation technologies should be considered simultaneously. There are not only implementation difficulties but also performance matter for aggregating multiple connections. If Bandwidth Aggregation is realized in a mobile environment, this gives us better mobility support, reliability and resource sharing.

Thus, we have proposed and evaluated an innovative mechanism of Bandwidth Aggregation in this paper. While we have focused on aggregation of several WiFi connections, it is possible to apply this proposal and the evaluation results to heterogeneous aggregation of WiFi, WiMAX, and LTE, for example.

The rest of paper is organized as follows. First, background of Bandwidth Aggregation is discussed in Section II. In Section III, our proposed model for Bandwidth Aggregation is introduced. An outline of evaluation of the proposed method is mentioned in Section IV. Various scenarios of experiments are introduced and buffer size of receiver Middleware is evaluated in Section V. In Section VI, the method of Bandwidth Aggregation on Network layer and the proposed methods are compared. Finally, concluding remarks are related in Section VII.

II. Background of this research work

A. Bandwidth Aggregation in Various Layers

Bandwidth Aggregation is supposed to be realized on several layers, while they have merits and demerits respectively.

An approach on Datalink layer [5] will give the most effective result, and upper layers do not need to care about Bandwidth Aggregation. However, we can install it only world using same protocol for datalink layer and have to install specific hardware to their nodes. In other words, all network interfaces should be replaced to use this approach.

An implementation in Network layer will provide efficient Bandwidth Aggregation by intelligent methods [6][7]. The advantages using Network layer are they perform transparently to widely used Transport protocol such as TCP and UDP. However, TCP may not achieve estimated efficiency due to a possibility that they receive packets in incorrect order. Although only the incorrect order might not necessarily be a problem, this causes congestion control more than required. That is to say, this may cause unnecessary packet retransmission and reduce performance of all connections [8]. Although Reordering-Robust TCP (RR-TCP) is proposed to prevent this phenomenon [9], it is impossible to recover it when the volume of packet loss is larger than a certain level.

In Transport layer, they have congestion window for each path. It enables more effective transport by doing packet distribution and retransmission for each path [10]. However, the system has to be installed into each operation system in all the end-end way.

An implementation on Application layer does not demand to replace current operating systems [11]. However, there are variety of applications and it is difficult to implement aggregation method for all of them. After connections established, we have to consider how to distribute packets for each connection.

B. Packet Loss Problem in Bandwitdth Aggregation on Network Layer

If multiple interfaces are used for concurrent communications, there are possibilities that receiving node may take packets incorrect order. In such a case, receiver recognizes occurring of packet loss incorrectly due to receiving packets different from expected order of packets. Then TCP requests retransmission unnecessarily. This is one of problems in Bandwidth Aggregation on Network Layer. Although incorrect judgement of packet loss is not only the problem of Bandwidth Aggregation, the problem becomes complicated because multiple packets are delivered through different routes.

For the purpose of eliminating this problem, Earliest Delivery Path First (EDPF) was proposed [6]. EDPF is implemented to the node that delivers packets to different paths. EDPF chooses on which path each packet should be sent in consideration of their bandwidth, delay and congestion. EDPF decides the fastest path to transmit the packet to receiver node. All packets are sent through the route on which estimated time is the shortest. Therefore, receiver can receive any packets in correct order. It makes Bandwidth Aggregation effective as estimated efficiency in no packet loss circumstances, and its effectiveness has been verified by previous researches.

C. Performance Problem in Bandwidth Aggregation on Network Layer

In the case of wireless communication, there are so many packet losses more than the case of wired communication. When Bandwidth Aggregation is operating on Network layer or lower layer, TCP cannot recognize which path causes the packet loss. Thus, TCP executes congestion control and throughput is degraded more than necessary. This is the second problem in Bandwidth Aggregation on Network Layer.

Packet-Pair based Earliest-Delivery-Path-First algorithm for TCP applications (PET) and Buffer Management Policy (BMP) were proposed for the purpose of fixing that problem on Network layer [8]. PET has functions estimating which path should be used more strictly and dynamically. BMP is implemented in receiver node, evaluates whether a received packet is needed to line up or caused packet loss. When BMP receives later sequence number packet, it informs packet loss was occurred for sure. Otherwise BMP delivers correct order packets to TCP.

With PET and BMP, more effective communication is realized compared with implemented EDPF, in particular, when packet losses occur. However, in circumstances with a lot of packet losses, even PET-BMP cannot execute efficient Bandwidth Aggregation. This is because TCP congestion window is reduced when a packet loss is considered to happen at one connection. As a result, throughput of all connections is degraded since TCP congestion window is shared among all connections in the case of Bandwidth Aggregation on Network layer.

This is one of the most difficult problems to solve in Bandwidth Aggregation on Network Layer. Referenced researches claim that it is possible to achieve expected results with eliminating packet losses using other methods [8]. In reality, it is too difficult to eliminate packet losses in wireless communication.

III. Our Proposal for Bandwidth Aggregation

As shown in previous chapters, we face various obstacles using Bandwidth Aggregation on Network layer and/or lower layer. Thus, we have proposed Middleware layer that aggregate bandwidth on the middle between Application layer and Transport layer. Figure 1 shows comparison between Bandwidth Aggregation on Network layer and our proposed model. In our model, all TCP connections are treated separately and aggregated them at the Middleware.

A. An Overview of Our Proposal

Our proposed model has different TCP connections per paths and aggregates their connections at the Middleware. Therefore, applications are not required to be conscious of aggregating bandwidth. It uses independent TCP congestion windows per paths, which prevent throughput degradation more than necessary, explained in the previous section, in the case of many packet losses.

This feature avoids the problems that happen in implementation on Network layer. If the bandwidth is aggregated on Network layer, TCP cannot determine on which path packets are lost, because TCP is upper layer and receives only after data is aggregated. Our previous research work shows their problems on Network layer are solved [12]. The defect which PET-BMP could not solve is overcome by our method, which means Bandwidth Aggregation on Middleware is more effective than that on the other layers.

This approach can also be implemented by modifying TCP which aggregates some connections on Transport layer. However, with the easier way with Middleware, we can use existing TCP for the purpose of achieving the most efficient Bandwidth Aggregation.

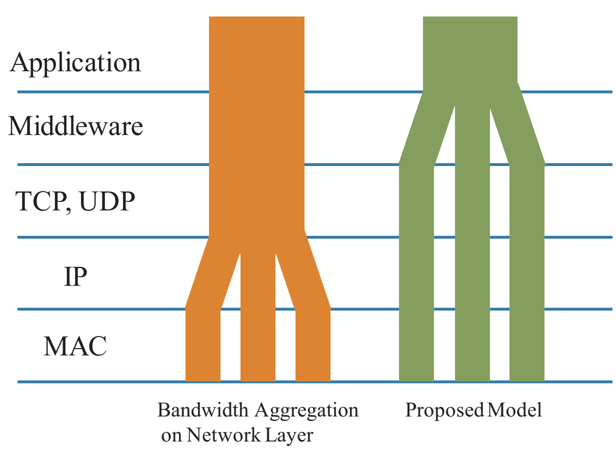


Figure 1. Comparison Between Bandwidth Aggregation on Network Layer and Our Proposed Model

B. The Design of Our Proposed Model

The sender Middleware establishes TCP connections on all possible paths. Packet of sent data is handed over from an application to Middleware, and a sequence number is given to a packet. A packet is sent out through enabled connection. EDPF for Middleware (Earliest Delivery Path First for Middleware) is used as an algorithm for the route selection when the packet is sent out.

When EDPF is used for the packet delivery, it calculates how long it takes from the sender to a receiver on each route, and it chooses the fastest path. For the estimation of the packet delivery, the bandwidth of wired and wireless route, delay time, and the congestion state is used. For the sake of next estimation, these parameters are updated with every packet delivery.

On the other hand, the receiver Middleware should put received packets in correct order and give them to an appropriate application. The receiver Middleware has a possibility that some packets arrive in incorrect order and needs to have a buffer to keep packets, for the purpose of waiting for the packet with expected sequence number. Thus, Middleware has a role to reorder the packets and hands over to the application afterward.

Estimation of required buffer size in each circumstances is one of the significant points for designing the Middleware. BMP also discusses about buffer size and controls how packets should be deriverd. We propose the method on other layer and suppose that they will behave differently.

IV. Evaluation with Simulation Software

In this experiments, we are motivated by the advantages that uses Bandwidth Aggregation through simultaneous use of multiple interfaces. We have used simulation software QualNet for the experiments [13].

For the purpose of designing Middleware, the buffer size of Middleware receiver has to be estimated clearly. We have investigated the size under various circumstances in Section V. The ratio of bandwidths between multiple wireless connection is changed and the required buffer size is evaluated at each case.

In Section VI, we have evaluated our proposed method. Since we have used EDPF for Middleware as an algorithm for routing in Bandwidth Aggregation, we have compared this method with a simple Weighted Round Robin (WRR) algorithm. Bandwidth Aggregation at Network layer, which is evaluated in existing literature, has been implemented and evaluated at first. That is to say, the existing method has been double-checked by an experiment. Next, our proposed method, Bandwidth Aggregation at Middleware, has been implemented and compared with the existing method in terms of performance stability when a packet loss occurs.

V. Evaluation of a Buffer Size of Receiver Middleware

In order to design Bandwidth Aggregation on Middleware, one of significant parameter value is a required buffer size of receiver Middleware. Therefore, we have evaluated the buffer size in various cases by changing the number of available routes, bandwidth, and delay time. With this experiment, we have determined the required buffer size of receiver Middleware.

For the evaluation, a simple WRR is used as a packet delivery algorithm, In WRR, a packet delivery route is determined depending on the ratio of wireless part bandwidths. For example, if the bandwiths of wireless part of each route are 200kbps, 100kbps, and 50kbps, respectively, the ratio of packet delivery to each route is 4:2:1.

A. Low Bit Rate Wireless Communications

As Scenario 1, the case with low bit rate wireless communications is evaluated. This is shown in Figure 2. Node 1 sends data to Node2, which has 2 wireless interfaces and receives data through 2 paths.

The bandwidth at wired connection is 10Mbps in this scenario. One of wireless connections is fixed to 100kbps and the other is varied from 100kbps to 800kbps. That is to say, the ratio of two bandwidths of wireless connections is varied from 1:1 to 1:8. Transport protocol used in this evaluation is TCP new Reno, and parameters are configured as Table I.

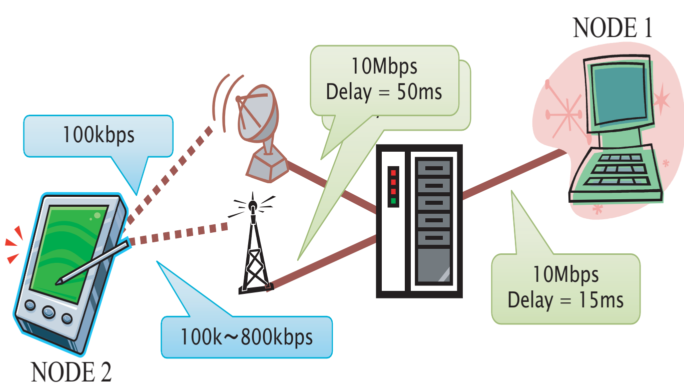


Figure 2. An Overview of Scenario 1

Table I
TCP PARAMETERS

MSS	1,460Bytes
Send buffer	65,535Bytes
Receive buffer	65,535Bytes

B. High Bit Rate Wireless Communications

As Scenario 2, the case with relatively high bit rate wireless communications is evaluated. This is shown in Figure 3. Node 1 sends data to Node2, which has 2 wireless interfaces and received data through 2 paths. The bandwidths of wireless connections as well as wired connections are different from the case of Scenario 1.

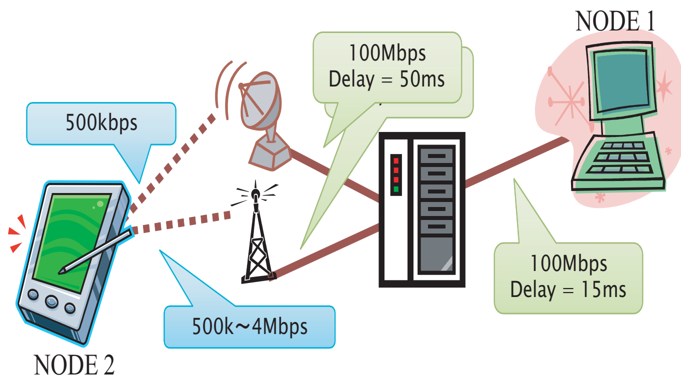


Figure 3. An Overview of Scenario 2

The bandwidth at wired connection is 100Mbps in this scenario. One of wireless connections is fixed to 500kbps and the other one is varied from 500kbps to 4Mbps. The ratio of two bandwidth of wireless connection is varied from 1:1 to 1:8, which is the same with the case of Scenario 1. Transport layer protocol and TCP parameters are also configured as the same with Scenario 1.

C. Evaluation in Various Cases

We have also evaluated two more other cases. In Scenario 3, as shown in Figure 4, while the bandwidths of wireless connections are the same with that of Scenario 2, that is, one of wireless connections is fixed to 500kbps and the other one is varied from 500kbps to 4Mbps, the bandwidth of wired connection is 50Mbps, the half size of that in Scenario 2.

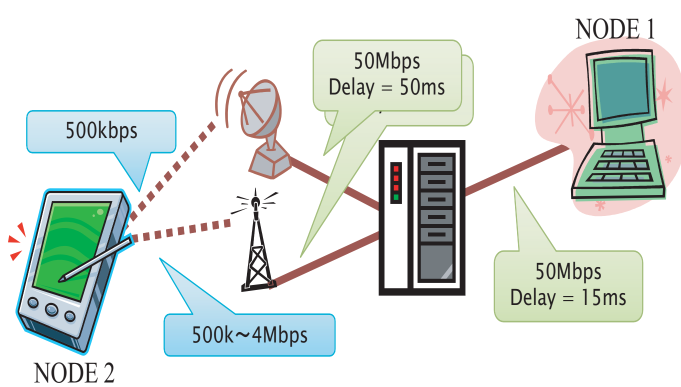


Figure 4. An Overview of Scenario 3

In Scenario 4, as shown in Figure 5, the bandwidth of wired connection is 50Mbps, which is the same with that of Scenario 3. However, the bandwidths of wireless connections are the half size of those in Scenario 3, that is, one of wireless connections is fixed to 250kbps and the other one is varied from 250kbps to 2Mbps.

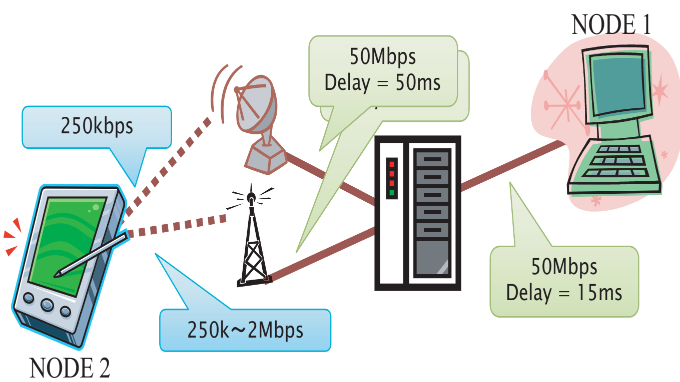


Figure 5. An Overview of Scenario 4

In both scenarios, the ratio of two bandwidth of wireless connection is varied from 1:1 to 1:8, which is the same with the previous scenarios. Transport layer protocol and TCP parameters are also configured as the same with previous Scenarios.

We have evaluated the required buffer size of receiver Middleware using these models. The experimental results are shown in the following subsections.

D. Period of Steady State and Unsteady State

Figure 6 shows throughputs of two connections and buffer size of receiver Middleware when bandwidths of wireless connections are set to 100kbps and 300kbps in Scenario 1.

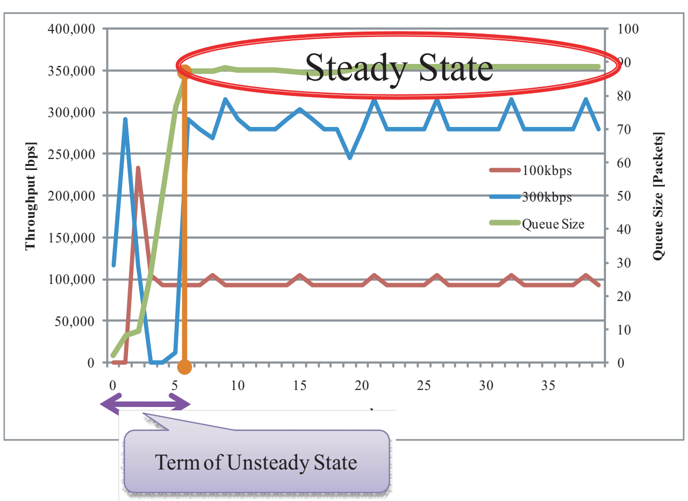


Figure 6. Throughputs and Queue Size

At beginning, the communication is a little unstable for a while. After a short period, two of wireless connections' throughput show stable and efficient communication.

Queue size of receiver Middleware is growing at first and becomes stable at a value. We call the period that buffer size is stable "Steady State", and the time until being stable "Term of Unsteady State". We focus on their values at various circumstances.

E. Association Between Ratio of Bandwidths and Required Buffer Size

Figure 7 shows required buffer size for Middleware of reciever at the period of Steady State in Scenario 1 when ratio of two bandwidths is changed from 1:1 to 1:8.

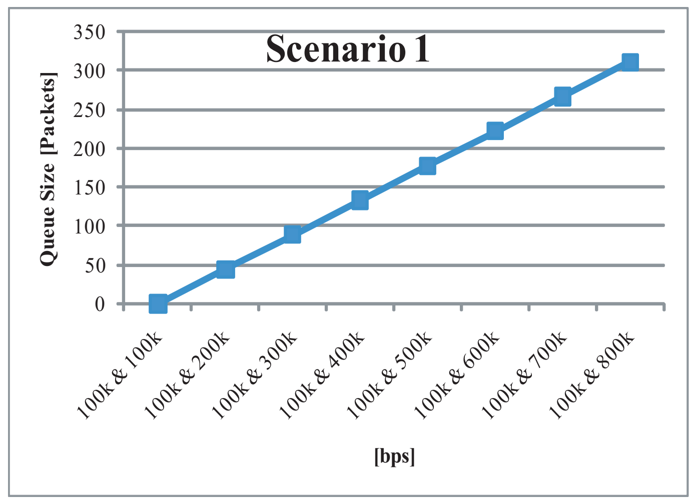


Figure 7. Buffer Size in Scenario 1

The value of buffer size when two interfaces have the same bandwidths is zero. That is to say, almost no buffer is required when two wireless connections have the same bandwidth.

On the other hand, when two interfaces have different bandwidths, the required buffer size is proportional to the ratio of one interface's bandwidth to the other.

Figure 8 shows the required buffer size at the period of Steady State in Scenario 2 when ratio of two bandwidths is changed.

The value of required buffer size is proportional to the ratio of one interface's bandwidth to the other, as is the same with Scenario 1. Although Scenario 1 and Scenario 2 have different bandwidth and different ratio of bandwidth between wired and wireless connections, buffer size is only

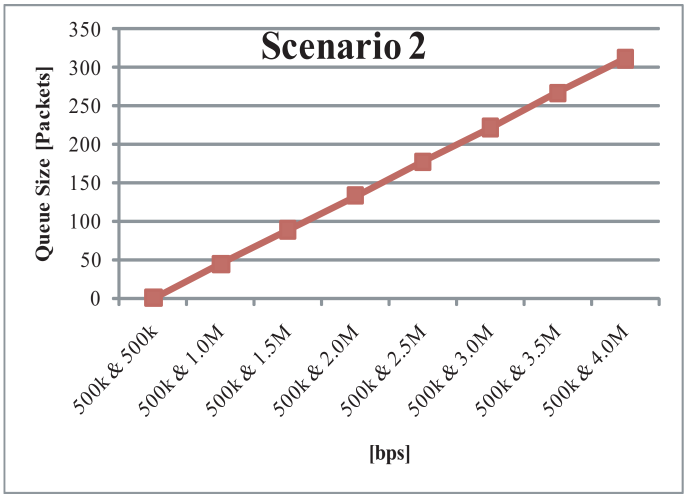


Figure 8. Buffer Size in Scenario 2

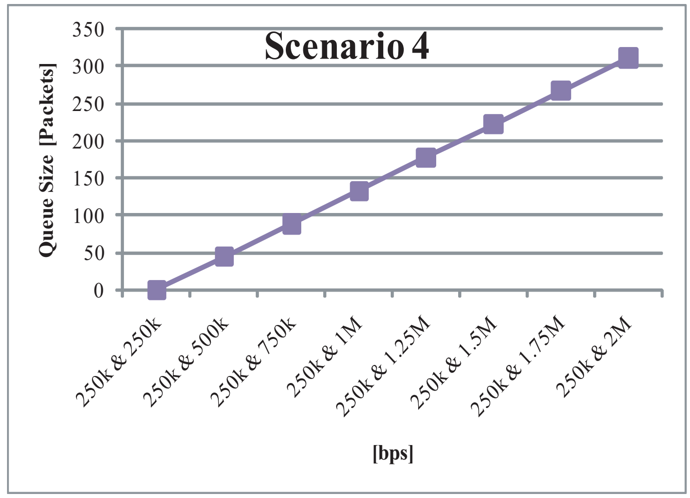


Figure 10. Buffer Size in Scenario 4

determined by the ratio of two bandwidths of wireless connections.

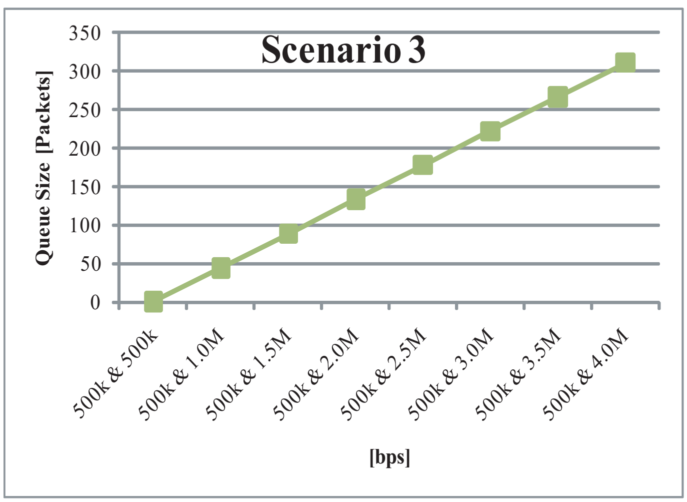


Figure 9. Buffer Size in Scenario 3

In addition, the required buffer sizes at the period of Steady State in Scenario 3 and 4 are shown in Figure 9 and 10, respectively. In these figures also, the value of required buffer size is proportional to the ratio of one interfaces' bandwidth to the other.

In all scenarios, the absolute value of the buffer size is completely the same when the ratio of two interfaces' bandwidths is the same. This is one of the most significant parameters to design Middleware, and it is possible to determine the value like this. It is interesting to see that required buffer size can be determined only by the ratio of bandwidths of two wireless interfaces, regardless of their absolute values.

F. Association Between Ratio of Bandwidths and Period of Unsteady State

Figure 11 shows period of Unsteady State, the period until throughput becomes stable from the beginning, when the ratio of two wireless connections' bandwidths is changed. All cases of Scenario 1 to 4 are shown in this figure.

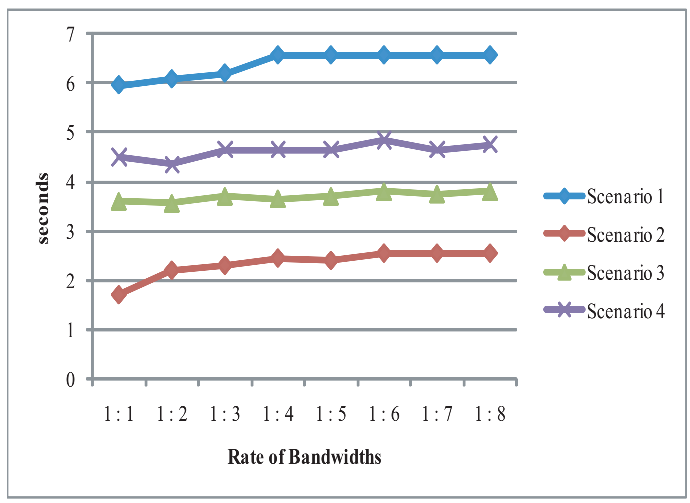


Figure 11. Period of Unsteady State

The period of Unsteady State in Scenario 2, whose connections have higher bit rate, is shorter than that of Scenario 1, whose connections are lower bit rate, in all cases. The ratio of bandwidths between two wireless connections does not affect their length.

VI. Comparison between Bandwidth Aggregation on Network Layer and Our Proposed Method

In this section, Bandwidth Aggregation on Network layer and that on Middleware are evaluated. In order to evaluate the effectiveness of EDPF for Middleware as an algorithm for routing used in Bandwidth Aggregation, this is compared with the simpler method, Weighted Round Robin (WRR).

First, the evaluation result of Bandwidth Aggregation on Network layer has been double-checked by an experiment. Next, performance of Bandwidth Aggregation on Middleware, EDPF for Middleware proposed in this paper, has been evaluated and compared with that of Bandwidth Aggregation on Network layer.

A. An Overview of Experiment

In this evaluation, as shown in Figure 12, a mobile terminal that has three wireless connections (Node 2) receives data sent from Node 1, using three routes simultaneously.

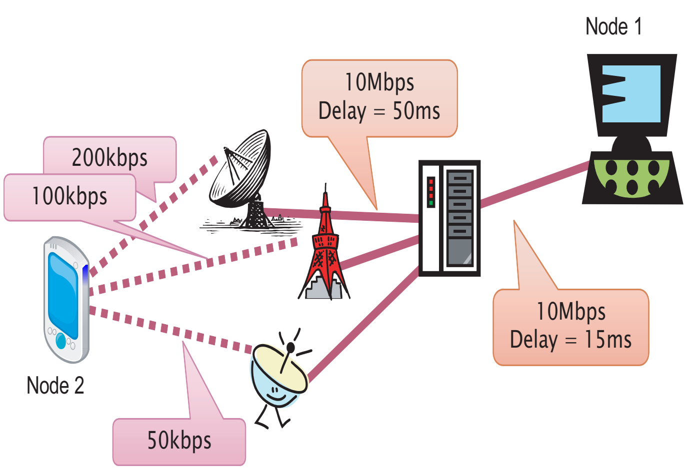


Figure 12. Scenario of Experiment

The bandwidth of wired connection, presented in a real line, is 10Mbps and the delay time is presented in the figure. The bandwidths of wireless connections are 200kbps, 100kbps, and 50kbps, respectively. Transport layer protocol used in this experiment is TCP new Reno, and TCP parameters are set as shown in Table II.

Table II
TCP Parameter

MSS	1,460Bytes
Send buffer	65,535Bytes
Receive buffer	65,535Bytes

B. Experiment with No Packet Loss

For Bandwidth Aggregation on Network layer and on Middleware, EDPF and WRR are used as routing algorithm for data delivery. In the first case, throughput and required buffer size have been evaluated with no packet loss environment in Figure 12.

1) Evaluation Result of Bandwidth Aggregation on Network Layer: Throughput when two routing algorithms are used, EDPF and WRR, with Bandwidth Aggregation on Network layer is compared in Table III. In this table, the number of Duplicated ACK and the number of retransmission in both cases are also indicated.

Table III
Experimental Result

Algorithm	Thr(kbps)	Dup ACKs	Retransmitted
EDPF	329	0	0
WRR	265	522	96

In this experiment, the sum of bandwidths of three routes is 350kbps. Therefore, while performance of WRR is 75.7% of total bandwidth, 94.0% in the case of EDPF.

This is because no Duplicated ACK occurs and no retransmission of packet is observed in the case of EDPF as shown in Table III. Thus the performance of EDPF is higher.

2) Evaluation Result of Bandwidth Aggregation on Middleware: Throughput when EDPF for Middleware is used for Bandwidth Aggregation on Middleware is shown in Figure 13. Throughput of each route is shown in this figure.

For the comparison, throughput when WRR is used for Bandwidth Aggregation on Middleware is shown in Figure 14.

Compared with the simpler WRR, EDPF for Middleware, which delievers data based on the situation of each route, achieves stable communication at each moment. That is to say, EDPF for Middleware has succeeded in stabilizing the behavior of multiple connections.

The volume of data buffered in receiver Middleware, using EDPF and WRR, is shown in Figure 15.

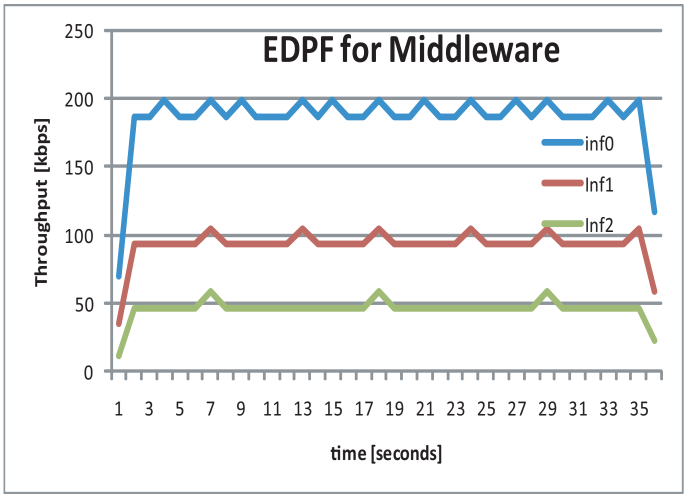


Figure 13. Throughput of EDPF for Middleware

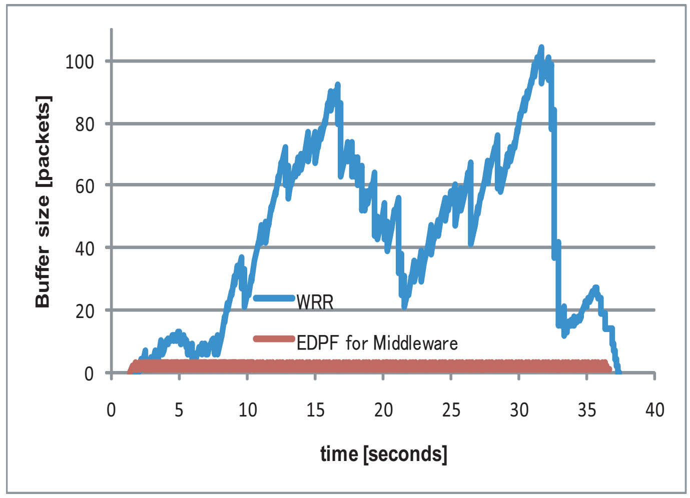


Figure 15. Buffer Size of Receiver Middleware

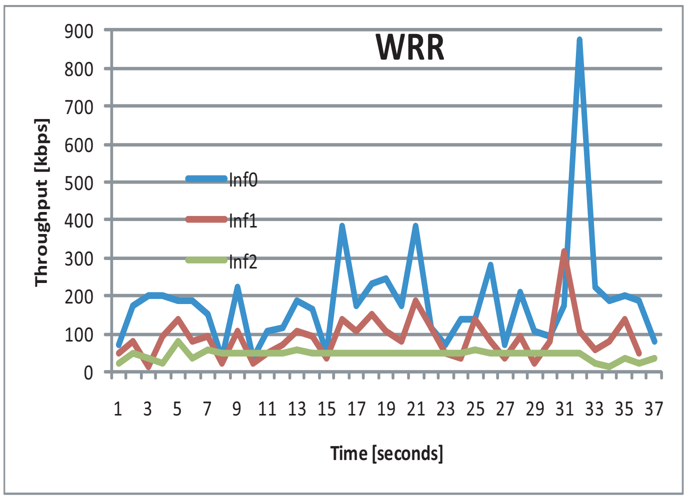


Figure 14. Throughput of WRR

In the case of using EDPF, the maximum number of packets stored in the buffer of receiver Middleware is only three, while the maximum number of packets is about 100 when WRR is used. Therefore, throughput of WRR is not stable as shown in Figure 14, and delay becomes large from a sender application to a receiver as a result.

C. Experiment with Packet Loss

In Bandwidth Aggregation on Network layer and that on Middleware, using EDPF as an algorithm of routing, the case with one packet loss 18 seconds after the beginning of the scenario is evaluated by observing throughput at each route.

1) Evaluation Result of Bandwidth Aggregation on Network Layer: Throughput of Bandwidth Aggregation on Network layer with packet loss is shown in Figure 16. The packet loss has occurred on one of three connections.

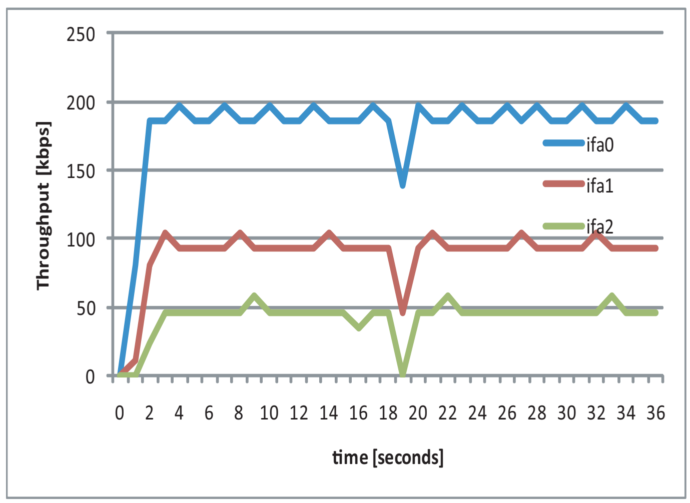


Figure 16. Throughput of Bandwidth Aggregation on Network Layer with Packet Loss

It is observed that throughput of all interfaces is reduced when the packet loss occurs, which is 18 seconds after the beginning of this scenario. In this case, packet loss at one connection affects other connections, and thus, total throughput is reduced. This is because TCP cannot decide on which connection the packet loss occurs, and congestion window of all connections should be decreased as a result. According to this evaluation results, it is impossible to achieve high performance with Bandwidth Aggregation on Network layer in the case of packet loss.

2) Evaluation Result of Bandwidth Aggregation on Middleware: Figure 17 shows the throughput of Bandwidth Aggregation on Middleware, using EDPF algorithm for packet routing, with packet loss on one of three connections.

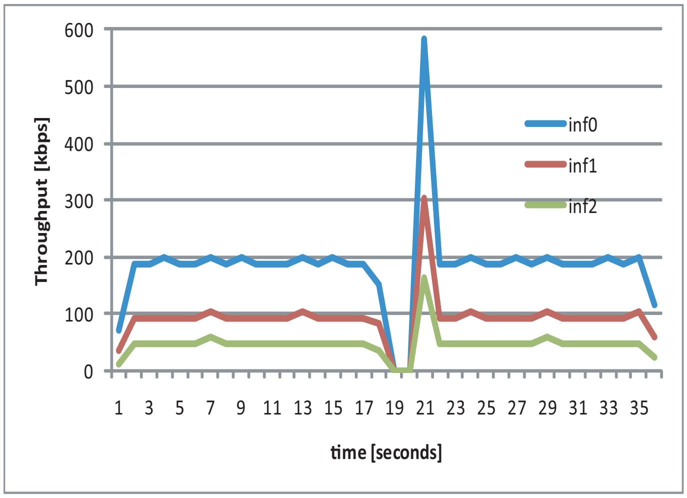


Figure 17. Throughput of Bandwidth Aggregation on Middleware with Packet Loss

In this case, the number of packet stored at buffer of receiver Middleware is shown Figure 18.

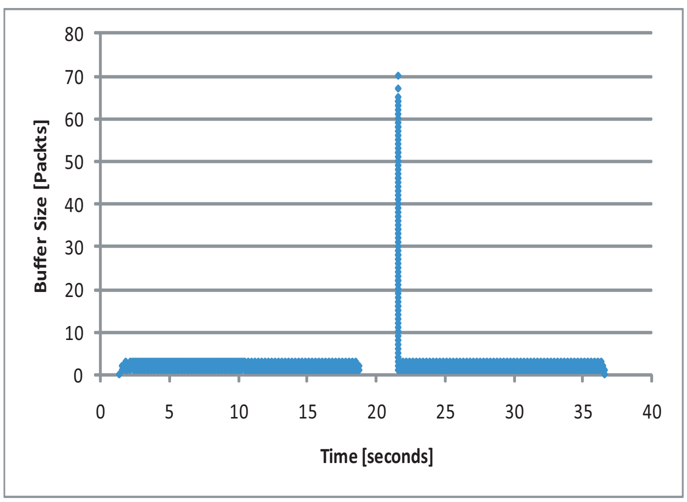


Figure 18. Buffer Size of receiver middleware

According to Figure 17, it seems that a single packet loss happened in one route reduces throughput of all routes. This seems something similar with the case of Bandwidth Aggregation on Network layer.

However, this figure shows the volume of data that can be passed from Middleware to an application. As shown in Figure 18, the volume of data stored in the buffer of receiver Middleware becomes large in order to wait for a retransmission of the lost packet. This causes performance degradation of all routes only temporarily and it is recovered afterwards as shown in Figure 17. That is to say, this is different from the phenomenon of throughput reduction in the case of Bandwidth Aggregation on Network layer. In this case, congestion windows are not decreased and data transfer continues during that period, and the data transfered during that period is stored in the buffer.

VII. Conclusion and Future Work

In this paper, we have experimented with network simulator for the purpose of evaluation of the communication using multiple interfaces simultaneously. The methods of Bandwidth Aggregation on network layer still have problems, for instance, because they can not recognize which path causes the packet loss. We have proposed the model of Bandwidth Aggregation on Middleware in order to eliminate the problem. The effect are verified compared with previous method since we can get comparable throughput as well as aggregating throughput of multiple connection.

The receiver Middleware needs to have buffer to restore the order of packets' sequence number. We have investigated how large buffer is needed in various situations. The mobile node which has two interfaces varies one of interface's bandwidth and observes the buffer size. The result shows it proportional to the ratio of one interface's bandwidth to other one.

As a routing algorithm for Bandwidth Aggregation, EDPF for Middleware and simple WRR are compared. As a result, EDPF for Middleware achieves stable and efficient performance of communications. In addition, experiment of a case with packet loss is performed. According to the evaluation results, Bandwidth Aggregation on Middleware is able to perform superior communications compared with Bandwidth Aggregation on Network layer.

In the future, we will implement the feature of buffer size that demonstrated by the experiments and function on the sender Middleware considering how to distribute each packets to the paths. Moreover, we will suppose that mobile node can have many wireless interfaces and study the result in such cases. In addition, we try to achieve more efficient Bandwidth Aggregation in a various situations, for instance, various pattern of lower layer and dynamically-changed bandwidth.

Acknowledgement

We would like to thank to Dr. Onur Altintas in TOYOTA InfoTechnology Center, Co. for the conscientious advice and help for this research work.

References

[1] E. Miyazaki and M. Oguchi : "Evaluation of Buffer Size for Middleware using Multiple Interfaces in Wireless Communication," Tenth International Conference on Networks (ICN2011), pp.202-205, January 2011.

[2] M. Stemm and R. Katz : "Vertical handoffs in wireless overlay networks," Mobile Networks and Applications Vol.3, No4, pp.335-350, January 1998.

[3] OpenFlow : http://www.openflowswitch.org/

[4] NEC Corp. : http://www.nec.co.jp/press/ja/1002/0402.html, February 2010.

[5] IEEE P802.3ad Link Aggregation Task Force : http://grouper.ieee.org/groups/802/3/ad/

[6] K. Chebrolu and B. Raman : "Bandwidth Aggregation for Real-Time Applications in Heterogeneous Wireless Networks," IEEE Transactions on Mobile Computing, Vol.5, No4, pp.388-403, April 2006.

[7] K. Koyama, Y Ito, H. MINENO and S. Ishihara: "Evaluation of Performance of TCP on Mobile IP SHAKE," Transactions of Information Processing Society of Japan, 2004.

[8] K. Chebrolu, B. Raman, and R.R. Rao : "A Network Layer Approach to Enable TCP over Multiple Interfaces," Journal of Wireless Networks (WINET), Vol.11, No5, pp.637-650, September 2005.

[9] M. Zhang, B. Karp, S. Floyd and L. Peterson: "RR-TCP: A Reordering-Robust TCP with DSACK," IEEE International Conference on Network Protocols, 2003.

[10] M. Zhang, J. Lai, A. Krishnamurthy, L. Peterson and R. Wang: "A transport layer approach for improving end-to-end performance and robustness using redundant paths," USENIX 2004 Annual Technical Conference, pages 99-112, 2004.

[11] H. Nozawa, N. Honda, K. Sakakibara, J. Nakazawa, and H. Tokuda: ARMS: "Application-level Concurrent Multipath Utilization on Reliable Communication," Internet Conference 2008, October 2008.

[12] E. Miyazaki, O. Altintas, and M. Oguchi : "A Study of Bandwidth Aggregation Using Multiple Interfaces on Middleware Layer," DICOMO 2010, July 2010.

[13] Scalable Network Technologies : http://www.scalable-networks.com/

Opportunistic Sensing in Train Safety Systems

Hans Scholten and Pascal Bakker
Pervasive Systems
University of Twente
Enschede, the Netherlands
hans.scholten@utwente.nl, aboe@aboe.nl

***Abstract*—Train safety systems are complex and expensive, and changing them requires huge investments. Changes are evolutionary and small. Current developments, like faster - high speed - trains and a higher train density on the railway network, have initiated research on safety systems that can cope with the new requirements. This paper presents a novel approach for a safety subsystem that checks the composition of a train, based on opportunistic sensing with a wireless sensor network. Opportunistic sensing systems consist of changing constellations sensors that, for a limited amount of time, work together to achieve a common goal. Such constellations are self-organizing and come into being spontaneously. The proposed opportunistic sensing system selects a subset of sensor nodes from a larger set based on a common context. We show that it is possible to use a wireless sensor network to make a distinction between carriages from different trains. The common context is acceleration, which is used to select the subset of carriages that belong to the same train out of all the carriages from several trains in close proximity. Simulations based on a realistic set of sensor data show that the method is valid, but that the algorithm is too complex for implementation on simple wireless sensor nodes. Downscaling the algorithm reduces the number of processor execution cycles as well as memory usage, and makes it suitable for implementation on a wireless sensor node with acceptable loss of precision. Actual implementation on wireless sensor nodes confirms the results obtained with the simulations.**

***Keywords*-opportunistic sensing; wireless sensor network; context awareness; activity recognition; train safety.**

I. Introduction

In this paper, we show how opportunistic sensors, which use a common pattern of movement, are deployed to detect the composition of a train. The problem was introduced in [1] and [2] describing how wireless sensor networks enhance safety in moving linear structures such as trains. The wireless sensor network is a subsystem of a railway safety system to monitor the initial composition of a train and to detect changes in composition once the initial composition has been established. Movement as a discriminating factor for context awareness in wireless sensor networks has been described earlier [3] [4], but not for trains.

A. Railway Safety Systems

Currently, the rail system in Europe consists of multiple different safety systems: the countries still rely mostly on a country-specific system. International European trains have multiple systems onboard to be able to pass borders and enter another country and hence another safety system. Projects in different European countries are working on a uniform system [5], the European Rail Train Management System (ERTMS), under supervision of the European Railway Agency [6]. The eventual goal is the adoption of the system in all participating countries. The draft for the latest version of ERTMS is known as ERTMS level 3. ERTMS level 1 and level 2 still consider track sections instead of trains as sections. Like the old system, a track is divided in fixed length sections. When a train enters a section, called a block, no other train is allowed in. The length of a block is determined on the worst case, considering length, weight and maximal speed of trains. This implies that a small light train takes as much space as a large and heavy train. ERTMS level 3 considers a train as a moving block, keeping a safety zone around the train in accordance with its length, weight and speed.

By representing trains as moving blocks, short and light trains will form smaller blocks than heavier trains, which have a much longer brake path. This allows more trains on the same track when compared to a system that uses static safety zones. Level 1 and level 2 ERTMS systems use a straightforward approach to detect whether a block is occupied. When a train enters or leaves a block, its axles are counted. When the number of axles is equal at both counts, the block is empty. If not, (part of) the train still occupies the block. ERTMS level 1 uses signals located at the side of track for the indication of the availability of the next zone, whereas ERTMS level 2 communicates this information by radio using GSM-Rail (GSM-R). ERTMS level 3, in contrast to levels 1 and 2, does not rely on a trackside signaling system. Instead, it uses an onboard safety system that checks and controls the safety zone of the train. It is imperative that a train can guarantee its integrity, i.e. its composition is known and no carriages are lost. As no trackside backup system is available, a dangerous situation occurs when a train loses a carriage. The next train may crash into the lost carriage if it is not informed in time. A system that accurately monitors in real time the integrity of the composition is a crucial part of ERTMS level 3. Leaving a carriage behind is not as farfetched as it sounds. It happens all the time, for example, in switchyards or at stations. In

Figure 1. Freight carriage connection

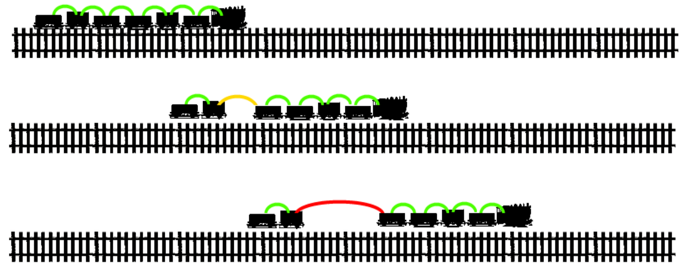


Figure 2. Beacons, single train

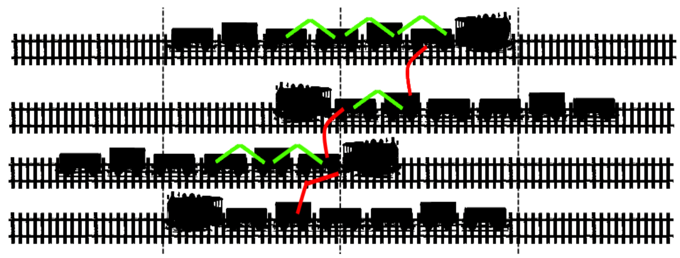


Figure 3. Beacons, multiple trains

any case, the train must know its composition and report changes. Passenger trains are equipped with numerous wired links between wagons, which can be used to monitor its carriages.

Freight trains, however, are normally not equipped with electrical connections between carriages (see Figure 1). When a train rides at night in the Netherlands, the engineer manually puts up a light at the last carriage, since the carriages lack all electrical provisions.

A system for guaranteeing freight train integrity should preferably operate wirelessly. Because carriages are scheduled for maintenance once or twice every year, the new system should be able to run for at least a year without human intervention. The system has to operate in all kinds of weather and should be able to withstand dust, water and other kinds of abuse.

B. Opportunistic Sensing

"Opportunistic sensing is seen as a way to gather information about the physical world in the absence of a stable and permanent networking infrastructure." (Opportunity Workshop at Ubicomp 2010, Copenhagen, Denmark). The absence of a stable and permanent networking infrastructure dictates that collected information is either processed and acted upon inside the network by opportunistic collections or clusters of nodes [7], or the information is preprocessed and stored inside the network until there is an opportunity to forward it outside the network, as is the case in delay tolerant networks [8] - [18].

Opportunistic sensing and opportunistic networking is often associated with human-centric ubiquitous systems, such as in crowd sourcing and participatory sensing applications [19] - [21] or are focusing on human activity recognition [22] - [24].

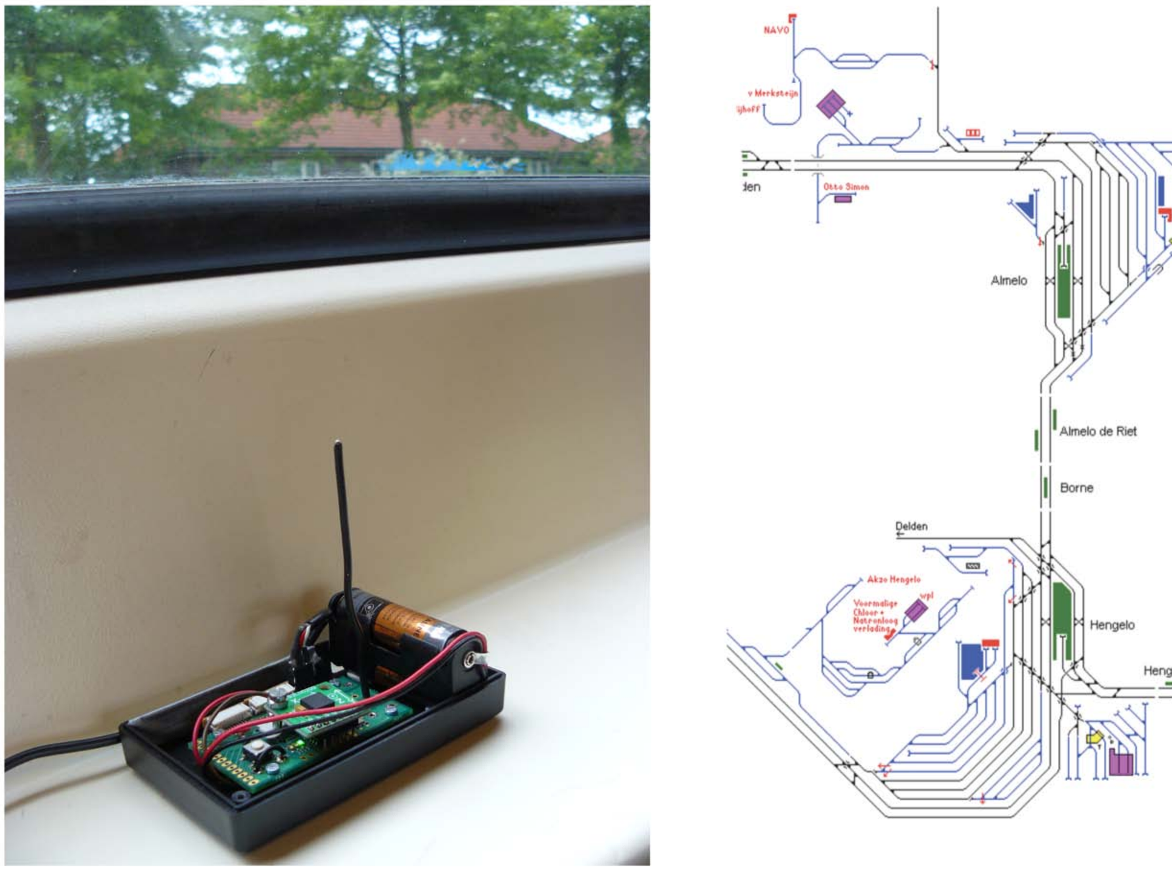
C. Opportunistic Sensing for Train Safety Systems

The proposed method of using opportunistic sensing with movement as discriminating factor seems overkill, where a simple detection system based on radio beacons would be sufficient. Figure 2 illustrates the principle. Once the decoupled carriages at the back of the train are out of radio range, an alarm will be raised. The system works for a single train with no other trains in range. This is not the case at stations or switchyards, or, generally, when a train passes an other on an adjacent track and comes in range of the other train's beacons, as shown in Figure 3. The system would still work if the train had knowledge about its composition. Periodically checking all beacons in range would detect any change in the composition of the train. Because the ID of the beacons in sight are not known a priori, nor the mapping of IDs to carriages, nor carriages to trains, there is no way to discriminate carriages in different trains in close proximity. Additional measures are needed to discriminate between trains.

This paper investigates movement, or more precise acceleration, as a way to distinguish carriages in different trains and determine a train's composition. In the remainder of this paper, we will discuss the collection and analysis of the data sets to be used in the simulations, the simulation itself and the implementation on wireless sensor nodes respectively, followed by a discussion of the results.

II. Data Collection and Analysis

As explained in the introduction, movement is used as discriminating feature. However, motion is multi-dimensional and probably too complex to use in wireless sensor nodes in an energy efficient manner. Not only processing power and memory usage might be issues, also running complex algorithms for longer periods of time consumes large amount of energy, negatively influencing the operational time of the system. Because of all these limitations, it might seem advantageous to deploy a centralized solution with simple sensors and a server doing all the processing. Raw data is sent directly to the server, where the correlation of all carriages is calculated. While this saves on processing and



(a) Wireless sensor node

(b) Railway track

Figure 4. Data collection

memory on the sensor node, data sent by radio will be much higher than in the decentralized case, where processing takes place on the node. The increase in power consumption by the radios exceeds the energy savings due to less processing. Still, the sensors would depend on batteries for their energy provisioning. The result would be less operational time.

For the distributed, wireless sensor network version, several measures can be taken to enable implementation on nodes:

- Minimize the time the algorithm executes,
- Simplify the algorithm as much as possible, and
- Simplify the input data for the algorithm.

Under normal conditions, a train's composition will not change while underway and moving. If a change occurs under these circumstances, it will be accidentally. We have seen that a system with radio beacons works to detect these changes, but only under the condition that the composition of the train is known a priori. Such a system would be much simpler and more reliable than any implementation with accelerometers. It is also much more energy efficient than running complex algorithms on a wireless sensor node.

A train run has four distinct phases:

- The train is standing still,
- The train starts moving and accelerates,
- The train has a steady speed, and
- The train decelerates and stops.

During the first phase, standing still, nothing changes and sensor nodes sleep. When the second phase begins, the sensor nodes wake up and the train's composition is determined. Once the composition is known, radio beaconing detects any changes from the initial composition. At the end of phase four, the train has come to a complete standstill, the nodes go to sleep again to preserve energy. The algorithm that runs on the sensor nodes reflects the four phases of a train run: sleep, determine composition, detect changes from initial composition, and sleep.

In the following, we will focus on the most complex second phase, discriminating carriages in different trains and

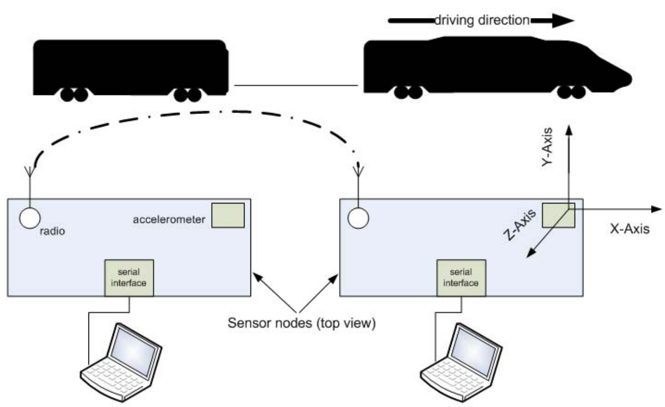


Figure 5. Collecting data sets

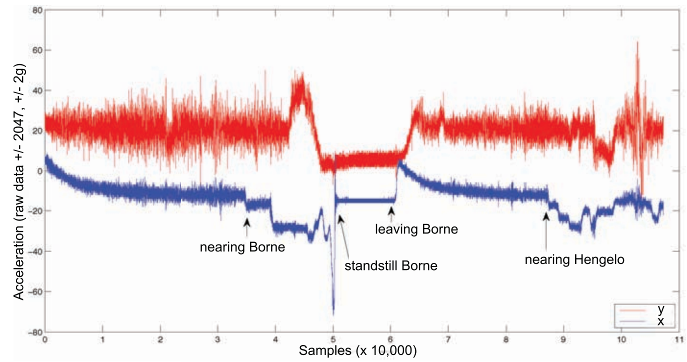


Figure 6. Accelerometer x- and y-axis

determination of a train's composition.

A. Collecting the Data

To check the feasibility of the proposed algorithm a number of simulations with realistic data sets are executed. The data sets consist of sensor data recorded on a track featuring multiple stops and curves (see Figure 4b), resulting in a wide variety of data. Figure 4a shows the wireless sensor node that is used to sample the data, consisting of an Ambient muNode 2.0 [25] provisioned with an STMicroelectronics LIS3LV02DQ accelerometer. The maximum sample rate of this sensor is 640 Hz, but the combination of hardware and software limits the sample rate to 160 Hz. The sensor nodes are aligned with the horizontal x-axis in the driving direction, the y-axis in the horizontal sideways direction and the z-axis in the vertical direction.

The data sets are recorded with two sensor nodes in two distinct carriages in the same train during several runs (see Figure 5). The data sets from sensors in different runs of the same train are used to simulate different trains. The sensor data from each sensor node is recorded real-time with laptops. All data are time stamped by the sensor nodes and the sensor nodes are connected via their radios to synchronize them.

B. Analyzing the Data

Figure 6 depicts raw data with a sampling frequency of 155 Hz of a journey between two stations with one station in between. The train is already moving when the graph begins. The two graphs are shifted up (y-axis) and down (x-axis) for a better overview.

The bottom graph shows the x-axis, the driving direction, and illustrates changes in speed of the train. After the initial acceleration (not shown) the speed settles to a constant value (acceleration is 0). Just before the middle station, the train decelerates in stages and comes to a standstill with a shock. Leaving the middle station the train accelerates to a constant speed.

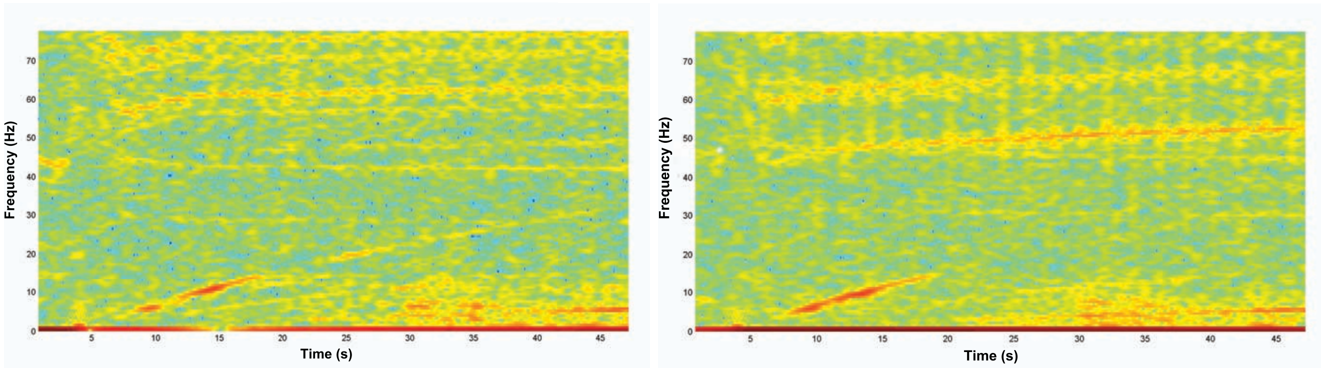
The top graph depicts the sensor's y-axis, movement sideways, and just before reaching the middle and the terminal station the train is crossing a switch changing tracks. In between the graph exposes short sideway movements of the train inside the track. Even when standing still, sideway movement is detected, though with a smaller amplitude than when the train is moving. This is contributed to noise the accelerometer generates. This noise is also present in the x and z direction and must be filtered out before the sensor data can be processed further.

Not shown in the graph is the sensor data of the z-axis. This data did show clearly when a train is moving or not. Further analysis learned that the data did not have enough features to make a distinction between trains when they start moving at the same time. In the remainder of this section we will focus on movement in the other two directions.

Figures 7 and 8 show frequency spectrum diagrams for sampled data from the x- and y-axes of the accelerometers in carriages over a time period from 0 till 47 seconds. Color is used to depict the intensity of frequencies in the sampled data. Going from low to high intensities, the colors blue, green, yellow and red are used.

Figure 7 shows the frequency spectrum for the x-axis data of two carriages in the same train starting to move. Only frequencies lower than the sampling frequency/2 are considered: 0 to 80 Hz. The spectra seem similar in the low frequencies, but quite different in the high frequencies. The latter can be contributed to the already mentioned noise of the accelerometers. The spectra of two carriages in two different trains on the same section of the railway track (not shown) differ in all frequencies, but the difference in the lower frequencies is smaller than in the higher ones. Closer inspection of the spectra and a preliminary correlation of sensor data in different frequency bands indicates that the most usefull discriminating features are present in a frequency band of 0 to 2Hz. Before the sensor data are further processes a high-off filter is applied with a cut-off frequency of 2 Hz.

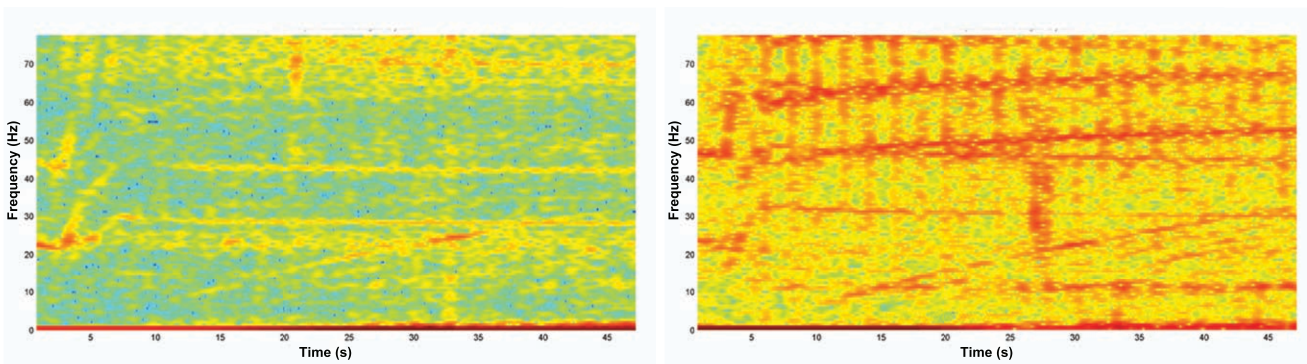
Figure 8 shows the spectrum for movement in the y



(a) Spectrum carriage 1, train 1, x-axis

(b) Spectrum carriage 2, train 1, x-axis

Figure 7. Frequency spectrum x-axis of two carriages in the same train



(a) Spectrum carriage 1, train 1, y-axis

(b) Spectrum carriage 2, train 1, y-axis

Figure 8. Frequency spectrum y-axis of two carriages in the same train

direction of two carriages in the same train. Though there are similarities in both spectra, they are much less than for the x direction. A complication is that the spectrum of a carriage in a different train (not shown) is similar to those shown.

Analysis of the collected movement samples shows that from all data in the three axis x, y and z, only the sensor data of the x-axis, which is the direction the train rides, are usefull. The data are highly polluted by noise, visible in the spectrum in the higher frequencies. The noise is generated by high frequency movements of the train and by the accelerometer and must be filtered before the data can be used. We also found that all discriminating features are in a frequency band of 0 to 2Hz, the relatively slow movements of the train.

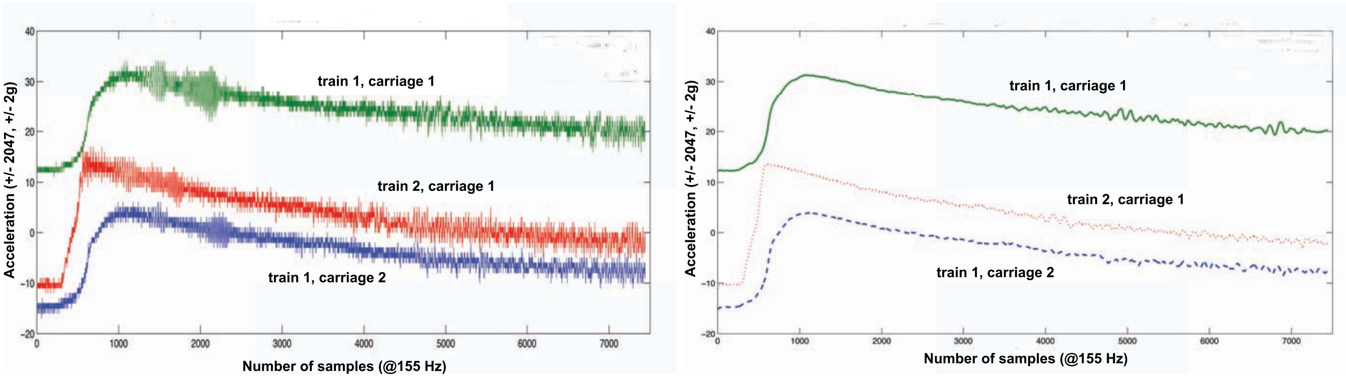
C. Preprocessing the Data

To filter the data, a second order Butterworth low pass filter is used. This choice is made because this type of filter is tested and runs on the used wireless sensor nodes. Figure 9 shows the results for the x-axis data samples of three carriages. The top and the bottom graph are from carriages in the same train. The graph in the middle is from a carriage in a different train. The graphs are synchronized, i.e., they are shifted in time so they show the trains starting at exactly the same time. In practise it will rarely happen that two trains in communication range will accelerate at exactly the same time. This is even discouraged, as it will cause peaks of electricity consumption in the power grid. Their is a clear difference between the two trains. The graph of the second train is steeper as a result of a higher acceleration.

The data sampling rate in the original data set as shown in the graphs is 155 samples per second. Because the data is filtered at 2 Hz, this high sampling rate is clearly overkill and can be reduced significantly in the final implementation. A frequency of 35 samples per second gives the same results as before. We did not test lower sampling rates, although this might have been better to reduce the CPU and memory consumption.

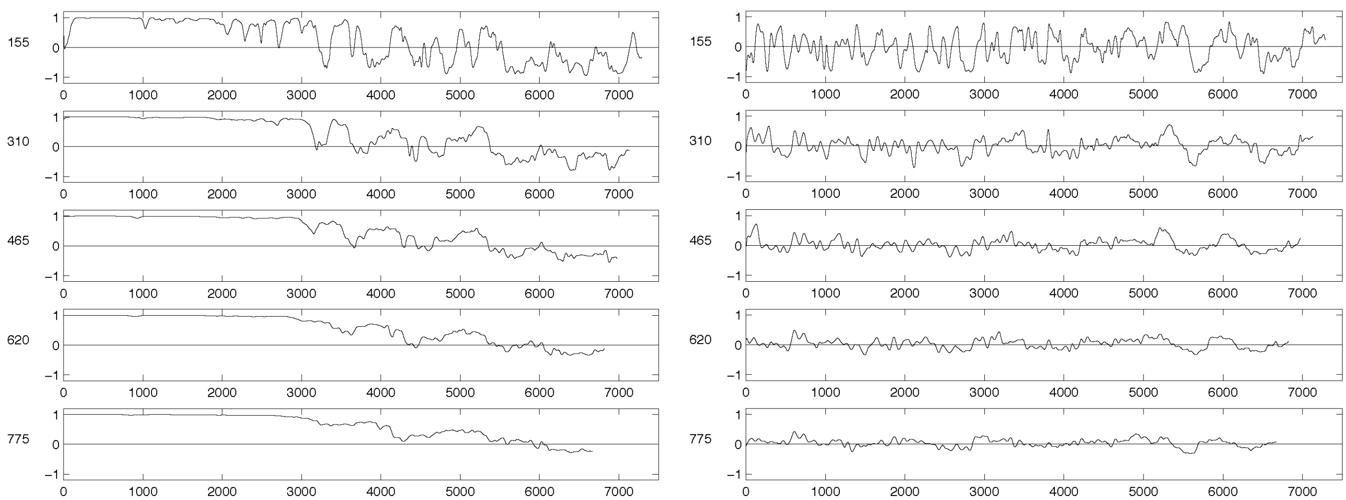
D. Data Correlation

The last step in the algorithm is to check whether two carriages share the same context by way of correlating the data. The correlation between two nodes is done by



(a) Unfiltered data (b) Filtered data

Figure 9. X-axis acceleration data from three carriages



(a) two carriages in the same train (b) two carriages in different trains

Figure 10. Correlation of two carriages with varying window sizes

using the Pearson product-moment correlation coefficient. The Pearson correlation is +1 if there is a perfect positive linear relationship, and -1 if there is a perfect negative linear relationship. If the value is 0 there is no relationship. The closer the coefficient is to +1 or -1, the higher the relationship between the nodes. The correlation process uses a sliding window over which the data are compared. A wider window normally leads to a more precise result, but also takes longer to produce this result. A smaller window gives a better reaction time, but the result is less reliable. So a trade-off has to be made.

Figure 10a gives the correlation results for two carriages in the same train with window sizes varying from 155 to 775 samples, corresponding with 1 to 5 seconds. The measured time frame is just over 7000 samples, or 45 seconds. Remember that the detection of the composition of the train only takes place during the first seconds after the train starts moving. Once the composition is known, it suffices to periodically test the presence of the initially found carriages by means of the radio beacons. The composition must be known within the first seconds of a ride. Therefore, the correlation results after 2000 samples, or approximately 13 seconds will be ignored. Figure 10a shows that over the time frame of the first 2000 sample a window size of 155 samples, or 1 second, is enough. The Pearson correlation coefficient is already very close to +1. However, the algorithm must also avoid false positives: carriages in other trains that are seen as carriages of the same train. Thus, a carriage in a different train has to be positively identified as such. This translates in a correlation coefficient of 0 for two carriages in different trains. Figure 10b shows that a window size of 155 samples is not enough. The correlation coefficient alternates between positive and negative values and is nowhere near a constant value of 0. To distingish two carriages in different trains a minimum window size of 620, or better 775 samples is needed.

To find the composition of a train, the correlation algorithm not only has to identify carriages in the same train, but also, to avoid false positives, identify carriages in other trains. Identifying carriages in the same train is very fast and only takes 1 second. However, identifying carriages in other trains takes 5 seconds.

III. Implementation

In the following, some implementation details will be discussed. All previous simulations are done with Matlab [26]. They show that identifying carriages in the same or different trains is feasible. They do not show that the algorithms to identify carriages will execute on a wireless sensor node. Matlab runs on powerfull computers with sufficient resources, while wireless sensor nodes are resource lean. Before the algorithms are suitable to run on the nodes they must be optimized, taking into account limitations such as CPU power, CPU speed, available memory and energy consumption.

A. Optimizations

The Matlab simulation takes an approach where everything is centralized. It assumes that the identification algorithm executes in one central place and that all data are available when needed. In reality this is not the case, the algorithm and data are distributed over the wireless nodes. Every carriage only has its own data, but to correlate its movement with that of its neighbors, it needs their movement data as well. It will be clear that the correlation of two neighbors only needs to be calculated once. This calculation can take place at either one of the neighbors. This process is optimized by dynamically forming master/slave pairs. The slave sends its data to the master and the master calculates the correlation. When the calculation has finished, the master sends the correlation results back to the slave. For every master/slave pair the movement data are communicated once and the correlation calculations is performed once, thus reducing both bandwidth and execution load. Which of the neighbors is master and which one slave is not essential. After each calculation the neighbors turn role to balance the energy consumption in the nodes more evenly.

The next step is optimization of the correlation algorithm. In its original form, all data in the sliding time window are used in the calculation [27]. When the window moves, the oldest data is replaced by new data, while all other data in the middle of the window stay the same. Marin-Perianu et al. [4] propose an optimization to this correlation calculation algorithm. The proposed algorithm stores intermediate values that can be used in the next calculation. This reduces the amount of calculations necessary for the computation of the correlation coefficient in the next window at the cost of a slightly increased memory usage. One disadvantage is, when running on small devices, the accumulation of rounding errors. The wireless sensor nodes execute their calculations with a limited number of bits and thus with limited precision. Every time a previous intermediate result is used, its rounding error is added to the rounding error of the current calculation. In due course this rounding error accumulates in unacceptable error margins, giving the wrong results. This can be counteracted by periodically resetting the intermediate results and start with a ”fresh” sliding window. Because in our case the algorithm runs for a limited time - only the first 5 seconds-, the error is small and within acceptable margins. Comparing the algorithm as run on the node with the Matlab version, no differences were found in the time frame of 5 seconds the correlation takes.

One more change in the original Matlab routines must be made before they can be implemented. The Matlab versions of the high-off filter and correlation algorithm are based on calculations that use floating point numbers. Using floating-point calculations on the wireless sensor nodes would stress the CPU unacceptably. Fixed-point calculations are better, though at the cost of possible loss of precision. Errors in rounding results would accumulate and might lead to significant deviations from the desired results over time. Figure 11 shows the difference between (Matlab version) floating point and fixed point calculation for the correlation algorithm. The deviation starts to become evident 15 seconds. Since the correlation algorithm to determine the train composition takes place in the first 5 seconds, replacing floating point by fixed point calculations does not influence the end result. The same results are observed for the high-off filter.

B. Timing and Memory Usage

Running the composition algorithm is distributed over all carriages of the train. The carriages are split in master/slave pairs, where each pair calculates the correlation between the master and the slave. The actual correlation calculation is done by the master. Before this calculation begins, the input sensor data is filtered. Master and slave filter their own data. When ready filtering the data, the slave sends its filter output to the master, after which the master correlates. The execution time depends on the size of the sliding window size of the correlation algorithm. Figure 12 shows the execution times for master and slave for increasing window sizes from 35 to 175 samples. Because the final sample rate implemented on de wireless sensor nodes is 35 samples per second, this correspondents with window sizes from 1 to 5 seconds. For a window size of 5 seconds, the execution times for master and slave are 171.9 msec and 85.9 msec respectively. The routines are executed uninterrupted and do not show significant deviations. Some of the ”extra” time the slave has, is used to assemble the filter data into packages to be sent to the master. The number of calculations that can be executed is 1000 / 171.9 is 5.8 per second.

The amount of memory that is needed for running the algorithm on a wireless sensor node depends on the number of neighbors it sees. With every neighbor the node will

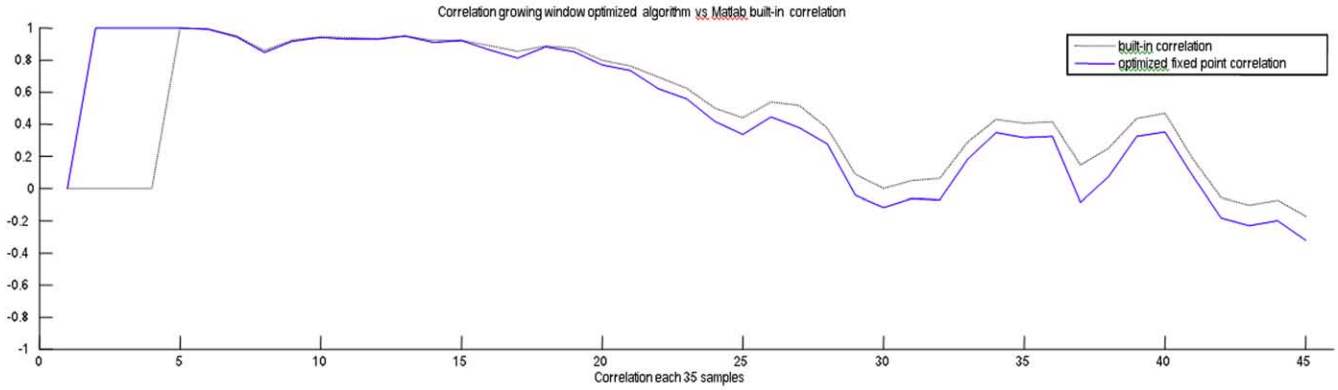


Figure 11. Comparison of floating point and fixed point calculations

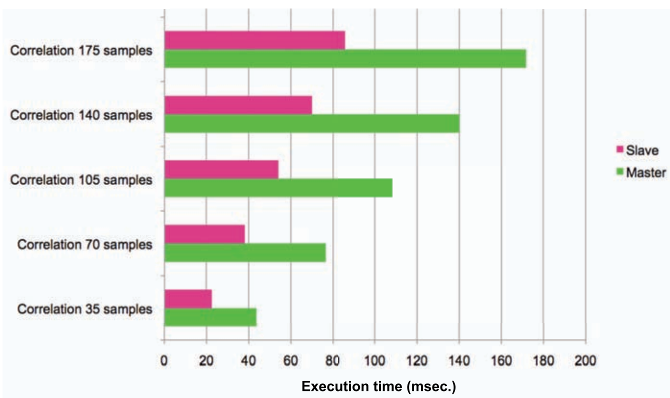


Figure 12. Execution times on a muNode with MSP430 microcontroller

Data structure	Size (bytes)
Dataset slave 175 samples	350
Timestamp start	2
Master mean data	4
Slave mean data	4
Master squares data	20
Slave squares data	20
Master sum data	20
Slave sum data	20
Sum products data	20
Total	460

Table I
MEMORY CONSUMPTION PER MASTER/SLAVE PAIR

for a master/slave pair. Table I summarizes the memory consumption per master/slave pair. Besides memory to store the sample data for one sliding window, the node stores intermediate results of the correlation calculation. The total amount of bytes is 460 per master/slave pair.

A standard Ambient Systems muNode 2.0 has 10kB of RAM available. For the normal operation of the node 2kB has been reserved, which leaves 8kB for the correlation algorithm. Given the memory consumption of 460 bytes for one master/slave pair and the availability of 8192 bytes, a node can store up to 16 master/slave pairs in memory.

IV. Conclusion

Train safety systems are going through a transition at the moment. Traditional safety systems depend on infrastructure-based sensors to detect the whereabouts of a train. These systems have serious drawbacks. It is a static infrastructure, where the track is divided in sectors, or blocks, of equal length. The length of a block is defined worst case, by the fasted, heaviest and longest train possible. As a result, the rail infrastructure is used far under its real capacity. This situation is complicated by the fact that European countries have their own safety system. Border crossing trains must have provisions on board for all safety systems on their route, which can accumulate to up to seven systems. The European countries have decided to develop a new system (ERTMS), where the safety system is not in the infrastructure, but on board the trains. One of the safety subsystems is responsible for the integrity of the train. It checks the composition of the trains and reports and changes of the initial composition.

In this paper, we have shown that such a subsystem can be implemented with a wireless sensor network. This network is deployed as an opportunistic sensor system that makes a selection out of a much larger set based on a common context. In this case, motion information is used to distinguish carriages belonging to different trains. The motion information consists of data obtained by accelerometers attached to individual train carriages. This data stream then was analyzed with Matlab on a PC to extract the best possible features to discriminate carriages. While sideways motion gives information on track changes and vertical movement indicates the quality of a track, the best information for our purpose is movement in the direction of travelling (x-axis). A spectrum analysis learns that not all frequency components in the sampling data are equally useful: only those below a frequency of around 2 Hz are significant to separate carriages. After filtering, the data from two different carriages are correlated with a correlation window of 5 seconds. We found that a smaller window of approximately 1 second suffices to find two carriages in the same train, but also leads to false positives for carriages in different trains. Extending the window to 5 seconds, no false positives were detected.

After the theoretical confirmation that motion information can be used for context awareness, the algorithm is implemented on wireless sensor nodes. However, the nodes used have several limitations. One limitation is the absence of floating point calculations. The filter and correlation routines have been rewritten, so only fixed point calculations are used. This might lead to errors due to accumulation of rounding successive results, but we showed that in the time frame the algorithms run this is not a problem.

The second limitation is power consumption. Filtering and correlation are so computing intensive that they cannot run over longer periods of time without exhausting the battery quickly. A first step is the reduction of the sampling rate from the original 160 Hz to 35 Hz. This is made possible because only the lower frequencies in the sampling data are significant. A lower sampling frequency would have been possible, but this does not substantially contribute to decreasing the processing load. Sampling data, filtering and performing one correlation per second takes 171.9 ms (worst case), which results in a duty cycle of around 17 percent. This exhausts the battery in a couple of hours, at most days, where 6 months is needed. The solution is found by executing the algorithm only for a period of 5 seconds from the moment the train starts moving after each stop. This is enough to establish the composition of the train and distinguish own carriages from those of different trains. During the ride, the initially detected carriages (but not the sequence of the carriages) needs to be confirmed, which can be accomplished by pinging all known carriages at regular intervals.

The last limitation is memory capacity. In our algorithm, carriages are correlated in pairs. A carriage is part of as many pairs as it has neighbors. Each pair consumes up to 460 bytes of memory in both partners. With the given memory capacity, a node can accommodate up to 16 neighbors. With a maximum of 6 correlations per second, it takes a node 3 seconds to check all its neighbors.

The circumstances in which the data are collected for the simulations, and the implementation is tested are a worst-case scenario. The trains that are used in the tests are of the same type with similar characteristics. They all exhibit the same pattern in acceleration and braking, making the data to correlate very similar. This is the main reason it takes up to 5 seconds to check carriages from different trains and only 1 second when they are in the same train.

A future research topic is the use of better accelerometers. Those used now measure up to 2g acceleration and can be used on trains that accelerate moderately. However, heavy trains that accelerate and brake more slowly have less distinctive movement patterns and need more sensitive sensors.

The principle of opportunistic sensor networks is shown here in an application that determines the composition of a train. The same principle can be used in many more applications, where movement, or any other type of sensor input, is a distinctive feature. An example is a training application, pairing people with objects they hold in their hands [28]. An other class of applications concerns logistics of goods, where groups of objects must be tracked. An example of this application class is a flower auction, where flowers in containers are loaded in trucks for transport. Correlation of the movement of the truck with the loaded containers checks whether the right containers are in the right truck. More recently, research has started in the SenSafety project in the Dutch national research program COMMIT, using sensors in mobile phones in opportunistic sensor networks [29]. Anticipated distinctive features will be acceleration, direction of movement (compass), sound and camera input. The area of application is safety in public spaces.

ACKNOWLEDGMENT

The work on opportunistic sensing in this paper is supported by the SenSafety Project in the Dutch national research program COMMIT. Other parts are supported by the iLAND Project, ARTEMIS Joint Undertaking Call for proposals ARTEMIS-2008-1, Project contract no. 100026

REFERENCES

[1] J. Scholten, J. and P. Bakker., *Opportunistic Sensing in Wireless Sensor Networks*, The Tenth International Conference on Networks, ICN 2011, January 23-28, 2011, St. Maarten, The Netherlands Antilles, pp. 224-229. IARIA.

[2] J. Scholten, R. Westenberg and M. Schoemaker , *Trainspotting, a WSN-based train integrity system*, The Eighth International Conference on Networks, ICN 2009, 1-6 March 2009, Gosier, France. pp. 226-231. IEEE.

[3] S. Bosch, M. Marin-Perianu, R.S. Marin-Perianu, J. Scholten and P.J.M. Havinga, *FollowMe! Mobile Team Coordination in Wireless Sensor and Actuator Networks*, Proceedings of the IEEE International Conference on Pervasive Computing and Communications 2009, 9-13 March 2009, Galveston, Texas, USA. pp. 151-161. IEEE.

[4] R.S. Marin-Perianu, C. Lombriser, P.J.M. Havinga, J. Scholten and G. Troester, *Tandem: A Context-Aware Method for Spontaneous Clustering of Dynamic Wireless Sensor Nodes*, Proceedings of the First International Conference on Internet of Things (IOT2008), March 2008, Zurich, Switzerland. pp. 341-359. Lecture Notes in Computer Science (4952). Springer Verlag.

[5] European Rail Traffic Monitoring System, [Online]. http://www.ertms.com. January 2012.

[6] European Railway Agency, [Online]. http://www.era.europa.eu. January 2012.

[7] H. Scholten, R. Westenberg, and M. Schoemaker, *Sensing train integrity*, IEEE Sensors 2009 Conference, 25-28 October 2009, Christchurch, New Zealand. pp. 669674. IEEE.

[8] R.S. Schwartz, E.M. van Eenennaam, G. Karagiannis, G. Heijenk, W. Klein Wolterink and J. Scholten, *Using V2V communication to create Over-the-horizon Awareness in multiple-lane highway scenarios*, IEEE Intelligent Vehicles Symposium (IV) 2010, 21-24 June 2010, La Jolla, CA, USA. pp. 998-1005. IEEE.

[9] M. Kumar, *Distributed computing in opportunistic environments*, UIC 09: Proceedings of the 6th International Conference on Ubiquitous Intelligence and Computing, Berlin, Heidelberg, 2009. Springer Verlag.

[10] L. Lilien, A. Gupta, and Z. Yang, *Opportunistic networks for emergency applications and their standard implementation framework*, Performance, Computing, and Communications Conference, 2002. 21st IEEE International, 0:588593, 2007. IEEE.

[11] L. Pelusi, A. Passarella, and M. Conti, *Opportunistic networking: data forwarding in disconnected mobile ad hoc networks*, Communications Magazine, IEEE, 44(11):134 141, November 2006. IEEE.

[12] I. Akyildiz, W. Su, and Y. Sankarasubramaniam, *A survey on sensor networks,* IEEE Comm. Magazine, vol. 40, pp. 102114, 2002. IEEE.

[13] T. Spyropoulos, K. Psounis, and C. Raghavendra, *Single-copy routing in intermittently connected mobile networks,* in Proc. Sensor and Ad Hoc Communications and Networks (SECON), 2004, pp. 235244.

[14] A. Vahdat and D. Becker, *Epidemic routing for partially connected ad hoc networks,* Department of Computer Science, Duke Univeristy, Durham, NC, Tech. Rep., 2000.

[15] Y. Wang and H. Wu, *Delay/fault-tolerant mobile sensor network (dftmsn): A new paradigm for pervasive information gathering,* IEEE Trans. Mobile Computing, vol. 6, pp. 1021 1034, 2007. IEEE.

[16] A. Lindgren and A. Droia, *Probabilistic routing protocol for intermittently connected networks,* Internet Draft draft-lindgren-dtnrg-prophet-02, Work in Progress, 2006.

[17] J. Burgess, B. Gallagher, D. Jensen, and B. Levine, *Max-prop: Routing for vehicle-based disruption-tolerant networks,* in Proc. of IEEE INFOCOM, 2006. IEEE.

[18] D. Camara, C. Bonnet, and F. Filali, *Propagation of public safety warning message: A delay tolerant approach,* in Proc. IEEE Communications Society WCNC, 2010. IEEE.

[19] R. Murty, G. Mainland, I. Rose, A. R. Chowdhury, A. Gosain, J. Bers, and M. Welsh, *Citysense: A vision for an urban-scale wireless networking testbed*, Proceedings of the 2008 IEEE International Conference on Technologies for Homeland Security, pages 583588. 2008. IEEE.

[20] M. Wirz, D. Roggen, and G. Troester, *Decentralized detection of group formations from wearable acceleration sensors*, Proceedings of the 2009 IEEE International Conference on Social Computing, August 2009. IEEE.

[21] M. Wirz, D. Roggen, and G. Troester, *A methodology towards the detection of collective behavior patterns by means of body-worn sensors*, Proc. of UbiLarge workshop at Pervasive, 2010. IEEE.

[22] N. Davies, D. P. Siewiorek, and R. Sukthankar, *Special issue: Activity-based computing*, IEEE Pervasive Computing, 7(2):2021, 2008.

[23] S. Mann, *Humanistic computing: wearcom as a new framework and application for intelligent signal processing*, Proceedings of the IEEE, 86(11):21232151, 1998. IEEE.

[24] B. Myers, J. Hollan, I. Cruz, S. Bryson, D. Bulterman, T. Catarci, W. Citrin, E. Glinert, J. Grudin, and Y. Ioannidis, *Strategic directions in human-computer interaction*, ACM Computing Surveys, 28(4):794809, 1996. ACM.

[25] Ambient Systems, [Online]. http://www.ambient-systems.net. January 2012.

[26] MATLAB - The Language Of Technical Computing, [Online]. http://www.mathworks.com/products/matlab/. January 2012

[27] Correlation and dependence, [Online]. http://en.wikipedia.org/wiki/Correlation. January 2012.

[28] S. Bosch, R.S. Marin-Perianu, P.J.M. Havinga, M. Marin-Perianu, A. Horst and A. Vasilescu, *Automatic Recognition of Object Use Based on Wireless Motion Sensors,* International Symposium on Wearable Computers 2010, 10-13 October 2010, Seoul, South Korea. pp. 143-150. IEEE.

[29] COMMIT, [Online]. http://www.commit-nl.nl. January 2012.

Access Control in a Form of Active Queuing Management in Multipurpose Operation Networks

Vladimir Zaborovsky*, Vladimir Mulyukha**, Alexander Ilyashenko***, Oleg Zayats
St. Petersburg state Polytechnical University
Saint-Petersburg, Russia
e-mail: vlad@neva.ru*, vladimir@mail.neva.ru**, ilyashenko.alex@gmail.com***

Abstract — **Internet processes information in the form of distributed digital resources, which have to be available for authorized usage and protected against unauthorized access. The implementation of these requirements is not a simple task because there are many ways for its realization in the modern congested multipurpose operation networks. The problem of access control can be presented as the task of identifying the characteristics of virtual connections by calculating the appropriate access code. The paper considers the aspects of the filtering algorithms that reduce the computational requirements of the access code and the dynamic priority processing of the packets in the buffer firewall.**

Keywords-access control, virtual connection, priority queueing management, randomized push-out mechanism

I. INTRODUCTION

Access control to the network resources is an important task of the information security. Distributed digital resources that have to be available for authorized usage, and protected against unauthorized access. In the modern computer networks, informational interaction is occurred using application protocols over virtual transport connections. As the result, the problem of access control can be presented as the task of identifying the characteristics of virtual connections by calculating the appropriate access code.

The complexity of this problem is the fact that the access code can be calculated exactly only after the virtual connection is finished. However, in this case, the access control problem can't be solved, because the access becomes irreversible.

The information protection in computer systems has been discussed for almost 50 years. However, the well-known methods of protection of the local data from a remote attacker don't take into account the specifics of modern computer networks such as:

- Territorial distribution and concurrency;
- The dual nature of access control procedures that doesn't allow to form a "security perimeter" as a static requirement concerning network services;
- Non-locality of network resources and characteristics;
- A semantic gap between security policy description and firewall configuration parameters.

The paper considers the problem of computing the access code for virtual connections passing through the corporate firewall based on the analysis of the packets that form the virtual connections. The estimates of the result are probabilistic, but they could improve the effectiveness of information security introducing various mechanisms to control throughput of such virtual connections.

We propose a formalism in which virtual connections are considered as network “meso” objects and packets are the “micro” ones.

Properties of “meso” objects, such as its throughput, could be changed according to the security policy and the characteristics of the “micro” objects, which are determined while passing through the firewall.

The proposed formalism is applied to the management task of the local user access to the external information resources, which are considered as network “macro” objects. To solve the problem of calculating the dynamic code we suggest using the indicator function, whose properties depend on the information model of the macro object and on the description of the access policy, which defines the rights of users and measured data of packets generated by virtual connection.

In this paper, we propose a new approach to access control flexibility enhancement based on active queuing management mechanism and randomized preemptive procedure. The offered solution can be implemented by a firewall and can be applied in the existing network environments. The adaptability of the proposed mechanism improves network security, but it requires large computational resources of the firewall. The paper suggests the aspects of the filtering algorithms that reduce the computational requirements of the access code and the dynamic priority processing of the packets in the buffer firewall.

In order to realize information security in multipurpose operation networks we propose: 1) the new classification of virtual connections (VC) based on access code: security VC characteristics and throughput requirements; 2) VC model, which takes into account fractal characteristics of packet flows; 3) randomized preemptive queuing management mechanism in congested operation networks. We use a combined method of VCs throughput management that unites principles of feedback and program control within a framework for Policy-based Admission Control (Fig. 1):

- Policy Decision Point (PDP).
- Policy Enforcement Point (PEP) – security-critical component, which protects the resources and enforces the PDP's decision.
- Policy Administration Point (PAP).

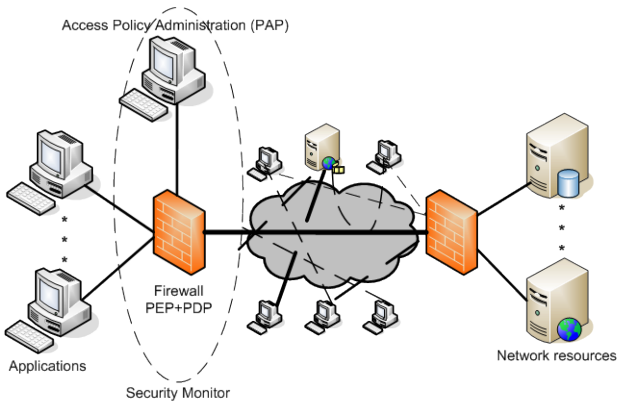


Figure 1. Firewall as a central component of access policy enforcement

In this framework, firewall combines PDP and PEP by controlling access request and enforcing access decisions in real-time. In this case, access control can be considered as the throughput control of VC. So, access to the specific network resource is prohibited when the corresponding VC between the user and resource has no available throughput. Therefore from PAP firewall receives two types of access policy rules: packet filtering rules and data flow rules.

The parameters of firewall rules depend on the set of network environment and/or protocols characteristics *A*. This set can be divided in two classes with different access conditions. In proposed approach, the classification decision is based on access code *F* and firewall has three modes according to possible *F (A)* values (Fig. 2):

- "–1", if the data flow is forbidden according to the access policy (filtering rules);
- "1" and "0" for permitted VCs.

The state of the virtual connection is controlled throughout its lifetime, since the value of access code for "meso" object could change while receiving new "micro" objects.

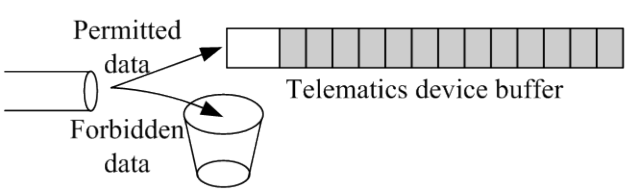


Figure 2. Data flows devision in firewall

When the network environment is congested or when VCs have different QoS requirements the subset of permitted connection has to be divided into new subsets with different access codes:

- "1" for prior "meso" objects that have low throughput and demand low stable delivery time;
- "0" for background ones that demand high throughput and have no delivery time requirements.

For more accurate data sorting we propose to use multiple priority levels. In this paper we consider the simplest situation with two priority levels. It is not enough for practice tasks so we propose some easy ways to increase the number of levels using subsets of permitted VCs (Fig. 3).

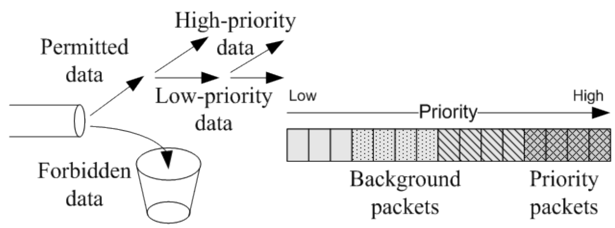


Figure 3. Multiple priority level in congested operation networks

In order to provide this classification procedure we proposed active queuing management mechanism, which is based on randomized preemptive control. Therefore in the firewall, the data flow throughput and time that packets spend in queue (minimum value for priority permitted flows and infinity for denied) are the functions of randomized control parameter α. Each of the firewall rules has a set of attributes, access code: identifiers of subject and object and the access rights from one to another. In the modern network environment, access rules have much more attributes that need to identify two subsets of permitted flows. Therefore the actual problem of access control within framework for Policy-based Admission Control is the flexible configuration of firewall rules, which considers dynamics of network environment including specific congested conditions.

The paper is organized as follows: In Section II, we suggest the architecture of the security monitor. In Section III, there is a new classification of virtual connections. In Section IV, a model of the virtual connection is presented. Sections V and VI are the theoretical parts of the paper where the mathematical model and basic equations are analyzed and estimated. The Section VII is about the practical usage of the proposed method. The Section VIII concludes.

II. Security Monitor Architecture

Computer network security is a main issue of modern information infrastructure. This infrastructure stores information in the form of distributed digital resources, which have to be protected against unauthorized access. However, the implementations of this statement are far from simple due to the dynamic nature of the network environment and users activity [1].

The virtual connection can be described entirely only when it is closed, but in this case, it will not allow us to provide the required level of information security. While we receive information from the VC, there is always the non-zero probability that the VC's properties have been wrongly estimated. In this paper, we consider the architecture of security telematics device that have to decrease this probability using multiple sources of information:

- current data about network traffic that the firewall receives from packet headers fields;
- current data about network environment and users from IDS and special user-activity monitor;
- prior data from informational resource model about expected traffic properties.

So below we describe a new approach to configure the security network appliances, which allows an administrator to overcome the semantic gap between security policy requirements, the ability to configure the firewall filtering rules [2] and to decrease the wrong VC's properties estimation probability. The architecture of the proposed system is presented in Fig. 4.

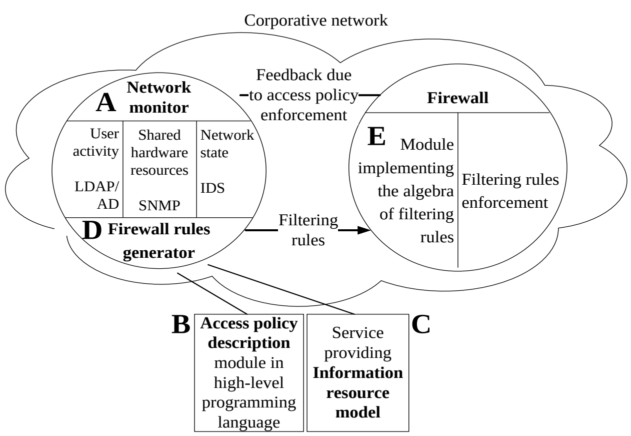


Figure 4. Security monitor architecture

where:

A. Network monitor

Network monitor controls the whole system. Network environment state consists of three main parts:

- "User activity" is the information about what computer is currently used by which user. This information can be obtained from Microsoft Active Directory (AD) by means of LDAP protocol.
- "Shared hardware resources" is the information about network infrastructure and shared internal resources that can be described by network environment state vector Xk
- "Network state" is the information about external network channel received from Intrusion Detection Systems (IDS).

B. Access policy description module

Filtering rules of a firewall in itself are a formalized expression of an access policy. An access policy may simply specify some restrictions, e.g., "Mr. Black shouldn't work with Youtube" without the refinement of the nature of "Mr. Black" and "work" [2],[3].

There is a common structure of access policy requirements, which uses the notions of subject, action and object. Thus, the informally described requirement "Mr. Black shouldn't work with Youtube" can be formally represented as the combination of the subject "Mr. Black", the action "read", the object "www.youtube.com" and the decision "prohibit". This base can also be augmented by a context, which specifies various additional requirements restricting the cases of rule's application, e.g.: time, previous actions of the subject, attributes' values of the subject or object, etc.

However, access rules, which are based on the notions of subject, action and object are not sufficient alone to implement complex real-world policies. As a result, new approaches have been developed. One of them, Role Based Access Control (RBAC) [4], uses the notion of role. A role replaces a subject in access rules and it's more invariant. Identical roles may be used in multiple information systems while subjects are specific to a particular system. As an example, remember the roles of a system administrator and unprivileged user that are commonly used while configuring various systems. Administrator-subjects (persons) may be added or removed while an administrator-role and its rules are not changing.

However, every role must be associated with some subjects as only rules with subjects can be finally enforced. During policy specification roles must be created firstly, then access rules must be specified with references to these roles, then the roles must be associated with subjects.

The OrBAC [5] model expands the traditional model of Role Based Access Control. It brings in the new notions of activity, view and abstract context. An activity is to replace an action, i.e., its meaning is analogous to the meaning of a role for a subject. A view is to replace an object. "Entertainment resources" can be an example of view, and "read" or "write" can be examples of an activity. Thus, the notions of role, activity, view and abstract context finally make up an abstract level of an access policy. OrBAC model allows to specify the access rules only on an abstract level using the abstract notions. Those are called the abstract rules. For instance, an abstract rule "user is prohibited to read entertainment resources", where "user" is a role, "read" is an activity, and "entertainment resources" is a view. The rules for subjects, actions and objects are called concrete access rules.

To specify an OrBAC policy, a common language, XACML (eXtensible Access Control Markup Language) was introduced. The language maintains the generality of policy's specification while OrBAC provides additional notions for convenient editing.

C. Firewall rules generator

There is a feature common for all firewalls: they execute an access policy. In common representation, the main function of access control device (ACD) is to decide whether a subject should be permitted to perform an action with an object. A common access rule "Mr. Black is prohibited to read www.youtube.com".

As was mentioned above, "Mr. Black" is a subject, "HTTP service on www.youtube.com" is an object, and reading is an action. So the configuration of ACD consists of common access rules that reference the subjects, actions and objects.

Although a firewall as an ACD must be configured with common access rules, each implementation uses its own specific configuration language. The language is often hardware dependent, reflecting the features of firewall's internal architecture, and usually being represented by a set of firewall rules. Each rule has references to host addresses and other network configuration parameters. An example of the verbal description of a firewall rule may go as follows:

Host with IP address 10.0.0.10 is prohibited to establish TCP connections on HTTP port of host with IP address 208.65.153.238.

The main complexity of this approach is to find out how such elementary firewall rules could be obtained from common access rules.

Each firewall vendor reasonably aims at increasing its sales appeal while offering various tools for convenient editing of firewall rules. However, so far the problem of obtaining firewall rules from common access rules is not resolved in general. Moreover, this problem has not been paid much attention to.

The most obvious issue concerning this problem is that additional information beyond access rules is necessary in order to obtain the firewall rules. This information concerns the configuration of network services and the parameters of network protocols that are used for data exchange – "network configuration". In general, it can be stored among the descriptions of subjects, actions and objects. An example:

Mr.Black: host with IP-address = 10.0.0.10;
www.youtube.com: HTTP service (port 80) on host with IP-address = 208.65.153.238.

Thus, the final firewall rules can be obtained by addition of the object descriptions to the access rules. It should be noted that even for small and especially for medium and large enterprises it is necessary to store and manage the network configuration separately from the security policy. The suggested approach allows us to achieve this goal: the security officer can edit the access rules with reference to real objects while the network administrator can edit the parameters of the network objects [2].

It should also be noted that there is no need to specify any fixed rules regarding association of the network parameters with the objects. For instance, HTTP port may be a parameter of an object or it may be a parameter of an action. A criterion is that the most natural representation of access policy must be achieved.

While generating the rules, the parameters of network objects can be automatically retrieved from various data catalogs. DNS is the best example of a world-wide catalog, which stores the network addresses. Microsoft offers the network administrators the powerful means, Active Directory, to store information about users. Integration with the above mentioned technologies greatly simplifies the work of a security officer as he has only to specify the correct name of an object while forming firewall rules.

D. Information resource model

Interaction between subject and object in computer network can be presented as a set of virtual connections. Virtual connections can be classified as technological virtual connections (TVC) or information virtual connections (IVC). (see Fig. 5).

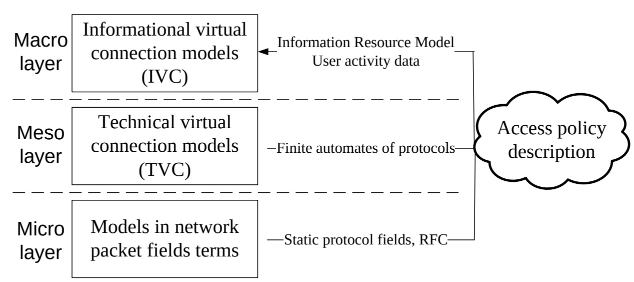


Figure 5. Layers of access control policies.

To implement the policy of access control, the filtering rules are decomposed in the form of TVC and IVC. These filtering rules can be configured for different levels of the data flow description based on the network packet fields at the levels of channel, transport, and application protocols.

At different layers of access control policy model, the filtering rules have to take into account various parameters of network environment and objects. At the packet filter layer, a firewall considers standards static protocol fields described by RFC. At the layer of TVC, firewall enforces the stateful inspection using finite automata describing states of transport layer protocols. On the upper layer of IVC firewall must consider a-priori information about subject and object of network interaction [6].

As was mentioned above, the information about subject can be obtained from catalog services by LDAP protocol, e.g. Microsoft Active Directory.

According to existing approach [7] a resource model can be presented in:

1) logical aspect – an N-dimensional resource space model [8];
2) representation aspect - the definition based on standard high-level description languages like XML or OWL [9];
3) location aspect – the physical storage model of the resource including resource address.

All these approaches describe the network resource as a whole but don't take into account the specific access control task. Any remote network resource can be fully classified when the connection between this resource and local user would be closed. So it is necessary to control all virtual connections in real time while monitoring traffic for security purpose.

In this paper, we propose to implement a special service external to the firewall that would collect, store and renew information about remote network objects. It should automatically create information resource model, describing all informational virtual connections that have to be established to receive this resource. This service should periodically renew information about resource to keep it alive.

Firewall should cooperate with this external service to receive information resource model and enforce access policy requirements.

E. Algebra of filtering rules

As was mentioned above, the information security is defined by an access policy that consists of access rules. Each of these rules has an access code, a set of attributes; the basic ones among them are identifiers of subject and object and the rights of access from one to another. In TCP/IP-based distributed systems, access rules have additional attributes that help to identify flows of packets (sessions) between the client and network application server. Generally these attributes identify the network subjects and objects at different layers of TCP/IP interaction model: MAC-addresses at link layer, IP-addresses at network layer, port numbers at transport layer and some parameters of application protocols.

The access policy in large distributed informational system consists of a huge number of rules that are stored and executed in different access control appliances. The generation of the access policy for such appliances is not very difficult: information must be made available for authorized use, while sensitive data must be protected against unauthorized access. However, its implementation and correct usage is a complex process that is error-prone. Therefore the actual problem of rule generation is representation, analysis and optimization of access policy for large distributed network systems with lots of firewall filtering rules. In our papers, we proposed an approach to description, testing and verification of access policy by the means of specific algebra with carrier being the set of firewall filtering rules. According to proposed approach we define a ring as algebraic structure over set of filtering rules or R [10].

III. VIRTUAL CONNECTION CLASSIFICATION

In this paper, we use the term "access management" as the combination of access control and traffic management. Access control is the basic technical method of information security in the computer networks. It is providing confidentiality by blocking the denied data streams, availability by permitting legal connections and integrity by reducing the risk of data modification or destruction. Confidentiality, integrity and availability are the core principles of information security. Access control is based on subject-object model, where subjects are the entities that can perform actions in the system and influence the environment condition and objects are the entities representing passive elements between which access need to be controlled. Data flows between objects and subjects named virtual connections. As it was mentioned before, the virtual connection is the type of information interaction between applications on object and subject by means of formation one-way or duplex packet stream, and also the logical organization of the network resources necessary for such interaction.

Computer network can be considered as the set of such VCs. In classical subject-object model the set of VCs is divided into two subsets by security characteristic:

- Non forbidden connections that do not harm the protected information;
- Forbidden connections that can low the confidentiality, integrity or availability of protected information.

From another point of view the set of VCs can be divided into several subsets by the type of transmittable information and its quality of service request:

- priority ones, which demand low stable delivery time;
- non-priority ones, which demand high throughput and have no delivery time requirements.

The last subset of non-priority VCs also could be divided into several subsets with different priority levels. So in this paper, we present the simplest example with three subsets:

1) priority non forbidden connections;
2) background non forbidden connections;
3) forbidden connections.

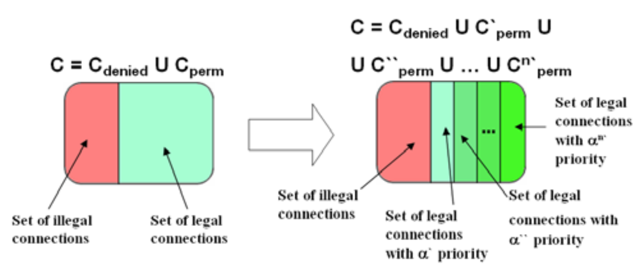


Figure 6. Virtual connections classification model

As it was described there are always type I and type II classification errors, but using additional information in proposed architecture of security monitor we are trying to minimize them.

On Fig. 6 there is graphical interpretation of considered classification.

IV. Virtual Connection Model

The modeling of the VC behavior has received considerable attention in recent years. In this paper, we present a simple model of VC. Each connection can be described by several parameters:

$$Vc(S, O, Th, Type, Fr)$$

where S, O are the subject and object of information interaction, Th – virtual connection throughput, $Type$ – the resource requirements, Fr – fractal nature of VC.

From this point of view we suggest to divide set of virtual connections into two subsets by Fr characteristic:

- fractal natured virtual connections based on transport protocols with feedback (TCP connections;
- data flows without fractal properties like UDP data streams.

Researches have shown that fractal properties of VCs influence its throughput. For calculation the average throughput of TCP connection it is necessary to create a model of connection with fractal properties.

In this paper, we suggest to use a simple discrete time model of TCP connection: at each discreet time moments "k" TCP throughput "Th" can be describes by formulas:

$$X_{k+1} = R(A, X_k, \xi_k) X_k, \quad Th_k = F(X_k),$$

where X – congestion window, which size measures in conventional unit, A – vector of the protocol deterministic characteristics; ξ - stochastic variable described by density distribution function [1],[11]

$$R(A, X_k, \xi_k) = \begin{cases} 1;\ \xi_k = 0,\ X_k = C \\ 1/2;\ \xi_k = 1 \\ 1/X_k;\ \xi_k = 2 \\ 2;\ \xi_k = 0,\ X_k < C,\ X_k < S \\ (X_k + 1)/X_k;\ \xi_k = 0,\ X_k < C,\ X_k > S \end{cases}$$

where C is TCP receive window size, S – threshold.

As it is known from an example of Cantor set the fractal properties appears at loss of the set's part. Fractal properties of TCP-connection characterize the throughput losses because of feedback mechanism. On Fig. 7 there are shown the throughput losses because of CWND adaptation mechanism.

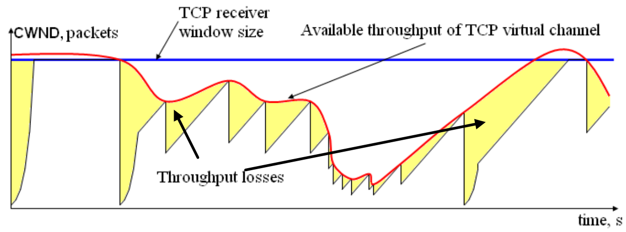


Figure 7. TCP throughput losses because of CWND mechanism

We suggest using different algorithms to calculate the throughput of VC with fractal properties and without ones.

For the connections of type 2 without fractal properties we will use the simple formula:

$$Th = Th_0 \cdot (1 - p),$$

where Th_0 is the connection throughput from the stream source and p is the packet loss probability.

For TCP connections (type 1) we use the well-known formula:

$$Th = \min\left(\frac{C}{RTT}; \frac{1}{RTT \cdot \sqrt{\frac{2}{3} p}}\right),$$

where C is TCP receive window size, RTT is round trip time and p is the packet loss probability (loss rate). The graph of this function for C = 100 packets and RTT=100 ms is shown on Fig. 8.

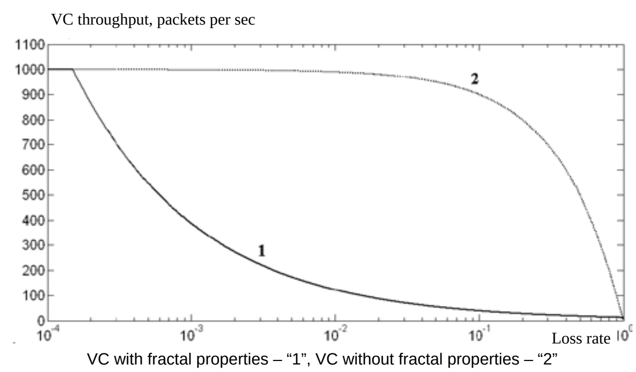


Figure 8. Dependence of TCP throughput on packet loss probability for TCP connections

V. Model of Network Environment

According to the VC models written above we consider the preemptive priority queueing system with two types of customers. First type of customers has priority over the second one. The customers of the type 1 (2) arrive into the buffer according to the Poisson process with rate λ_1 (λ_2). The service time has the exponential distribution with the same rate μ for each type. The service times are independent of the arrival processes. The buffer has a finite size k ($1<k<\infty$) and it is shared by both types of customers. The absolute priority in service is given to the customers of the first type. Unlike typical priority queueing considered system is supplied by the randomized push-out mechanism that helps precisely and accurate to manage customers of both types. If the buffer is full, a new coming customer of the first type can push out of the buffer a customer of type 2 with the probability α. We have to mention that if $\alpha=1$ we retrieve the standard non-randomized push-out.

The scheme described priority queueing is resulted on Fig. 9. The priority queueing without the push-out mechanism ($\alpha=0$) and with the determined push-out mechanism ($\alpha=1$) are well-studied. The concept of the randomized push-out mechanism with reference to network and telecommunication problems is offered in [12] where this mechanism was combined with relative priority, instead of absolute, as in our case.

The summarized entering stream represented on Fig. 9 will be the elementary with intensity: $\lambda=\lambda_1+\lambda_2$. If we'll trace only the general number of packets in system, then simplified one-data-flow model would be $M/M/1/k$ type. In the modified by G.P.Basharin Kendel notation, the general structure of a label and sense of its separate positions remains, however in each position the vectorial symbolic is used [13]. There is an additional symbol $f_i^{\,j}$, where i specifies priority type (0 – without a priority, 1 – relative, 2 – absolute), and j specifies a type of the pushing out mechanism (0 – without pushing out, 2 – the determined pushing out). So $j=1$ wasn't used. In [12], authors offer to use this value for the randomized push-out mechanism, as an intermediate between variants $j=0$ and $j=2$. So, using this new notation, system represented on Fig. 9 has $\vec{M}_2/M/1/k/f_2^1$ type.

The history of one-channel two data-flow priority systems research includes already more than half a century, however, as far as we know, there is only one work [12] where the randomized push-out mechanism have been studied (in a combination with the relative priority for queueing $\vec{M}_2/M/1/k/f_1^1$ type). At the same time, for the typical models with the push-out mechanism ($j=0$ and $j=2$) the problem is solved basically.

Problems of research priority queueing have arisen in telecommunication with the analysis of real disciplines of scheduling in operating computers. Last years a similar sort of queueing model, and also their various generalizations are widely used at the theoretical analysis of Internet systems.

As has been shown in [12], the probability pushing out mechanism is more convenient and effective in comparison with other mathematical models of pushing out considered in the literature. It adequately describes real processes of the network traffic and is simple enough from the mathematical point of view. The randomized push-out mechanism helps precisely traffic management and security. Another control and security factor is the telematics device buffer size. It can be varied to increase the throughput of necessary connections and reduce throughput of suspicious ones.

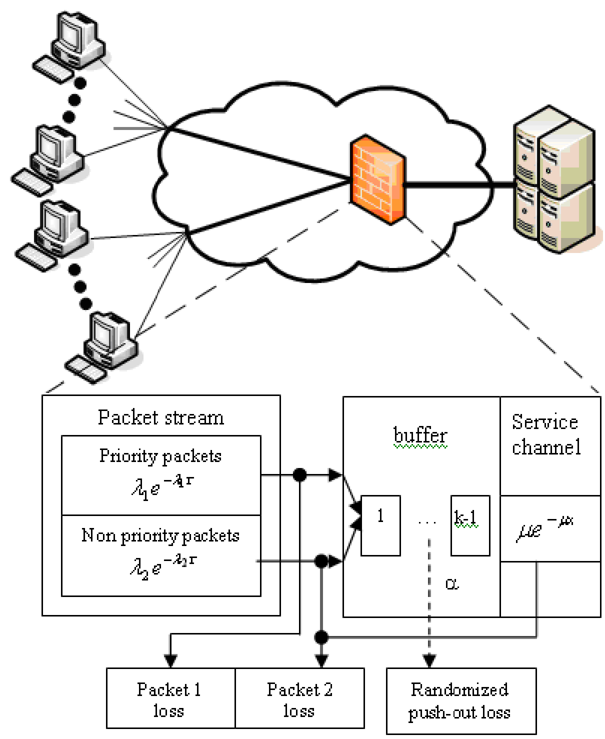


Figure 9. Priority queueing schema $\vec{M}_2/M/1/k/f_2^1$ of telematics network devide

VI. MAIN EQUATIONS

The state graph of system $\vec{M}_2/M/1/k/f_2^1$ is presented on Fig. 10.

Making by usual Kolmogorov's rules set of equations with the help of state graph we will receive:

$$\begin{aligned}&-[\lambda_1(1-\delta_{j,k-i})+\alpha\lambda_1(1-\delta_{i,k})\delta_{j,k-i}+(1-\alpha)\lambda_1\delta_{i,0}\delta_{j,k-i}+\\&+\lambda_2(1-\delta_{j,k-i})+\mu(1-\delta_{i,0}\delta_{j,0})]P_{i,j}+\mu P_{i+1,j}+\mu\delta_{i,0}P_{i,j+1}+\\&+\lambda_2 P_{i,j-1}+\lambda_1 P_{i-1,j}+\alpha\lambda_1\delta_{j,k-i}P_{i-1,j+1}+\\&+(1-\alpha)\lambda_1\delta_{j,k-i}\delta_{i,1}P_{i-1,j+1}=0,(0\le i\le k;0\le j\le k-i),\end{aligned} \quad (1)$$

where $\delta_{i,j}$ is the Kroneker's delta-symbol.

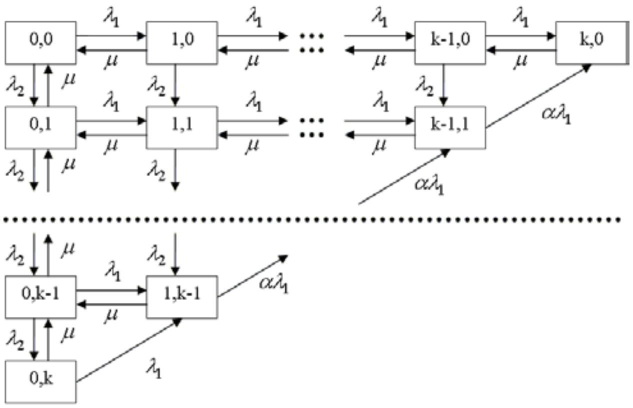


Figure 10. The state graph of $\vec{M}_2 / M / 1 / k / f_2^1$ type system

There is a normalization condition for the system:

$$\sum_{i=0}^{k}\sum_{j=0}^{k-i} P_{ij} = 1.$$

At real k (big enough) this system is ill-conditioned, and its numerical solution leads to the big computing errors. In this paper, we use the method of generating functions [12] in its classical variant offered by H.White, L.S.Christie [14] and F.F.Stephan [15] with reference to $\vec{M}_2 / M / 1 / f_2$ type systems. According to generating function method and normalization condition we have:

$$G(u,v) = \sum_{i=0}^{k}\sum_{j=0}^{k-i} P_{i,j} u^i v^j, \quad G(1,1) = \sum_{i=0}^{k}\sum_{j=0}^{k-i} P_{i,j} = 1.$$

And after several transformations result equation for generating function will be:

$$\begin{aligned}
&[\lambda_1 u(1-u) + \lambda_2 u(1-v) + \mu(u-1)]vG(u,v) = \\
&= \mu(u-v)G(0,v) + \mu u(v-1)G(0,0) + \\
&+\alpha\lambda_1 u^{k+1}(v-u)P_{k,0} + [\alpha\lambda_1(u-v) + \\
&+\lambda_1(1-u)v + \lambda_2(1-v)v]u\sum_{i=0}^{k} P_{i,k-i}u^i v^{k-i} + \\
&+(1-\alpha)\lambda_1 P_{0,k} v^k u(u-v).
\end{aligned}$$

Solving this equation and (1) system we receive some auxiliary variables:

$$p_i = P_{k-i,i}, \quad (i = \overline{0,k}),$$

$$q_{k-j} = (1-\alpha)\sum_{i=1}^{j} p_i \rho_1^{i-j} + p_0 \rho_1^{-j} - (1-\alpha) p_k \delta_{j,k}, (j = \overline{0,k}),$$

$$r_n = \frac{(1-\rho)\rho^n}{(1-\rho^{k+1})}, (n = \overline{0,k}),$$

$$\begin{aligned}
G(u,v) = &\frac{(u-v)G(0,v) + u(u-1)G(0,0)}{v\rho_1(u-u_1)(u-u_2)} + \\
&+\frac{\alpha\rho_1 u^{k+1}(v-u)P_{k,0} + (1-\alpha)\rho_1 P_{0,k} v^k u(u-v)}{v\rho_1(u-u_1)(u-u_2)} + \\
&+\frac{[\alpha\rho_1(u-v) + \rho_1(1-u)v + \rho_2(1-v)v]u\sum_{i=0}^{k} P_{i,k-i}u^i v^{k-i}}{v\rho_1(u-u_1)(u-u_2)}.
\end{aligned}$$

When using them, we can receive loss probability for priority ($P_{loss}^{(1)}$) and non-priority ($P_{loss}^{(2)}$) packets:

$$P_{loss}^{(1)} = q_k + (1-\alpha)\sum_{i=1}^{k-1} p_i, \quad P_{loss}^{(2)} = r_k + \alpha\frac{\rho_1}{\rho_2}\sum_{i=1}^{k-1} p_i. + \frac{\rho_1}{\rho_2} p_k$$

Exploring these formulas we found some useful properties of this system described in this article. One of them presented on Fig.11, 12. When incoming stream of priority packets getting more intensive, system starts to prohibit admission of non-priority packets. While the total flow rate is less than unity ($\rho_1 + \rho_2 \le 1$), the probability of loss is equal to zero. This means that the system is fully copes with the load (see Fig.11). In Figure 12, the graph does not start from zero because the system is initially overloaded with non-priority packets. Same effect and in this case.

On Fig. 13 an expected result can be seen that the probability of losing priority packet decreases with increasing size of a buffer, but not as much as has been expected. Probability of loss is decreasing not more than 5% for small values of α. Therefore, only for large probability values increasing buffer size effectively influences the losses. For priority stream influence of this effect is the same for all values of alpha, but for non-priority packets the situation is different. Figure 14 shows that it is sometimes advantageous to have a buffer of smaller size. With a small buffer probability of be pushed out much lower, what explains this effect.

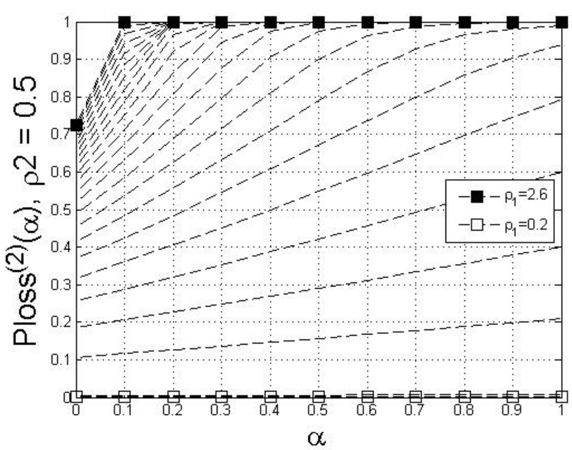


Figure 11. Loss probability of non-priority packets with 0.1 step for $0.2 \le \rho_1 \le 2.6$, buffer size 31 and weak non-priority stream

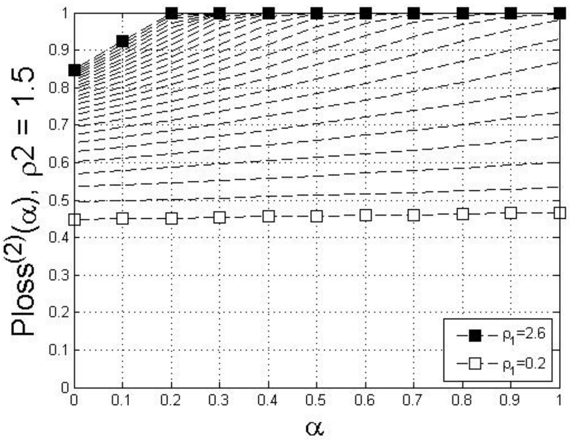


Figure 12. Loss probability of non-priority packets with 0.1 step for $0.2 \le \rho_1 \le 2.6$, buffer size 31 and more ntensive non-priority stream

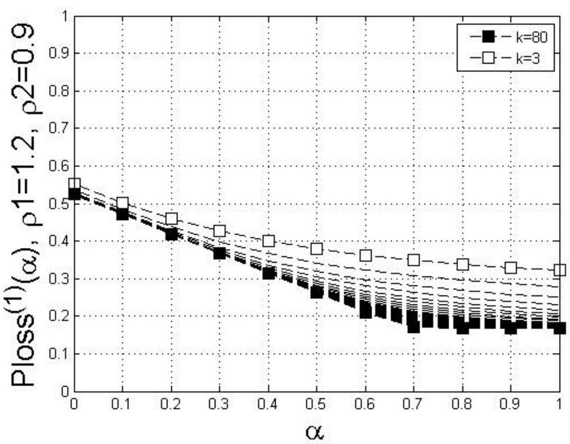


Figure 13. Loss probability of priority packets with buffer size K=3-80

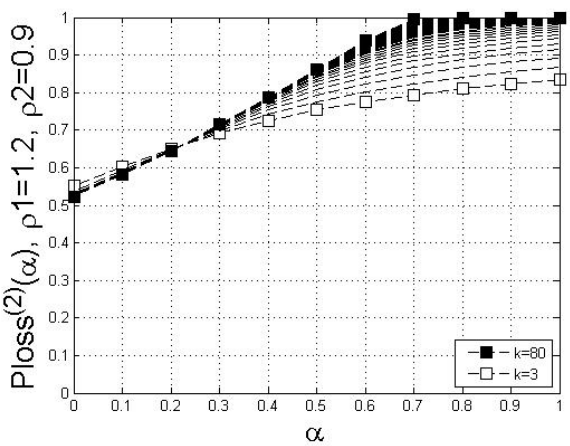


Figure 14. Loss probability of non-priority packets with buffer size K=3-80

Graphs on Fig.11, 12 are inverted images of the graphs of the relative throughput, which is computed by formulas (2) and are very important for research of processes in computer networks.

$$æ_i = 1 - P_{loss}^{(i)},\ (i = \overline{1,2}). \qquad (2)$$

From Fig. 11 and 12 we can see, that by choosing parameter α, we can change $P_{loss}^{(2)}$ in very wide range. For some ρ_1 values variable æ$_i$ changes from 0.7 to 1 while $\lambda_1 + \lambda_2 >> \mu$.

Next interesting variable is average queue length of priority packets (see Fig.15, 16, 17), computing as (3). While the system is not loaded, the average queue length is zero, as shown in the bottom of the chart (see Fig. 13). But once the system begins to fill, then average queue length begins to grow rapidly. And as seen in the Figures 15, 16, 17, that by using the α be strong enough to influence the filling of the queue. In some cases, change the setting at 0.1 entails the complete filling of the queue.

$$\overline{n_{оч}}^{(1)} = \sum_{i=1}^{k}(i-1)q_i = \overline{n}^{(1)} - \sum_{i=1}^{k} q_i = \overline{n}^{(1)} - (1-q_0). \qquad (3)$$

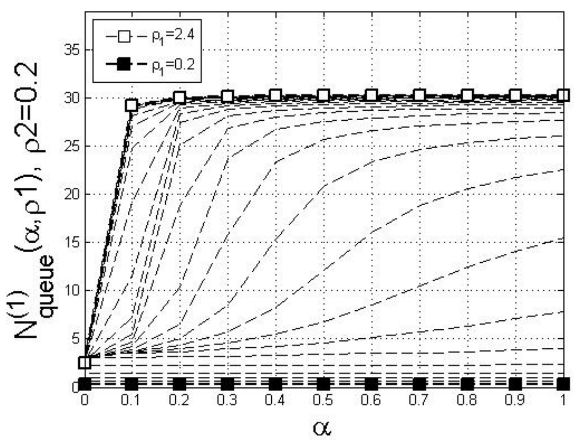


Figure 15. Average priority queue length with low intensity of second stream with buffer size 31

The relative time that the priority packet spend in queueing can be calculated by Little's Formula (Fig 18,19,20) [16]:

$$\theta_i = \frac{\overline{s}_i}{\overline{\tau}_i} = \frac{\overline{n}_{system}^{(1)} + \delta_{i,2}\overline{n}_{system}^{(2)}}{(1-\overline{P}_{loss}^{(i)})},\ \overline{\tau}_i = \frac{1}{\lambda_i},\ (i = \overline{1,2}).$$

Fig 18, 19, 20 show that proposed queueing mechanism provide a wide range of control feature by randomized push-out parameter α and buffer size k. According to the packet's mark (Forbidden, Priority, Background) the period that packet spend in queue can vary from 1 to 10^{14} times, which can be used to control access to information resource providing confidentiality.

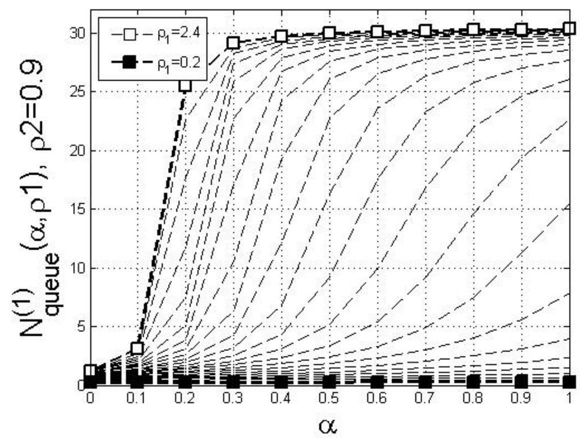


Figure 16. Average priority queue length with medium intensity of second stream with buffer size 31

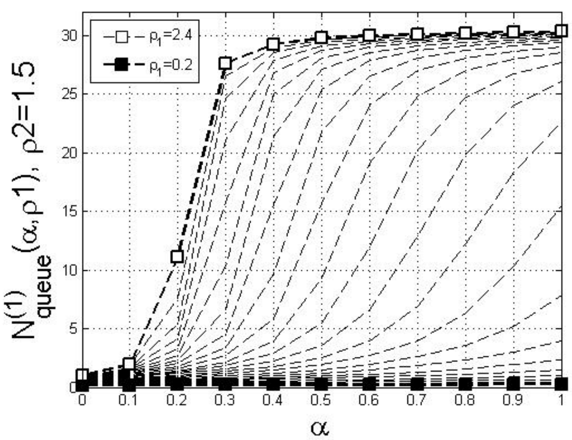


Figure 17. Average priority queue length with high intensity of second stream with buffer size 31

For highly congested network the priority type is much less important, than the push-out mechanism and the value of α parameter. The push-out mechanism allows to enforce access policy using traffic priority mechanism.

By choosing α parameter we can change the time that packets spend in the firewall buffer, which allows to limit access possibilities of background traffic and to block forbidden packets. So by decreasing the priority of background VCs and increasing the push-out probability α we can reduce the VC throughput to low level without interrupting it.

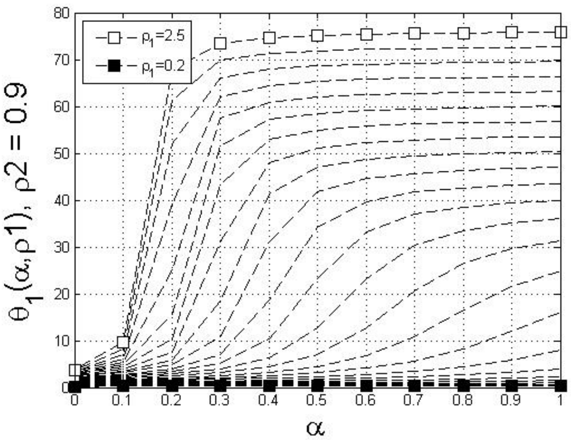


Figure 18. The time that priority packet spend in queueing

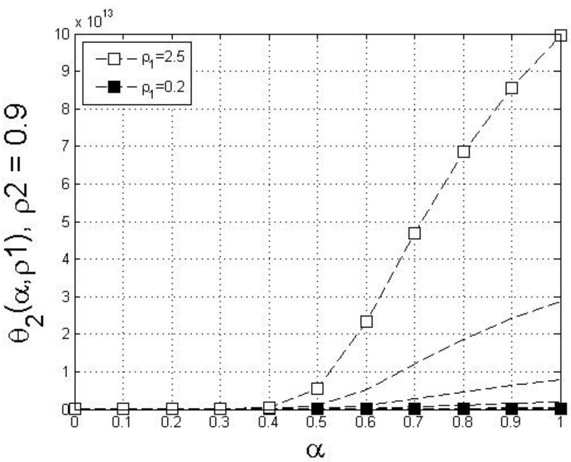


Figure 19. The time that non-priority packet spend in queueing

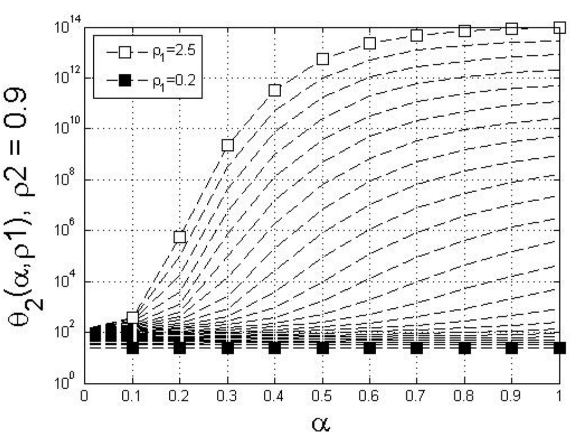


Figure 20. The time that non-priority packet spend in queueing built in logarithmic y scale

The most wide range of control can be reached in intermediate environment conditions when linear law of the losses has already been broken, but the saturation zone has not been reached yet. Numerical experiment [17] has been made to detect conditions in which ρ_1 varied over a wide range from 0,1 to 2,5, and few fixed values for ρ_2.

VII. Practical usage and future development.

Good example of opportunity to use such mechanism is the problem of controlling removed robotic object, which telemetry data and a video stream are transmitted on global networks. In this case, control commands are transmitted by TCP, and a video stream data are transmitted by UDP. A mean values of throughput of our robotic object: throughput of TCP channel (control and telemetry packets) ~100Kb/s, throughput of UDP video stream ~1,2Mb/s.

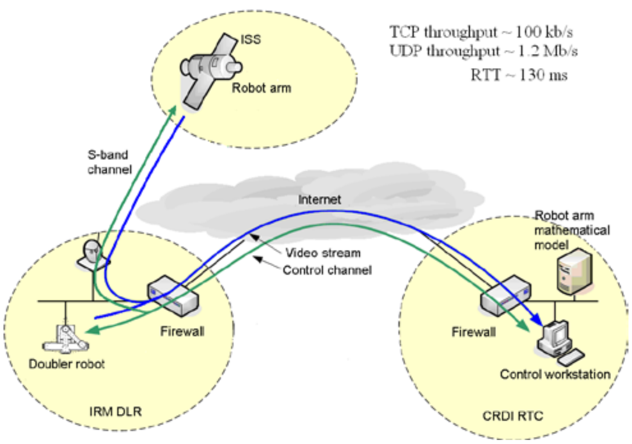


Figure 21. The scheme of space experiment "Contour"

In a considered example on Fig. 21 (ROKVISS mission [18]), the choice of a priority of service and loss-probability of a priority packet α allows to balance such indicators of functioning of a network, as loss-probability of control packets $p_{loss}^{(1)}$ and quality of video stream for various conditions of a network environment. The parameter α can vary for delay minimization in a control system's feedback.

The given problem is important for interactive control of remote real-time dynamic objects, in a case when the complex computer network is the component of a feedback control contour, therefore minimization of losses and feedback delays, is the important parameter characterizing an effectiveness of control system.

In future, this method of preemptive access management could be used in new joint space experiment METERON-R (Multi-purpose Experimental TElecommunication Robotic Operations Network - Russia) that will be carried out on ground and on-board the ISS, in order to research efficiency and security of robotic operations in space and ground environments, including the configuration of robotic control systems as a part of multipurpose operations network (Fig. 22). The joint experiments will focus on the analysis of how well astronauts can operate complex robotic systems based on operation networks with mobility and manipulation capability from within the highly constrained ISS and micro-gravity environment. Multiple human-robot interfaces will be used in combination, while simulating realistic robotic remote operations with round-trip time communication conditions representative of future human planet exploration missions.

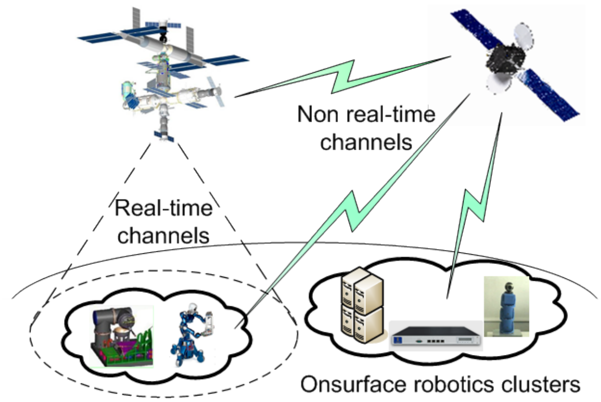


Figure 22. The scheme of space experiment "METERON-R"

For communication experiments, the primary focus will be on the usage of real-time duplex commanding, in combination with Delay Tolerant Network (DTN) approaches and Inmarsat channel. Real-time channel will have low delay (15-20 ms) and high throughput (4 Mb/s), but the connection would be established only when the space station is in the radio-optical range (7-10 min). DTN channels have high delay and low throughput, but function for 24/7. Inmarsat channel characteristics are between real-time and DTN and they are much depend on quantity of retransmission satellites.

Robotics objects within multipurpose operation network would execute the programs and interact without human involvement. However there would be always situations when the robot couldn't make a decision by its own. In that case, the human-operator will have several opportunities:

1) remote telecontrol through real-time channel;
2) to send several commands or additional data through Inmarsat channel;
3) to send new program through DTN.

Each of these data will have its own priority level. So in this case, two types of priority are not enough for traffic management in multipurpose operation network environment, but the recurrent mode of proposed procedure can increase the number of priority VCs subsets.

VIII. Conclusion.

1. The offered access control approach allows more deeply and more detailed understanding of requirements of access policy in the form of firewall configuration rules.

2. In multipurpose operation networks, we propose a new formalism in which the distributed digital resources are considered as “macro” objects, virtual connections are the network "meso" ones and packets are the “micro” objects. Proposed formalism allows to enforce security policy and provides authorized usage and protection against unauthorized access.

3. Proposed model based on DiffServ approach considers computer network as the set of VCs, which throughput is easy controlled by proposed classification procedure and algorithm that divides the set of non forbidden VCs in two subsets: non forbidden priority connections and non forbidden non priority or background connections.

4. Introduced VC model takes into account several parameters such as: dynamic and statistics characteristics including fractal properties of VC with feedback throughput control like TCP.

5. Considered preemptive queueing mechanism can be viewed as a background for DiffServ access control because it provides a wide range packet loss probability ratio using flexible randomized push-out algorithm.

6. Proposed push-out algorithm based on selecting priority parameter controls packet loss probability taking into account restricted capacity of packet buffer in DiffServ access point. The most interesting result obtained in congested network allows to keep priority VC throughput near the requested value, which is important for specific space experiment with robotics arm on ISS board.

7. We described the future usage of proposed formalism in joint space experiment where several types of operations are serviced by security monitor in multipurpose operation network.

References

[1] Zaborovsky V. and Mulukha V. Access Control in a Form of Active Queuing Management in Congested Network Environment // Proceedings of the Tenth International Conference on Networks, ICN 2011 pp.12-17.

[2] Zaborovsky V. and Titov A. Specialized Solutions for Improvement of Firewall Performance and Conformity to Security Policy, Proceedings of The 2009 International Conference on Security and Management, Volume II, Published by CSREA Press, USA 2009, p.603-608

[3] Titov A. and Zaborovsky V. Firewall Configuration Based on Specifications of Access Policy and Network Environment // Proceedings of the 2010 International Conference on Security & Management. July 12-15, 2010.

[4] Ferraiolo D.F. and Kuhn. D.R. Role-Based Access Control. 15th National Computer Security Conference. (October 1992), pp. 554–563. (http://csrc.nist.gov/groups/SNS/rbac/documents/ferraiolo-kuhn-92.pdf)

[5] http://orbac.org/index.php?page=orbac&lang=en

[6] Zaborovsky V., Lukashin A., and Kupreenko S. Multicore platform for high performance firewalls. High performance systems // Materials of VII International conference – Taganrog, Russia.

[7] Zhuge H., The Web Resource Space Model, Berlin, Germany: Springer-Verlag, 2007

[8] Zhuge H., Resource Space Grid: Model, Method and Platform, Concurrency and Computation: Practice and Experience, vol. 16, no. 14, pp. 1385-1413, 2004

[9] Martin D., Burstein M., J. Hobbs, O. Lassila. et al. (November 2004) “OWL-S: Semantic Markup for Web Services,” [Online]. Available: http://www.w3.org/Submission/OWL-S/.

[10] Silinenko A. Access control in IP networks based on virtual connection state models: PhD. Thesis 05.13.19: / SPbSTU, Russia, 2010.

[11] Vladimir Zaborovsky, Aleksander Gorodetsky, and Vladimir Muljukha Internet Performance: TCP in Stochastic Network Environment, Proceedings of The First International Conference on Evolving Internet INTERNET 2009, Published by IEEE Computer Society, 2009, p.447-452

[12] Avrachenkov K.E., Vilchevsky N.O., and Shevljakov G.L. Priority queueing with finite buffer size and randomized push-out mechanism // Proceedings of the ACM international conference on measurement and modeling of computer (SIGMETRIC 2003). San Diego: 2003, p. 324-335.

[13] Basharin G. P. A single server with a finite queue and items of different types // Teor. Veroyatnost. i Primenen., 1965, Volume 10, Issue 2, Pages 282–296

[14] White H. and Christie L.S. Queueing with preemptive priorities or with breakdown // Operations research, 1958, vol. 6, no. 1, p. 79-95.

[15] Stephan F.F. Two queues under preemptive priority with Poisson arrival and servive rates // Operations research, 1958, vol. 6, no.3, p. 399-418

[16] L. Kleinrock. Queueing Systems Volume I-II, 1976.

[17] Zaborovsky V., Zayats O., and Muljukha V. Priority Queuing with Finite Buffer Size and Randomized Push-out Mechanism // Proceedings of the Ninth International Conference on Networks ICN 2010, p.316-321.

[18] http://www.dlr.de/en/desktopdefault.aspx/tabid-727

Performance Analysis and Strategic Interactions in Service Networks

Marina Bitsaki
Transformation Services Lab
University of Crete
Crete, Greece
e-mail: bitsaki@tsl.gr

Christos Nikolaou
Transformation Services Lab
University of Crete
Crete, Greece
e-mail: : nikolau@tsl.gr

Manolis Voskakis
Transformation Services Lab
University of Crete
Crete, Greece
e-mail: voskakis@tsl.gr

Willem-Jan van den Heuvel
European Research Institute in Service Science
Tilburg University
The Netherlands
e-mail: wjheuvel@uvt.nl

Konstantinos Tsikrikas
Transformation Services Lab
University of Crete
Crete, Greece
e-mail: cotsik@tsl.gr

***Abstract*—Service businesses are currently viewed as interdependent entities that achieve competitive advantage by fostering partnerships and co-evolving with competitors. Service networks are formed to describe these relationships and reveal value created and shared among them. In this paper, we analyze network participants' behavior aiming to optimize their own value. We describe ecosystems in which more than one competing networks co-exist and interact with one another to their own benefit. We perform simulations to measure the performance of service networks and investigate optimal strategies for competing systems. We describe various scenarios defining dynamic strategies for competing players and show experimentally that after a small number of time slots these strategies reach an equilibrium in which no one is willing to diverge from its decision to his own benefit.**

Keywords-service networks; value optimization; performance analysis; strategic behavior; competing networks

I. INTRODUCTION

In recent years, economic globalization combined with rapid technological progress has led most service companies to coordinate their corporate operations in a world of interactions and partnerships. Companies organize their fundamental structure into service networks capitalizing on the advantage of collaboration. This paper is an extension of [1] and studies the behavior of competing service networks in terms of customer satisfaction and value.

Service networks consist of interdependent companies that use social and technical resources and cooperate with each other to create value [2], [3], [4] and improve their competitive position. The concept of service networks includes a network of relationships between companies where flexibility, quality, cost effectiveness and competitiveness are better achieved than in a single company. The interaction between two participants of a network does not depend solely on their direct connection, but on the impact other relationships within the network have on them. In addition, the network can strengthen business innovation as different parts contribute services to an overall value proposition combining their know-how and core capabilities.

The emergence of service networks forces companies to use alliances in order to achieve a competitive advantage. Man [5] describes various tactics followed by companies to advance their position and analyzes the types of competition that emerge in the network economy. He identifies three forms of competition: competition in networks (collaborating partners compete with each other), competition between networks and competition with organizational forms (smaller companies joining forces to compete with a network).

While various approaches have been proposed to measure the performance of service networks [6], [7], [8], [1], little experimental testing or theoretical investigation on competing networks has been done [9], [10]. Most of the research has focused on describing models that represent inter-organization exchanges. In [6], a quantifiable approach of value calculation is proposed that connects value with expected revenues. In contrast, Biem and Caswell [7] describe building block elements of a value network model and design a network-based strategy for a prescriptive analysis of the value network. Allee [8] provides a systematic way for approaching the dynamics of intangible value realization, inter-convertibility, and creation. Biem and Caswell [7] and Allee [8] use qualitative methods to describe value in a service network in contrast to Caswell et al. [6] that calculates value in a quantifiable manner. The above approaches do not study strategic behavior of network participants that would result in value optimization.

In [9], strategic behavior of service providers *within* service value networks is studied. An auction-based mechanism is proposed in order to efficiently match service offers and service requests and determine prices. It is shown that incentive compatibility holds under certain conditions.

In [10], industry structure is analyzed in the presence of value networks. A model was developed that deals with a number of design aspects such as the number of suppliers and the importance of partner investments. It is shown that industry structure is more likely to shift to competition between value networks with IT playing an important role on that. Systems that encounter competition *between* service networks and are analyzed with respect to their evolution through well defined strategies have received little attention in literature.

In this paper, we study the impact of strategic changes on the performance both at the level of the network as well as its participants compared to that of a competing network. In particular, we describe a framework to analyze competing networks based on the model introduced in [1]. In [1], we measured the performance of one service network solving value optimization problems with respect to service prices. Comparing to previous work that has been done, we improved the estimation techniques and we used a powerful simulation tool to perform our experiments and analyze dynamic "what-if" questions such as: what is the impact of setting optimal – for one participant - prices on the performance of the other participants as well as the entire network? What is the impact on the performance if a new participant suddenly enters the service network? Are there any equilibrium strategies among the participants that eliminate their conflicts of interests?

We observed that participants' value depends on their expected profits. Expected profits express the additional value that will be accrued by the relationship levels a participant develops when it sells goods and services to other participants or to the end customers. This value is related to the degree of satisfaction it obtains from its customers. There are many approaches that have been proposed to measure customer satisfaction. We used the methodology proposed by Fornell et al., known as American Customer Satisfaction Index [11].

In this paper, we extend the initial model to account for competitors of the existing service network. We define the conditions under which two competing networks co-exist, the types of information the networks share and various strategies chosen by the competing participants. We simulate those strategies and observe the behavior of our system over time in terms of profits and market share.

We use the System Dynamics approach [12], [13] to analyze the behavior of a complex system (competing car repair service networks) over time. System dynamics tools allow modelers to succinctly depict complex (service) networks, visualizing processes as behavior-over-time graphs, stock/flow maps, and causal loop diagrams. These models can be tested and explored with computer simulation providing for example better understanding of the impact of policy changes (e.g., through animation of (service) systems) and facilities for sensitivity analysis. Examples of such tools include iThink [13], Vensim [14] and PowerSim [15].

In this paper, we have adopted the iThink tool to investigate existence of equilibrium prices chosen by the keystone member of each network. The results of these simulations provide predictions about the future of the service networks in order to increase its adaptability to the changes of the environment and enable network participants to determine the most profitable co-operations and attract new ones. We show that the interactions among the participants of the competing networks force them to reach equilibrium otherwise the network will collapse.

The remainder of this paper is organized as follows: Section II describes the car repair service system. Section III presents the methodology proposed in [1] to estimate value in service systems. In Section IV, we describe the model of two competing networks and identify the objectives of our analysis. In Section V, we run experiments to measure the performance of a single service network as presented in [1]. The results of the simulations are presented in Section VI. In Section VII, we run experiments to derive equilibrium strategies of companies of competing networks. In Section VIII, we show the results. Finally, in Section IX, we provide some concluding remarks.

II. CASE STUDY

The motivating scenario revolves around a service network that links four types of participants: an Original Equipment Manufacturer (e.g., Volvo), Car Dealers (with repair facilities), Suppliers and Customers.

The scenario that we will use during the remainder of this article is an extension to [6] and basically looks as follows. OEM-franchised dealers may service and repair cars for their clients. Both activities require a car parts catalogue to ensure that repairs can be performed efficiently either in the replacement of parts or repairing after accidents. The part catalogue facilitates efficient installation, operation and lifecycle maintenance of intricate products describing detailed part information that can be fully integrated with other service applications supporting customer support processes, human resource management, and other service provisions.

The quality of the OEM parts, catalogues, and OEM support services influences how many OEM parts will be ordered and used for a car repair and how many parts will be used from Third Party Suppliers (TPS), and how many customers will go to OEM dealers or to TPS dealers. OEM obtains parts from certified supply-chain suppliers (SCS).

The technicians report the car service requirements that may include replacing teardowns, warranty replacements and collision repairs. On the basis of the car diagnosis, a cost estimate will be computed and communicated to the client for authorization. Once authorized the automotive technician will scrutinize failure symptoms, detect faulty parts, order parts and perform the repair. Ordering parts is a complex process that involves asking advice from expert technicians from the OEM, including acquiring information about parts under warranty, and getting approval from the dealer's part manager. The part manager then checks local inventory for the required part, and if necessary checks the stock at the OEM or supplier stocks, and eventually places an order. The part manager may either use third-party suppliers or suppliers from certified supply-chain suppliers.

III. THE MODEL

In this section, we introduce our service performance analytics model in support of strategic analysis of service network changes and improvements as presented in [1]. Theorizing on service networks, and particularly performance analysis, can be addressed from multiple and often complementary perspectives. In our work, we propose a methodology to calculate value in service systems. We focus on the dynamic environment in which service networks emerge, and especially on connectivity and profitable cooperation that play an important role in value creation. We use our model to investigate network profitability and give answers to the following:

- Determine the conditions under which it is profitable for a firm to participate in the network and identify the factors that influence its value.
- Identify keystone participants (participants that create the most value for the network).
- Determine participants' optimal strategic decisions (cooperating with someone or not, joining the network or not, etc.).

We consider the service network as a set *B* of participants connected through transfer of offerings that delivers value to them. All offerings are treated as services that are composed by participants' interactions and co-operations to provide a final service to a set *C* of end customers. Let p_{ij} denote the price participant i charges participant j for offering its services and r_{ij} denote the service time of the interaction between participants i and j. Price and time are the main parameters that affect customer satisfaction which is in turn the corner-stone for calculating value as we will see below.

A. Customer Satisfaction

Customer satisfaction measures the willingness of end customers to buy the services offered by the network and influences the increase or decrease of new entries. The calculation of satisfaction $SAT_{ij}(T_N)$ of participant j for consuming services from participant i at the end of the time interval $[T_{N-1}, T_N]$ for our model is a variation of the American Customer Satisfaction Index (ACSI) [12] and is basically described as follows. ACSI is operationalized through three measures: q_1 is an overall rating of satisfaction, q_2 is the degree to which performance falls short of or exceeds expectations, and q_3 is a rating of performance relative to the customer's ideal good or service in the category. Without loss of generality, we quantify the above measures using the following formula:

$$q_k = [(\beta_k/p_{ij})0.6 + (\gamma_k/r_{ij})0.4],\ k=1,2,3, \tag{1}$$

where [x] denotes the integer part of x and β_ks, γ_ks are the parameters that determine the effect of price p_{ij} and time t_{ij} respectively on q_k. In our analysis, we use the following function (see [11] for further details) to calculate the satisfaction:

$$SAT_{ij}(T_N)=(w_1q_1+w_2q_2+w_3q_3-w_1-w_2-w_3)/(9w_1+9w_2+9w_3), \tag{2}$$

where w_k are weights that indicate the importance of each measure q_k.

B. Participants' Value

We consider that an economic entity within a service network has value when it satisfies the entity's needs and its acquisition has positive tradeoff between the benefits and the sacrifices required. We emphasize on the gains or losses captured by the relationships between participants in order to compute value. We define the expected profits $Ep_{ij}(T_N)$ of participant i due to its interaction with participant j to be the expected value of participant i in the next time interval $[T_N, T_{N+1}]$ increased (or decreased) by the percentage change of the expected satisfaction $ESAT_{ij}(T_N)$ in the next time interval and is given by:

$$Ep_{ij}(T_N)=(ESAT_{ij}(T_N)/ESAT_{ij}(T_{N-1}))(ER_{ij}(T_N)-EC_{ij}(T_N)), \tag{3}$$

where $ER_{ij}(T_N)$ and $EC_{ij}(T_N)$ are the expected revenues and costs respectively for the next time interval. Thus, the value $V_i(T_N)$ of participant i at the end of time interval $[T_{N-1},T_N]$ is the sum of its realized profits (revenues minus costs) and the expected profits that come from its relationships with all other participants. The total value of the network is the sum of the value of each participant.

C. The Mechanism for Value Calculation

In this subsection we present our value-based model that provides a mechanism to calculate value divided in various hierarchical levels. Fig. 1 (generated by iThink) shows the upper level of the hierarchy and visualizes the basic elements of our framework. We use the example of Section II to simplify our description. Each node represents a module that calculates the value of a participant. Arrows represent dependencies between modules. Each module encloses a sub-system that calculates the value of the module (second hierarchical level). Complex variables inside the module are presented as modules too. Fig. 2 shows the dealer's value calculation process. The green arrows show the impact a module has on another module (e.g., dealer's expected profits increase as dealer's revenues increase). The module dealer's cost in the third hierarchical level is depicted in Fig. 3.

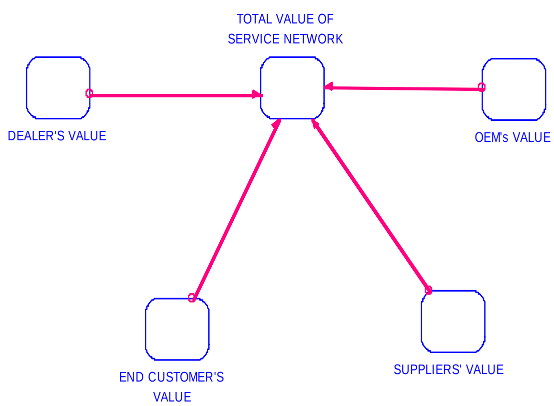


Figure 1. First hierarchical level of value mechanism.

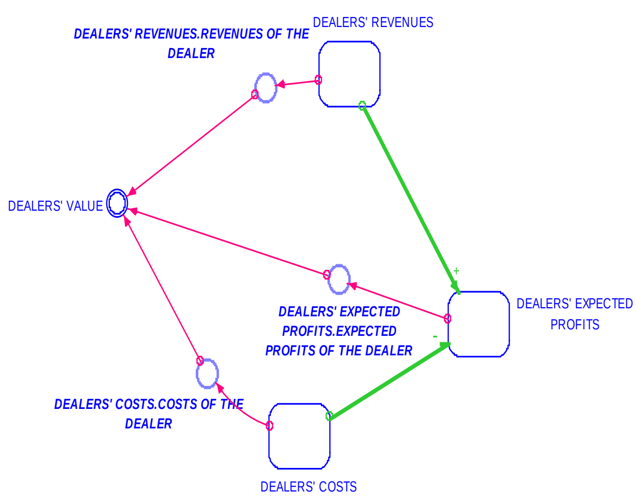


Figure 2. Second hierarchical level – dealer's value.

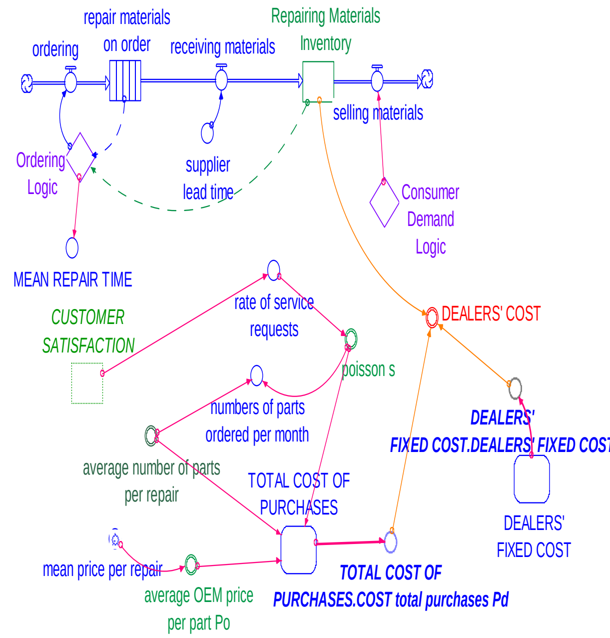


Figure 3. Third hierarchical level – dealer's cost.

IV. COMPETING SERVICE NETWORKS

We consider a complex system comprising two competing service networks A and B that provide the same service k. Our formalization captures the networks characteristics combining service-oriented economy with game theoretic tools.

We are interested in the case where network B is a new entry in the market of service k so that network A has already its own customers. Existing customers may move from network A to network B and vice-versa according to their satisfaction for their provider. New customers may choose a provider based on service price, service time and provider's reputation. We also assume that the two networks might share a common participant (e.g. they might have the same supplier).

Our model studies a two-player game where service providers make choices for prices and service times, have only one type of customers and compete for market share. We define a pure strategy s_i for provider i (i = A, B), the pair $s_i = (p_i, t_i)$, where p_i is the price for the service k and t_i is the service time needed to provide service k. Let S be the set of all pure strategies for each provider.

This game is different from the traditional economic models where providers are considered to be autonomous entities whose set of actions do not include interactions with others. The providers in our analysis are part of more complex systems whose actions affect other participants' actions that in turn affect them and so on.

In the absence of competition, a provider seeks to maximize its individual profits rather than the overall social welfare of the system it belongs to. In this case, the social welfare achieved by the provider's actions is less than the maximum social welfare achievable if a social planner were to select the provider strategies. In the model of competing providers (this is the case of oligopoly market and not perfect competition), the equilibrium strategies do not necessarily maximize social welfare as well.

Our objective is twofold. First, we seek to derive equilibrium strategies for the providers, given that all other participants of their own networks have revealed their actions. A Nash equilibrium is a pair of strategies, one for each provider, in which each provider is assumed to know the equilibrium strategies of his opponent, and no one benefits by changing only his own strategy unilaterally [16], [17]. In our game, $s^* = (s_1^*, s_2^*)$ is a Nash equilibrium in pure strategies if for each s_1, $s_2 \varepsilon$ S, value satisfies the inequalities:

$$V_A(s_1^*, s_2^*) \geq V_A(s_1, s_2^*), \quad (4)$$

$$V_B(s_1^*, s_2^*) \geq V_B(s_1^*, s_2), \quad (5)$$

where V denotes the value at a fixed time interval as given in Section III.

Second, we investigate the evolution of the two networks over time in terms of survivability and dominance in the market.

V. SIMULATION EXPERIMENTS FOR THE CAR REPAIR SERVICE NETWORK

In this section, we describe in short, simulation experiments (as performed in [1]) to measure the performance of a single service system. In Section VII, we design experiments involving two competing networks, in order to derive equilibrium strategies and observe the system's evolution over time.

We make use of 4 scenarios. First, we apply our approach to the car repair service system (Section II) to examine the network's evolution over time. We represent technicians, the parts manager, and the help desk experts as economic entities, each of which is offering its labor as a service to the service system. We measure rates of offerings and payment flows per month over a period of about 30 months. End customer service requests denoted by *s* are strongly affected by end customer satisfaction, since satisfied customers attract new customers to enter the network. Without loss of generality, we consider that the service requests are produced by the Poisson distribution with mean *es* being the output of the function:

$$es = -a_1 SAT^2 + a_2 SAT, \quad (6)$$

where $a_2 > 2a_1 > 0$ so that es is an increasing function of SAT in the range [0,1]. (We have chosen (6) because the rate of increase of *es* decreases with respect to SAT.) We also consider that the number of technicians is a function of the number of service requests; we take that the number of technicians increases linearly with the number of service requests. We calculate the value of each participant as a function of price and time and determine its optimal level with respect to price.

Second, we use the transformation of the basic model, as in [6], in order to cut costs and increase value. Specifically, a solution provider achieves interoperability between participants' information systems through application software operated by the OEM. The application allows everyone to have access to up-to-date information about parts at any time, as soon as this information becomes available to the data base of the application. The gain from the new IT infrastructure is twofold: repair time is reduced resulting in customer satisfaction increase and OEM's mailing costs are eliminated. We apply our methodology to the transformed network to show that the continuous changes of the environment push the network to restructure itself in order to remain competitive. We determine the time interval in which we observe positive effects in profitability in the transformed network compared to the initial one. We also determine which of the participants benefit from the transformation and which not.

Third, we consider a model in which the group of dealers is replaced by a new one that offers more complementarities to the end customers without increasing the mean repair price. This action seems to be profitable due to the increase of the satisfaction of the end customers of the service network. However new dealers have higher costs that may affect service network's value. We examine the value of these dealers and the value of the entire service network provided that OEM chooses to cooperate with them.

Fourth, we investigate Nash equilibrium strategies [16], [17] between OEM and the dealer. We define as a strategy for OEM and the dealers the mean profit rates a and b of selling parts and repair services respectively. Let p_s, p_0, p_d be the mean prices set by the suppliers, OEM and dealers respectively for offering their services. Then it holds that:

$$p_0 = p_s + ap_s = (1+a)p_s, \quad (7)$$

$$p_d = p_0 + bp_0 = (1+b)p_0. \quad (8)$$

We examine the existence of equilibrium strategies considering that the rest of the network participants (apart from OEM and the dealer) do not affect their decisions. We assume that OEM buys parts from certified suppliers at a given price p_s.

VI. RESULTS FOR THE PERFORMANCE OF THE NETWORK

In this section, we present the simulation results from our analysis. First, we compare the basic model with the transformed one.

A .Value Optimization in Basic and Transformed Network

We show the mean repair price p* that maximizes the dealers' and OEM's value in Table I.

TABLE I. COMPARISON BETWEEN THE BASIC AND THE TRANSFOMED NEWORK

Value	Model			
	Basic Network		***Transformed Network***	
p*	111 (dealer)	225 (OEM)	116 (dealer)	218 (OEM)
Dealer	51.469.012	34.700.000	46.874.332	34.985.000
OEM	$8500*10^6$	$26793*10^6$	$9100*10^6$	$29990*10^6$

We observe that:

- The dealers' optimal mean repair price in the basic service network is lower than in the transformed service network, since the mean repair time (that affects value) decreases, so the dealer charges his customers less. Consequently, the dealer is forced to increase the mean repair price in order to increase its revenues. Nevertheless, at the optimal mean repair price, dealers' value is less in the transformed network since the customer satisfaction has decreased as well (higher charges).
- OEM's value is much higher in the transformed network than in the basic one. This is explained by the fact that the mean repair time decreases and the customers are more satisfied (at OEM's optimal mean repair price). In addition, OEM in the transformed network has much lower mailing and labor costs.
- In both networks OEM's value at dealer's optimal mean repair price (111 and 116 respectively) is very low compared to OEM's value at his optimal mean repair price. This means that OEM will never be satisfied to offer its services at prices that reach dealer's optimal level.
- Dealers' value at OEM's optimal mean repair price is higher in the transformed network, since OEM's optimal price is lower (218).

Furthermore, the simulation results show that, OEM's value in the transformed network is not higher than that of the basic network from the first month. It dominates after 10-12 months, when both networks offer their final services at their optimal mean repair price (Fig. 4). When both networks offer their services at common prices in the range of 80 to 350, the transformed network dominates the basic network at month 8 to 17.

Finally, the total value of the transformed network (32.190.040.300) is maximized at mean repair price 216 and is higher than that of the basic network (28.593.400.000) which is maximized at mean repair price 223. This is explained by the fact that end customers are more satisfied and OEM (the keystone participant) has managed to cut costs at a great extend in the transformed network. Moreover, we see that the optimal mean repair price for both service networks is very close to the optimal mean repair price of OEM, since OEM contributes the largest part of the total value of the network.

B. Sensitivity Analysis of the Mean Repair Price

In this section, we investigate the impact of mean repair price changes to the dealers' value. As the mean repair price increases, the difference between the dealers' value in the basic network and that in the transformed network is smaller. This is justified by the fact that although the service requests decrease the mean repair price increases resulting in a decrease of the total value as shown in Fig. 5.

C. The Impact of New Entries

We call the network with the new group of dealers, the competitive network. We calculate values in the new scenario at mean repair price 216 which is the optimal price for the transformed network. We investigate the impact of the change of dealers letting the price unchanged so that the end customers are motivated to remain in the network. We show that dealers' value (31.527.812) is lower in the competitive network compared to the transformed one (35.481.031), since the new dealers' cost is higher due to the complementarities they offer. In addition, OEM's value increases (from 29.793.000.000 to 31.713.504.020) due to the increase of the service requests. The total value of the network increases from 32.190.040.300 to 32.792.529.000.

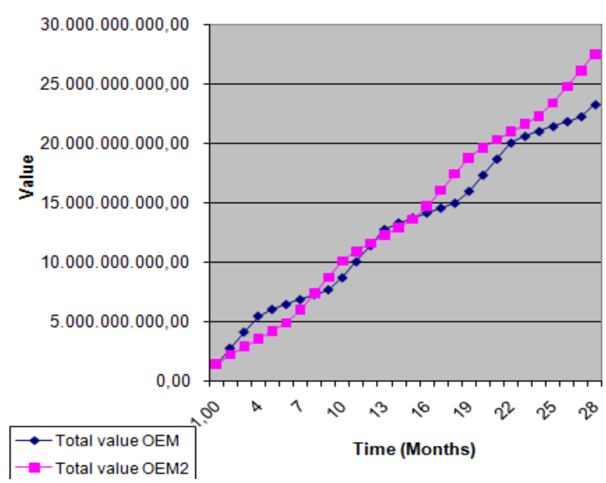


Figure 4. OEM's value in basic (1) and transformed (2) network at common mean repair prices.

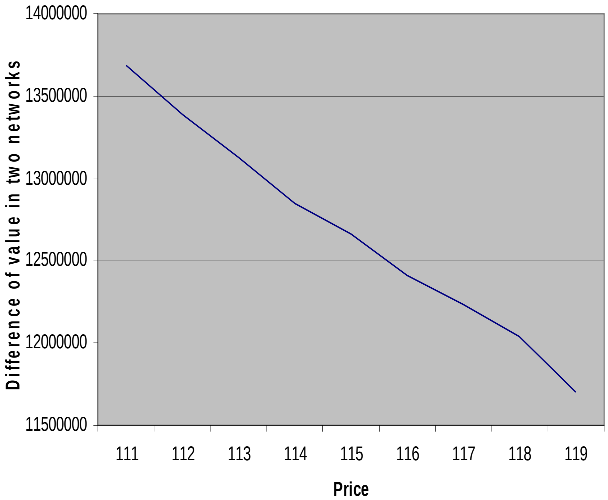


Figure 5. Dealers' difference of value in basic and transformed networks.

From the above we observe that a change in the network that improves its performance may affect positively some participants and negatively others. Naturally, dissatisfied participants abandon the network causing side effects to the others.

D. Participants' Equilibrium Strategies

We perform two experiments in order to investigate strategic interactions and determine equilibrium strategies of OEM and dealers. In the first experiment we calculate OEM's optimal profit rate at a given profit rate for the dealer. Simulations show that when the dealer increases its profit rate (e.g., from 6% to 10%), OEM's optimal choice is to decrease its optimal profit rate (from 24% to 21%). Conversely, if OEM increases its profit rates (e.g., from 14% to 21%), the dealer optimally decreases its profit rate (from 15% to 10%).

The second experiment calculates a set of equilibrium strategies for OEM and the dealer: at dealer's profit rate of 10% the optimal OEM's profit rate equals 21%. Conversely, at OEM's profit rate of 21% the optimal dealer's profit rate equals 10%.

VII. Simulation Experiments for the Model of Two Competing Service Networks

In Section V, we defined strategies involving two participants within the same repair service network (OEM and dealer). We figured out that their decisions were aligned so that value was increased for both of them.

In this section, we use the same model for value calculation to design experiments on a complex system of two competing networks; A and B as shown in Fig. 6. We consider that service network B has lately entered the market and serves a smaller portion of customers than service network A.

New customers are free to choose any of the two service networks to have their car repaired. More interestingly, customers may choose to abandon one network for the other when they are not satisfied from the services of their dealers. Even though OEMs are not directly connected to customers, their actions affect dealers' decisions which in turn affect customer satisfaction resulting in the restructuring of market share (Fig. 6).

We are interested in strategic behavior of the two OEMs provided that the other participants' and competitors' actions are kept fixed. Both OEMs seek to maximize their own value and achieve this by participating in a sequential game rather than solving the problem defined in Section IV in a one-shot game. At each time period, each OEM exploits information revealed by his opponent and makes decisions on which prices to charge dealers and which delivery times to complete dealers' orders for parts for the next time period.

We describe 4 scenarios defining dynamic strategies for the OEMs and show that after a small number of time slots these strategies reach an equilibrium in which no one is willing to diverge from its decision to his own benefit.

Let $v_i(t)$ be the value at time slot t, $n_i(t)$ be the number of ordered parts at time slot t, $p_i(t)$ the mean price per part at time slot t and $d_i(t)$ the mean delivery time at time slot t for OEM i.

In the first scenario, information revealed for each OEM in each time slot is the number of ordered parts of its opponent for the previous time slot. We consider symmetric strategies (same rule for both OEMs), where delivery time $d_i(t)$ is left fixed for the whole duration of the experiment and price $p_i(t)$ is changed according to the algorithm 1 shown in Fig. 7 (where i, j = A, B, $i \neq j$ and $\varepsilon > 0$).

As can be seen in Fig. 7, we use two criteria to determine the price for OEM i; we compare its value in two consecutive time slots and the number of orders with that of its competitor. At this point, we should mention that if we observe a loss in the value and a higher market share than our opponent, then we decide to increase price by a small increment since we can afford a small reduction in the

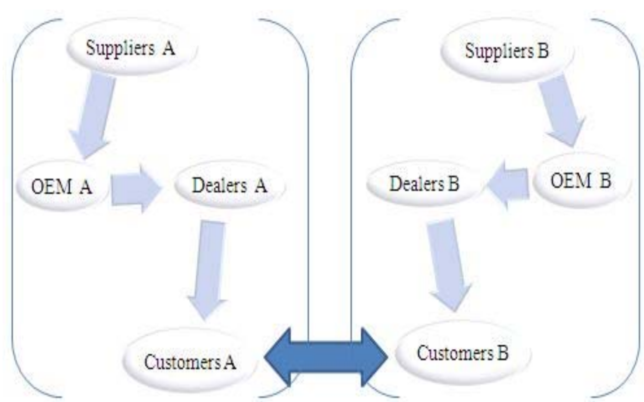


Figure 6. The competing networks interacting through their customers.

number of customers but we will gain more revenues aiming at a total increase of value.

In the second scenario, information revealed for each OEM in each time slot is the number of ordered parts of its opponent for the previous time slot and additionally for OEM of network B, the price of the previous time slot set by OEM of network A.

The strategy for each OEM is not symmetric (the two OEMs follow a different strategy), taking delivery time to be fixed for the whole duration of the experiment and price to be changed as shown in Fig. 8.

In this case, we change the order of the criteria placing the comparison of the number of orders first. Additionally, player B uses its opponent's former prices to gain competitive advantage when the performance of the network it participates is observed to be decreased.

In the third scenario, information revealed for each OEM in each time slot is the number of ordered parts of its opponent for the previous time slot. The strategy for OEM of network A is the same as in scenario 2 but OEM of network B follows a different strategy that is considered to be more risky since it takes into account the variability of mean delivery time (see Fig. 9).

In the fourth scenario, we study the behavior of the weaker competitor (network B) given that he first observes the behavior of his opponent. Again, the number of ordered parts in the two networks is common knowledge to both of them. OEM of network A (price leader) determines its own price following the same strategy as in scenario 1 (see Fig. 7). According to this price, OEM of network B (price follower) sets its price for the next time slot according to the rule given in Fig. 10.

We conduct simulations for the above scenarios and examine whether the strategies defined in each of them reach an equilibrium in the sequential game. In addition, we compare the values of the competitors and draw conclusions for their evolution over time in terms of dominance and survivability.

The time slot in all experiments is measured in months. We run simulations for 60 months and set the initial price for OEM A to be higher ($p_A(1) = 200$) than that of OEM B ($p_B(1) = 190$) since the new competitor needs to provide incentives to the forthcoming customers.

```
if  v_i(t) < v_i(t-1)
        if  n_i(t-1) < n_j(t-1)
                p_i(t) = p_i(t-1) – ε
        else
                p_i(t) = p_i(t-1) + ε
else
        p_i(t) = p_i(t-1)
```

Figure 7. Algorithm 1: strategy for OEM i in the first scenario.

```
STRATEGY FOR A

if  n_A(t-1) < n_B(t-1)
        if  v_A(t) < v_A(t-1)
                p_A(t) = p_A(t-1) – ε
        else
                p_A(t) = p_A(t-1)
else
        p_A(t) = p_A(t-1)

STRATEGY FOR B

if  n_B(t-1) < n_A(t-1)
        if  v_B(t) < v_B(t-1)
                p_B(t) = p_A(t-1) – ε
        else
                p_B(t) = p_B(t-1)
else
```

Figure 8. Algorithm 2: strategy for OEMsA and B in the second scenario.

```
if  n_B(t-1) < n_A(t-1)
        if  v_B(t) > v_B(t-1)
                p_B(t) = p_B(t-1) + ε
                d_B(t) = d_B(t-1) – ε
        else
                p_B(t) = p_B(t-1) – ε
                d_B(t) = d_B(t-1) + ε
else
        p_B(t) = p_B(t-1)
        d_B(t) = d_B(t-1)
```

Figure 9. Algorithm 3: strategy for OEM B in the third scenario.

VIII. Results For The Startegic Behavior of Competing Service Networks

The main results for the strategic behavior of the competing OEMs are presented in this section. In the first scenario, the value of OEM A decreases and the value of OEM B increases up to month 4, where equilibrium is reached as shown in Fig. 11. That is, neither OEM A nor OEM B are willing to change the derived prices followed by their strategies after month 4.

The intuition behind this result is that higher prices for OEM A imply higher service prices for dealers, thus, decreasing customer satisfaction. As a consequence, a considerable portion of market share has moved from OEM A to OEM B so that symmetric strategies have created networks of comparable size.

The second scenario in which the most recently set up network imitates its opponent's decisions in case of unstable situations, has similar results as the first one. Customers of network A join network B in presence of lower prices up to month 4 at which an equilibrium is reached (see Fig. 12). Despite the same shapes of value curves between the two networks under the two scenarios, the actual value level for each network is increased in the second scenario. This is due to the fact that the prices change more aggressively in the second scenario without implying losses in value.

In the third scenario, OEM B changes his strategy to encounter delivery time in addition to price. This has as a result an aggressive increase in customer satisfaction, since customers are given the opportunity to choose a relatively small increase in price at a shorter service time. This entails an increase in value for OEM B up to month 8 after which an equilibrium is reached. On the contrary, OEM A faces a loss in its value in the absence of time reductions (see Fig. 13). At month 4, the value of OEM B becomes higher compared to the value of OEM A showing that at this time slot the number of customers of the second network gets larger than that of the first one. The flexibility practiced by OEM B gave him the opportunity to change the balance of the market to its own benefit.

The simulations of the forth scenario verify our intuition that the leader (network A) faces the smaller loss in value (and consequently in the number of customers) compared to the other scenarios. This is explained by the fact that it is the first to choose a price and predict its opponent's choice that will be based on this price. As can be seen in Fig. 14, equilibrium is reached at month 6 with OEM A having a significantly larger value than that of OEM B.

```
if  v_B(t) < v_B(t-1)
        if  n_B(t-1) < n_A(t-1)
                p_B(t) = p_A(t) – ε
        else
                p_B(t) = p_A(t) + ε
else
        p_A(t) = p_A(t-1)
```

Figure 10. Algorithm 4: strategy for OEM B in the forth scenario.

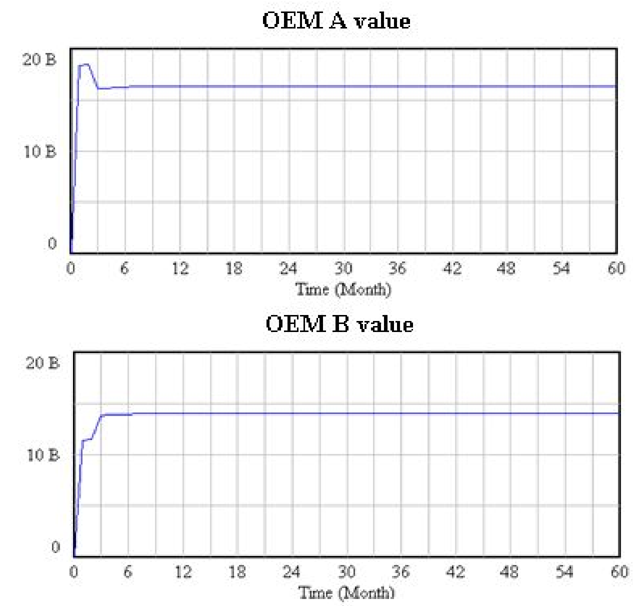


Figure 11. Value comparison in the first scenario.

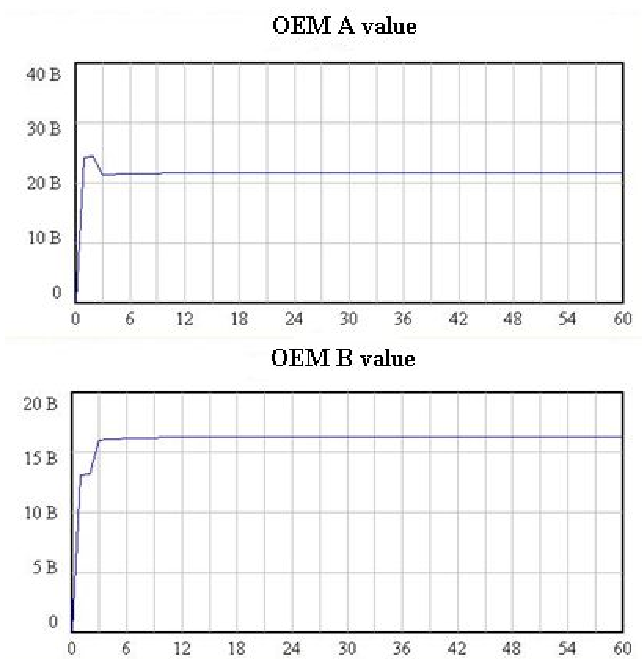


Figure 12. Value comparison in the second scenario.

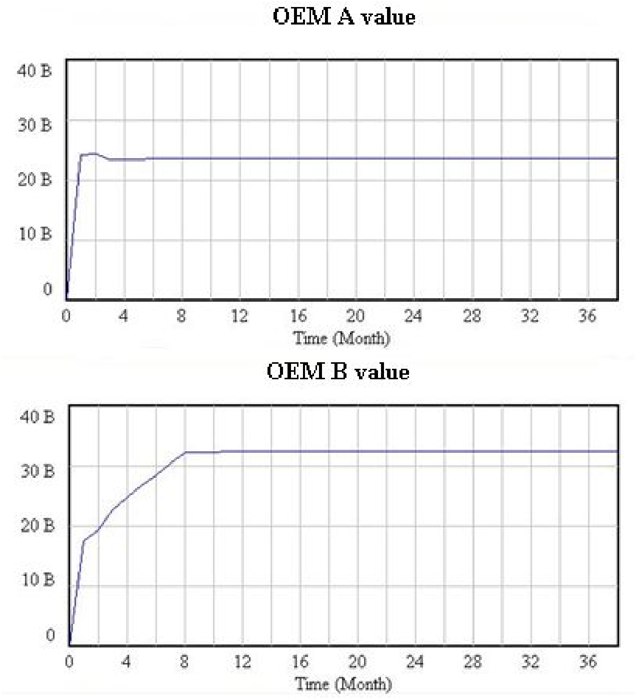


Figure 13. Value comparison in the third scenario.

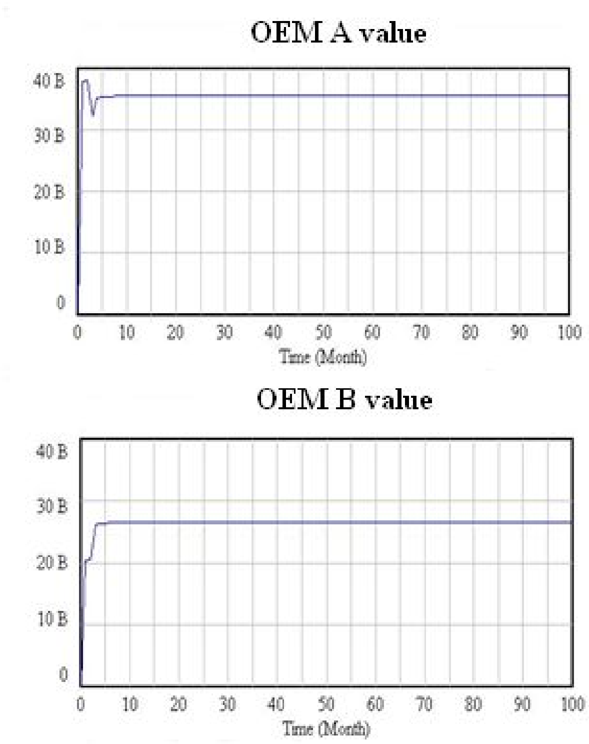


Figure 14. Value comparison in the forth scenario.

IX. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a methodology that evaluates the performance of service systems and analyzes strategic interactions between competing service networks. We applied this methodology to a car repair service network. We run simulation experiments to maximize the value of each participant and the total value of the network. In addition, we defined suitable scenarios to study the internal relationships that are developed inside the service network. We examined the interactions between the participants inside the service network in order to determine their optimal choices. We further designed simulations to investigate equilibrium strategies followed by leading participants of competing car repair service networks. We showed that in order to gain competitive advantage, a company has to align its objectives with those of the network it belongs to.

Directions for future work include the study of competitive service networks in which all participants take into account their rivals' and partners' strategies in order to calculate their optimal choice. Furthermore, additional work is needed on the estimation of value of intangible assets such as knowledge, sense of community, etc.

REFERENCES

[1] M. Voskakis, C. Nikolaou, Willem-Jan den Heuvel, and M. Bitsaki: "Servive network modelling and performance analysis". Proc. International Conference on Internet and Web Applications and Services (ICIW 2011), St, maarten, The Netherlands Antilles, 2011.

[2] J. Spohrer, P. Maglio, J. Bailey, and D. Gruhl, " Steps towards a science of service systems", Computer 40(1), pp. 71–77, 2007.

[3] V. Allee, "The future of knowledge: increasing prosperity through value networks", Butterworth-Heinemann, Boston, 2002.

[4] J. Gordijn and H. Akkermans, ''Designing and evaluating e-business models,'' IEEE Intelligent Systems 16, No. 4, 11–17, 2001.

[5] Ard-Pieter de Man: "The network economy: strategy, structure and management", Edward Elgar, USA, 2004.

[6] N.S. Caswell, C. Nikolaou, J. Sairamesh, M. Bitsaki, G.D. Koutras, and G. Iacovidis, "Estimating value in service systems", IBM System Journal, vol. 47, nr. 1, pp. 87-100, 2008.

[7] A. Biem and N. Caswell, "A value network model for strategic analysis", Proc. 41st Hawaii International Conference on System Sciences, 2008.

[8] V. Allee "Value network analysis and value conversion of tangible and intangible assets". Journal of Intellectual Capital , Volume 9, No. 1, 5-24, 2008.

[9] Clemens Van Dinther , Benjamin Blau , and Tobias Conte : "Strategic behavior in service networks under price and service level competition". Proc. International Conference on Business Informatics, 2009.

[10] Evangelos Katsamakas: "Competing value networks, incomplete contracts and IT". Proc. International Conference on System Sciences, 2005.

[11] C. Fornell, M.D. Johnson, E.W. Anderson, J. Cha, and Bryant B.E., "The American customer satisfaction index: nature, purpose, and findings", Journal of Marketing 60, 7–18, 1996.

[12] Jay W. Forrester, “Industrial Dynamics”, Pegasus Communications, 1961.

[13] “http://www.iseesystems.com/softwares/Business/IthinkSoftware.aspx”, January 2012.

[14] “http://www.vensim.com”, January 2012.

[15] “http://www.powersim.com”, January 2012.

[16] J.D. Sterman, “Business dynamics: systems thinking and modeling for a complex world”, McGraw-Hill, 2000.

[17] D. Fudenberg and J. Tirole, “Game theory”. MIT Press, 1991.

Constituting a Musical Sign Base through Score Analysis and Annotation

Véronique Sébastien, Didier Sébastien, Noël Conruyt
LIM - Laboratoire d'Informatique et de Mathématiques, EA2525
University of Reunion Island
Saint-Denis, La Réunion (FRANCE)
veronique.sebastien/didier.sebastien/noel.conruyt@univ-reunion.fr

***Abstract* - The recent progress in Information and Communication Technologies has given birth to advanced applications in the field of instrumental e-learning. However, most of these applications only propose a limited number of lessons on predetermined pieces, according to the vision of a single music expert. Thus, this article introduces a web platform to create music lessons dynamically and collaboratively, with the assistance of a semi-automatic score annotation module: @-MUSE. To do so, we first describe a new methodology to design such a platform: Sign Management. Then, we detail its general architecture as an Iterative Sign Base System based on a common practice in musical learning: score annotation. Lastly, we give various algorithms to generate relevant annotations on a score in order to explain it. These algorithms are based on the analysis of musical patterns difficulty. They are implemented within a module of @-MUSE called Score Analyzer. We present here its first results.**

Keywords - e-learning; knowledge management; sign management; multimedia; semantic web; musical score; music information retrieval; decision support

I. INTRODUCTION

Information and Communication Technology for Education (ICTE) expanded rapidly these last years. Indeed more and more teachers resort to platforms such as Moodle or Blackboard to design their own online courses. While this trend is being confirmed to learn academic disciplines such as mathematics and languages [10], it remains rare for know-how transmission and sharing, for instance in the field of music learning. Indeed, know-how transmission requires heavy multimedia usage and interaction to show the “correct gesture” and is thus complex to implement.

In music, some instrumental e-learning solutions exist in the form of offline tools, such as instructional DVDs (see the technical report of E-guitare [25]), or business software (Guitar Pro® [26], GarageBand® [27]). Nevertheless, getting a feedback from the teacher is capital in know-how acquisition: "is my gesture correct?". But few applications try to implement a learner to teacher communication axis through video upload and commentaries on the Web (see the FIGS [28] glosses system). Still, the lessons provided by these platforms remain limited to a fixed list of pieces. Although a student can suggest a new title, the realization of a whole lesson on these platforms requires heavy installations and treatments (multi-angle video recording, 3D motion capture), as well as the intervention of multiple actors other than the teacher himself. While these methods produce high quality teaching material, the realization of a new course remains a complex and expensive process. In parallel, several teachers, for instance retired experts, wish to transmit their know-how in a simple way, without any constraint on the recording location and time, and with minimal efforts for tool appropriation.

We thus introduce in this paper a complementary framework to rapidly create dynamic music lessons on new pieces with multimedia annotations [1]. This framework for music learning is called @-MUSE (Annotation platform for MUSical Education). As described in [18], an online annotation system is chosen because it allows musicians to work with digital scores in a way similar to traditional lessons, where scores are a support for memory and information sharing [24]. In addition, the digital transposition of this common practice enables to enrich it with multimedia incrustation, collaborative working and mobility. As such, its aim is also to constitute a scalable music playing sign base (see part II) to collect and share tips and performances on all possible artistic works referenced on music data warehouse such as MusicBrainz.org [29], and which can evolve according to the learners’ needs, whatever their level may be. Indeed, most of the existing music learning applications target beginners and do not provide an environment adapted to the teaching of advanced instrumental techniques. This sign base is generated with ISBS (Iterative Sign Base System), which aim is to define the structure of pieces with an ontology, describe them with a questionnaire, then capture interpretations with @-MUSE in order to preserve and share experts know-how. For the time being, the chosen field is music, and more precisely piano techniques, as it is a very demanding and historically rich domain. Still, the conceived system and its principles remain largely applicable to other instruments. Besides, as the authors are musicians themselves, it is easier for them to experiment and interact with professionals from this field. Indeed, the proposed methods and experimentations presented in what follows result from a collaboration with teachers and students from the Conservatory of Music of Reunion Island.

In this paper, we first introduce the methodology and principles of Sign Management that supports ISBS. Then, we describe the general architecture of @-MUSE, the ISBS annotation module designed to constitute a Musical Sign Base (MSB). To assist users into feeding and exploiting this base, we describe various methods to generate relevant annotations (i.e. explanations) on a score. These annotations are generated according to descriptive logics used by pianists when they study a new piece. This method is mainly based on extracting main parts of a piece and detecting its inherent difficulties. Therefore, after reviewing practical rules to structure a piano score, we present our algorithms to automatically analyze its playing difficulties. Lastly, we present Score Analyzer, a module of @-MUSE which implements the methods and algorithms we conceived to detect score and performance difficulties, and discuss its first results.

II. METHODOLOGY: SIGN MANAGEMENT

Sign Management deals with the management of know-how rather than knowledge. It manages live knowledge, i.e., subjective objects found in interpretations of real subjects (individuals) on the scene (live performances) rather than objective entities found in publications on the shelf (bookish knowledge). A Sign is a semiotic and dynamic Object issued from a Subject and composed of three parts, Data, Information and Knowledge. All these subjective components communicate together to build a chain of "sign-ifications" that we want to capture.

Sign management is thus more central than Knowledge management for our purpose in instrumental music learning. Indeed, the musical signs to treat are made of emotional content (performances), technical symbols (scores) and tacit knowledge (rational and cultural know-how). Thus, a Sign is the interpretation of an object by a subject at a given time and place, composed of a form (Information), content (Data) and a sense (Knowledge). The sign management process that we have created is made on a Creativity Platform for delivering an instrumental e-learning service [6][7][17]. It is founded on an imitation and explanation process for understanding gestures that produce a right and beautiful sound. The advantage for learners is that we are able to decompose the teacher's movement and understand the instructions that are behind the process of playing a piece of music. In fact, a lovely interpretation is made of a lot of technical and motivated details that the learner has to master, and the way we want to deliver this information is to show examples from experts through multimedia annotations indexed on the score. To do so, we introduce a new module to design dynamic music lessons through multimedia annotations: @-MUSE.

Indeed, as shown on Figure 1, an annotation can be considered as a structure including all three components of a sign: a symbolic form (the "written" document, which can be a score, or a tablature or lyrics in music), a substance (for example a video showing the musical performance) and a meaning (the explanation from the annotator point of view).

Annotations thus become a support to enable fruitful dialogs between users on @-MUSE. In order to let users compose lessons in a dynamic way, we propose the semantic architecture proposed in part III.

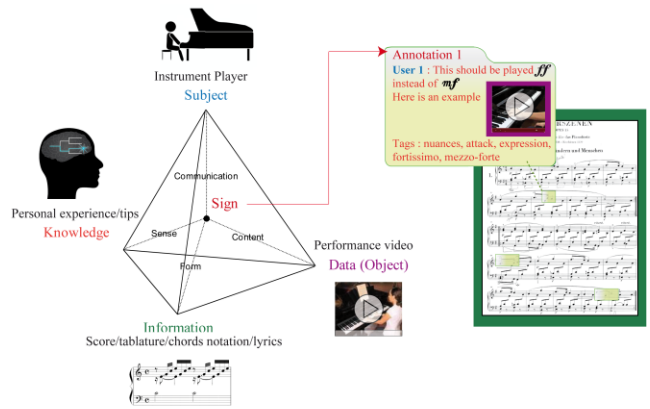


Figure 1. Sign management on @-MUSE

III. @-MUSE GLOBAL ARCHITECTURE

As the aim of @-MUSE is to enable dynamic teaching and learning through annotations, it is capital that its architecture remains flexible. The use of Semantic Web tools is thus an appropriate lead to allow the platform to benefit from a "networking effect". Indeed, a significant amount of scattered musical resources already exist on the Web and can be relevant in the context of music lessons. These resources can be music metadata (MusicBrainz.org), digital scores (images, PDF, MusicXML free or proprietary files available on Werner Icking Archive [30]), multimedia documents (video recordings of performances and lessons on YouTube [31] or eHow [32]) or simple textual comments. They constitute the different sign components listed in part II: data, information and knowledge. As many of these resources benefit from a Creative Commons License [33], they can be used in the context of a music lesson, complementary to high quality resources from a professional multimedia capture set [7]. Figure 2 exposes a comparison between architectures of traditional instrumental e-learning applications and @-MUSE. In the first case, lessons are defined in a static way. Each one corresponds to a musical piece, with its associated resources: video, audio and image files synchronized together to form the lesson. While this system produces complete instructions, it cannot establish relations between two distinct resources or pieces, which is an essential point when learning music as a whole. In the second case, @-MUSE dynamically creates lessons by linking related resources posted by users and presenting them through an adapted interface [18]. If a resource is not available (for instance, a logic representation of a score), the system still works with a temporary replacement (for instance a simple image representing the score) in the frame

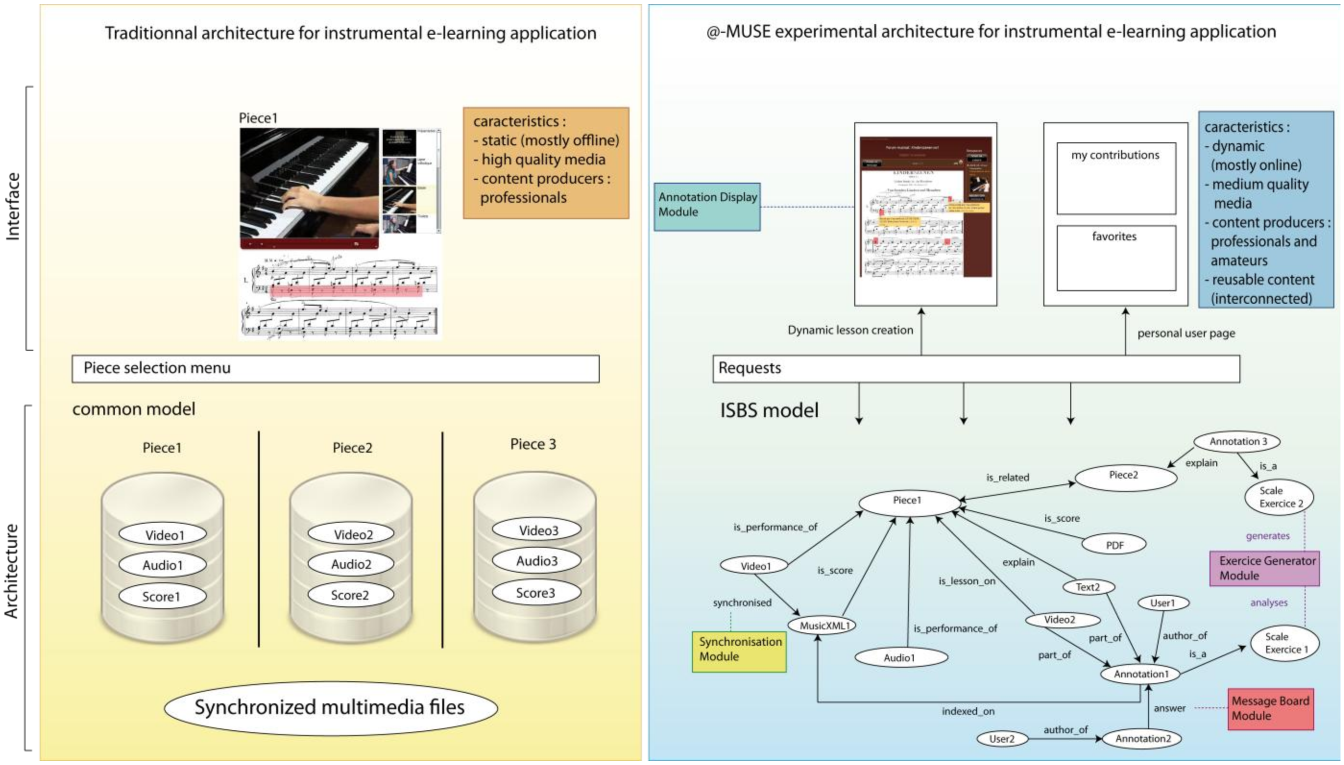


Figure 2. Architecture comparison between traditional instrumental e-learning application and @-MUSE

of a degraded mode. It can then point to any user the need to provide such resource to enable new functionalities on the platform. As more links are created between resources, different representations of the same piece can be proposed to learn how to play it. Some links such as a time synchronization between two representations (e.g. a video performance and a logical description of the score) can be realized by specific independent modules (Figure 2).

We have done previous work in [19] to propose an adapted ontology to link musical resources in an educational context using the Resource Description Framework (RDF [12]). This allows people to tag their annotations with appropriate concepts, such as “Technical exercise”, “Harmony”, “Fingering”, etc. The idea is to generalize existing annotations to include them on other relevant pieces. As such, a learner may start a new piece on his own, and still dispose of basic information, thanks to these semi-automatically generated annotations (see part IV.D).

In the end, the association of these elements will allow the creation of an Iterative Sign Base System for music, in the same vein as IKBS (Iterative Knowledge Base System [5]) for environmental data. The difference here lies in the manipulation of semiotic objects (signs), instead of conceptual ones (knowledge), as described in part II. The following chapter explains how new signs can be generated on this platform through semi-automatic score annotation, and thus participate in the enrichment of the MSB by demanding minimal efforts from the users of @-MUSE.

IV. KNOWLEDGE EXTRACTION ON DIGITAL SCORES

ISBS is a sign base model designed to collect musical signs such as scores (model) and performances (cases), in order to explain and compare them. A score can be considered as a database containing musical information. In this frame, score-mining represents the applications of data-mining methods to one or several digital scores. Thus, our aim is to infer knowledge by analyzing these particular data [23]. To realize such analysis in a semi-automatic way, we need to detect specific patterns within a score. This detection could be made directly on performances [21] but audio signal analysis algorithms are difficult to implement in a Web-application and may be less precise than those based on symbolic representations in an educational context. That is why we rely on XML representations of a score to design our inference system. MusicXML [3] is an XML open source format to describe digital scores staff by staff, measure by measure, and lastly note by note (Figure 3).

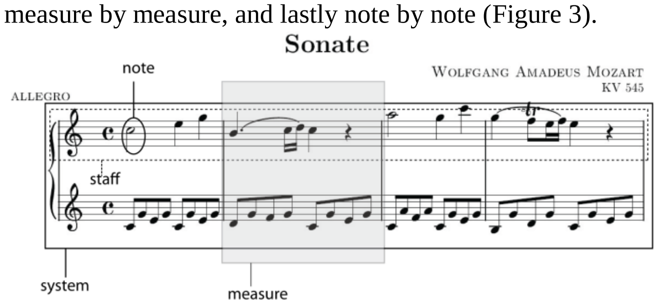


Figure 3. Score logical structure

In what follows, we review and propose different methods to extract various playing information from a piece metadata and structure, for educational purposes. We base these methods on how a pianist would address an unknown piece. As detailed in the descriptive model presented in [19], the musical work is first replaced in its context (composer, period, form metadata). Its global structure is then determined in order to visualize how the piece is shaped. This step may also be helpful to design a work schedule adapted to the piece. Then, its difficulty is evaluated, firstly globally (tempo, length), and then part by part, in order to determine what type of work can be made on this piece and where. The following parts present methods to set up these different steps in the frame of @-MUSE. Then, we propose several tracks to exploit the detected difficult parts to generate relevant annotations on the score. In the last part of this chapter, we present Score Analyzer: a module of @-MUSE designed to automatically determine the difficulty level of piano pieces, and discuss its results.

A. *Musical work context analysis*

The context of a musical work provides several pieces of information on how to play it. Information metadata such as its title or composer, present in the MusicXML file, allow to obtain more information about its genre. Indeed, music Web Services, such as MusicBrainz, Amazon, Last.fm, etc. query a piece by title and composer to identify it. This allows to extract metadata on a piece, for example a portrait and biography of its composer, or an indication about the piece style with a link to the corresponding Wikipedia page if it exists.

Several performances of the piece can also be requested on video sharing websites such as YouTube. Still it is important to evaluate the quality of these resources before posting it on educational platforms. Communities such as MuseScore.com [38] allow its users to choose the appropriate video by themselves and then synchronize it to their scores, thus insuring some reliability between content and form. All these additional multimedia resources contribute to the creation of a complete music lesson environment.

B. *Form and structure analysis*

To play a piece of music correctly, it is very important to grasp its structure. Indeed, the pianist's playing can change drastically from one part of the piece to another (especially on advanced pieces, which may contain several contrasted parts). Moreover, specific behaviors are often expected according to the encountered notes pattern. For example, a pianist is often taught to slow down at the end of the piece, to breathe between two phrases, or to augment the volume at the end of an ascendant arpeggio. For our purpose, structure detection is also important to refine annotations indexation. It allows musicians to annotate directly a phrase or a pattern, without having to indicate it as such beforehand. Moreover, it enables advanced searching on the pieces base (i.e. by musical pattern, by melody, by form [2]) and sensibly refines the automatic difficulty estimation.

However, the granularity scale to structure a musical work is large and complex, as it can go from whole pieces collections to simple notes. To guide users through a precise musical work structuring process, we propose the descriptive model shown in Figure 4 which follows some descriptive logics. We consider a *Musical Work* (or *Collection*) to be composed of several *Pieces* (for example a Sonata including three movements, i.e. three "sub-pieces"), each *Piece* being constituted of different *Parts* (for example, a theme and its developments). Each *Part* may contain several *Subparts* (recursive definition). A piece *Part* is a group of *Notes*, and can be designated as a "*Theme*", a "*Melody*", a "*Voice*", or a "*Phrase*" thanks to an appropriate Musical Form Taxonomy to create specialized part names. A *Part* may contain several repeated and/or remarkable *Patterns*, each one being composed of several *Notes* (or *Rests*). The descriptive model designed in this way can match any musical work, from the simplest to the most complex ones.

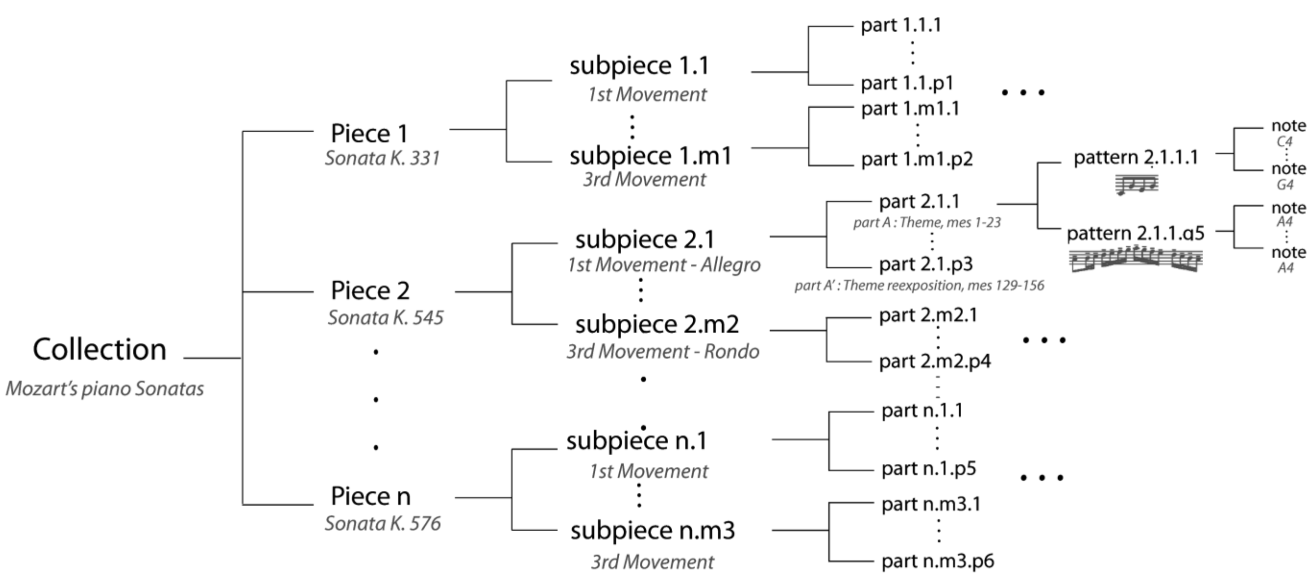


Figure 4. Musical piece structure descriptive model

To define the piece's structure as illustrated previously in a Semantic Web context, we need an appropriate Ontology. As Music Recommendation Systems are more and more popular, several works exist to tag musical content. For instance, the Music Ontology [14] allows to semantically describe pieces metadata. It is particularly fitted for the Music Industry and is largely used on web radios such as Last.fm [35] in order to describe *Artists*, *Albums* and *Tracks*. Concerning music structuring, it allows to tag a song in order to indicate its *Chorus* and *Verse* on a *TimeLine* [15]. This *TimeLine* can be synchronized with a score to obtain a symbolic decomposition of the piece. However, the Music Ontology does not provide a spectrum of structures large enough to be used in our frame (especially on classical music).

Its extension, the Symbolic Music Ontology [15], presents two interesting concepts for us: the *Voice* and the *Motif* (i.e. a pattern) concept, each designing a group of notes. Indeed, indicating voices on a score is an interesting feature in the frame of an educational platform such as @-MUSE, especially for polyphonic instruments. A *Motif* is a small group of notes which occurs repeatedly through the piece. In folk music, a *Motif* can be related to a type of dance by its rhythmic characteristics. For example, the siciliana gave its name to the eponymous rhythm pattern. As such, it constitutes an interesting annotation material.

The Neuma Ontology [16] is more specialized than the Music Ontology. It defines a *Fragment* entity allowing the separation of a piece into different parts, which can then be described thanks to a dedicated taxonomy. A *Fragment* has a *Beginning*, an *End*, both indicated as measure numbers, and several *Voices*. As Neuma is specialized in Gregorian music, it can define *Fragments* as *Phrases*, *Sub-Phrases*, or *Melody*. But to match a larger spectrum of musical pieces, more concepts have to be added, such as *Theme* and *Development* (necessary to describe Bach's Fugues for example), *Introduction, Variations, Coda*, etc.

To sum up, we dispose of various solutions to fragment a symbolic representation or a performance of a musical piece, but not to identify the created fragments appropriately, and thus uncover the form of the piece. Besides, The Music Ontology is already the base for several extensions related to music, such as the Chord Ontology, the Symbolic Music Ontology, or the Instrument Taxonomy [15]. Therefore, we intend to use the Music Ontology as a base to build a Musical Form Taxonomy in future works.

While users can enter this structuring information manually through score annotation, it is interesting to study how we can assist them on this operation through automated tools. Indeed, on a MusicXML score, several methods can be used to automatically detect and extract some of the parts described previously. The simplest method is based on symbols detection. Indeed, score symbols such as direction texts (e.g. "meno mosso"), tempo and key modifications, double bars generally indicate the beginning of a new part within the piece (Figure 5). This method gives acceptable results most of the time. Some exceptions may occur, especially on contemporary pieces, which often present unconventional structures.

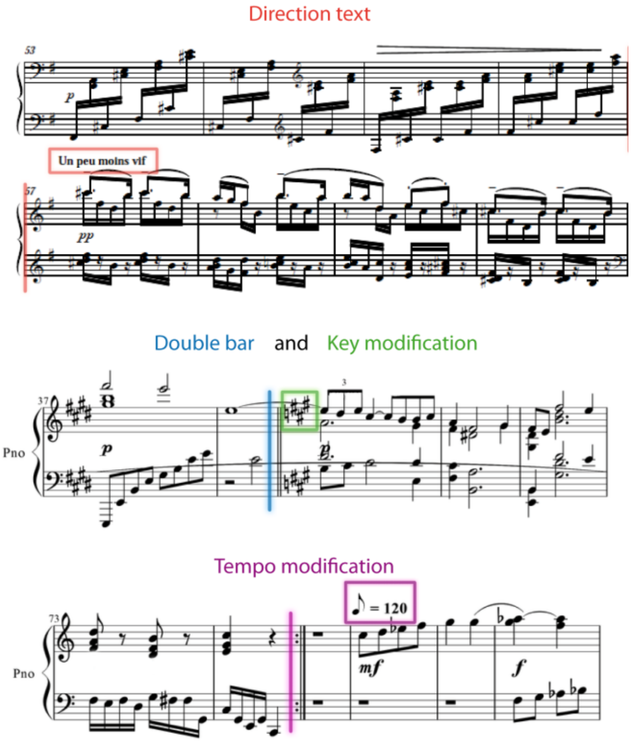


Figure 5. New parts detection on a score

The second method, more complex, is based on how musicologists and musicians generally cut a piece, according to melodic and harmonic features. The most representative harmonic feature marking the end of a musical phrase is called a cadence (characteristic chords sequence). Detecting cadences within a MusicXML consists in identifying specific harmonic sequences. Thanks to this method, we can identify more specific phrases.

For most tunes, repetitive patterns may be identified within phrases, sometimes with slight differences. For instance, Beethoven's Fifth Symphony starts off with the repetition of one of the most famous musical pattern (Figure 6). As the harmony evolves through the piece, the rhythm and the intervals of the pattern remain unchanged. Yet, pattern detection in music is much more complex than the given example, as it does not only involves rhythms and pitch features, but also polyphonic ones. Moreover, it does not present a fixed definition of "similarity", in opposition

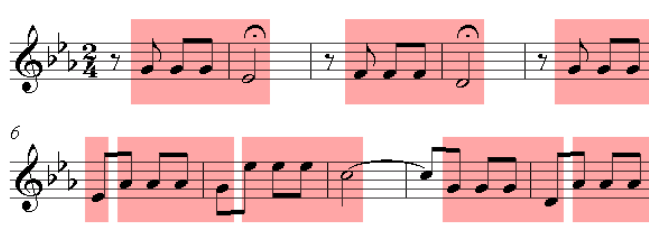
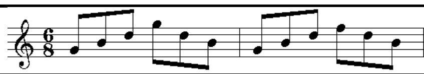


Figure 6. Musical patterns on a score (extract from the Fifth Symphony by Beethoven)

to text patterns detection. Two fragments can be considered as "similar", without having the same pitches (for example, the first two patterns in Figure 6). Several works exist on Musical Pattern Discovery [23]. Among them, [11] presents a method based on time windows and define different types of patterns (abstract patterns, prefixes, patterns network).

Discovering predefined patterns is easier if we know what features we are seeking for (mainly regular intervals). Table I gives some examples of *patterns*. They are not specific to a particular piece, as they can appear in any pieces through different sorts. For instance in jazz and classical music, *scales* are so common that they provide specific exercises collection [8]. In MusicXML files, using a memory window of successive intervals may identify these patterns. Identified sequences correspond to one of the pattern listed in Table I. For instance, arpeggios correspond to a sequence of thirds, scales to a sequence of ascendant or descendant seconds, and trills to a sequence of alternated ascendant and descendant seconds. As shown in part D, the identification of these patterns plays an important role in guiding a learner through annotations generation.

TABLE I. MUSICAL PATTERNS EXAMPLES

Pattern name	Example
Scale	
Arpeggio	
Trill	
Real sequence	

Melodies identification within polyphonic music can be linked to the problem of *Voices* detection. Sometimes, *Voices* are indicated in MusicXML files. Humans create them with a software (i.e. Finale® [36] or MuseScore® [37]). But most of the time, this indication is not given, and the *melody* should be extracted by identifying its characteristic features (form, length, occurrences within the piece, etc). Work is in progress concerning this issue.

C. Difficulty analysis

Once the general structure of the piece has been identified, we then analyze its technical difficulty, first globally, and then part by part. In Table II, we propose seven criteria affecting the level of a piece for piano and detail how they can be estimated from a MusicXML file. Globally, a piece difficulty depends on its tempo, its fingering, its required hand displacements, as well as its harmonic, rhythmic and polyphonic specificities. Of course, these various criteria affect each other in a complex manner. For example, hand displacement is strongly affected by fingering, as noted in Table II.

Indeed, among these seven criteria, fingering plays an important role. Several works present methods to automatically deduce fingering on a given musical extract for piano ([4][13][9]). Most of them are based on dynamic programming. All possible fingers combinations are generated and evaluated, thanks to cost functions. The latter are determined by kinematic considerations. Some functions, even consider the player's hand size to adjust its results, such as in [9]. Then, expensive (in term of effort) combinations are suppressed until only one remains, which will be displayed as the resulting fingering. While the result often differs from a fingering determined by a human expert, it remains largely playable and exploitable in the frame of an educational usage. However, few algorithms can process polyphonic extracts [9], and many other cases are ignored (i.e., left hand, finger substitutions, black and white keys alternation).

Even if more work is needed on this issue, the use of cost functions remain relevant as it is close from the process humans implicitly apply while working on a musical piece. That is why we extend this idea and create complementary criteria to design a piece difficulty analyzer for piano learning. For each criterion described in Table II, a score is calculated in percentage.

The speed of playing *P* is determined as a percentage of the speed of the fastest possible piece. This speed has been fixed to a tempo of 176 beats per minute for a quarter note, multiplied by a shortest note value of sixteen (sixteenth note value). This value *s* was estimated after a selection of piano pieces renowned for their fast tempi. To avoid insignificant short values (i.e. trills), only values occurring on more than 15% of the total number of notes are taken into account. As shown in Part E, this calculation gives results close to a pianist evaluation of a piece playing speed. To determine the proportion of difficult displacements, we first search all positions changing within the MusicXML file. Indeed, two successive note elements (<note>) in the file do not necessarily imply a new hand position as these notes can belong to the same chord (and thus be played at the same time), be tied, or be a rest (<note></rest>). Once we are sure that two successive <note> elements correspond to a hand movement, we estimate its realization cost. First, we calculate the length of the gap in semitones, then the time imposed to realize the movement and lastly, the required fingering. The fine tuning of these three parameters allow to get a precise estimation of the displacement degree of difficulty. Chords, alterations and irregular rhythms are detected through XML parsing (see Table II).

The piece difficulty level is thus the average rate of each criterion. Furthermore, some weighting coefficients can be affected to each criterion to reflect the particularities of the player. For instance, pianists who are really at ease with polyrhythm would not consider it a relevant factor, thus affecting it a 10% weight. However, we insist that the resulting difficulty rate should be interpreted with care and remains a simple approximation. As stated in [22], a nice performance is not a mere addition of criteria since it contains an important subjective part such as morphological or physical facilities, psychological attention or concentration, etc. Still, it proposes an interesting approximation of a piece level, especially for large scores databases such as Free-scores.com [39].

Although the presented criteria were modeled after piano players experience, they can be adapted to others instruments. For instance, the fingering criterion can be transposed to the guitar by switching cost functions, and

TABLE II. PLAYING DIFFICULTY CRITERIA IN PIANO PRACTICE

Performance difficulty criterion	Musicological definitions	Cost function definition	Examples	MusicXML implementation
Playing speed	Tempo: speed or pace of a musical piece. May be indicated by a word (ex: allegro) or by a value in BPM (Beats Per Minute) Pulsation: reference value indicated in the tempo: 𝅝 = 1, 𝅗𝅥 = 2, ♩ = 4, ♪ = 8, 𝅘𝅥𝅯 = 16, etc.	$\boxed{\text{Playing speed} = \frac{\text{tempo} \times \text{shortest value}}{176 * 16}}$ With all tempi using a quarter note ♩ as a reference Unit: percentage of quickest playable piece (fixed at 176 𝅘𝅥𝅯)	*P1*: tempo = 120 ♪ = 60 ♩ Shortest value = 𝅘𝅥𝅯 $P1\ playing\ speed = \frac{60 \times 16}{176 \times 16} = 34\ \%$ *P2*: tempo = 120 ♩ Shortest value = 𝅘𝅥𝅯 $P2\ playing\ speed = \frac{120 \times 16}{176 \times 16} = 68\ \%$	<note><type> elements Tempo attribute in <sound> element
Fingering	Fingering: choice of finger and hand position on various instruments. Different notations exist according to the instrument. (ex: in piano: 1 = thumb, 2 = index finger, 3 = middle finger, etc.) See [4][9][13][20] for more detail.	If *m1*, *m2*, ..., *mn* represent the measures of a given piece *P*, $\boxed{\text{Fingering\_difficulty(P)} = \sum_{i=1}^{i=n} \text{Fingering\_cost}(m_i) > 50}$	*P* =  Fingering_cost(m_1) = 10 Fingering_cost(m_2) = 0 Fingering_cost(m_3) = 70 Fingering_difficulty(*P*) = 70	<measure> and <note> elements
Hand Displacement	Interval: pitch distance between two notes, in semitones. A hand displacement is considered difficult when two successive positions are spaced by more than 12 semitones (7 if played by close fingers on the same hand) within a short time interval. The displacement cost of an interval increases with its gap length	If *P* is a piece containing *n* displacements, and among them *s* difficult displacements (*s*<*n*), $\boxed{\text{Displacement\_difficulty(P)} = \frac{s}{n}}$	*P* =  Displacement_difficulty(*P*) = 7 / 17 = 41 %	Combined <note> elements where <pitch> gap > 12. Associated fingering file.
Polyphony	Chord: aggregate of musical pitches sounded simultaneously.	Proportion of chords and chords sequences in the piece	*P* =  Chords_proportion(*P*) = 6/16 = 38%	<chord> element
Harmony	Tonality: system of music in which specific hierarchical pitch relationships are based on a key "center", or tonic. Various tonalities impose various sharps and flats as a key signature. The most basic ones (no alteration) are A minor and C major.	Proportion of altered notes	*P* =  Altered_notes_proportion(*P*) = 3/25 = 12%	<alter> and <accidental> elements
Irregular Rhythm	Polyrhythm: simultaneous sounding of two or more independent rhythms. Example: synchronizing a triplets over duplets	Proportion of remarkable polyrhythm patterns (Time reference = pulsation)	 *P* = Polyrhythm_proportion(*P*) = 4/4 = 100%	<time-modification> element
Length	The length of the piece in beats. NB: the number of pages cannot really reflect the length of a piece because of page setting parameters	Number of measures × number of beats per measure.	*P* =  Length(*P*) = 3*3 = 9	<beats> element of <time> element and <measure> elements

hand displacements by adapting the gap threshold according to a representative set of guitarists.

In the following part, we study how the criteria of Table II can be used to generate relevant annotations on difficult parts of a new piece added on our @-MUSE platform.

D. *Semi-automatic annotation generation*

The previous form, structure and difficulty analysis can be merged to serve as a basis for the next chain of knowledge extraction on the given piece. Indeed, the criteria proposed previously can also be used on fractions of the piece.As such, for a given part, if one of the rate is abnormally high, we can infer that it presents a characteristic difficulty, and thus recommend an appropriate technical exercise to the player (Figure 7).

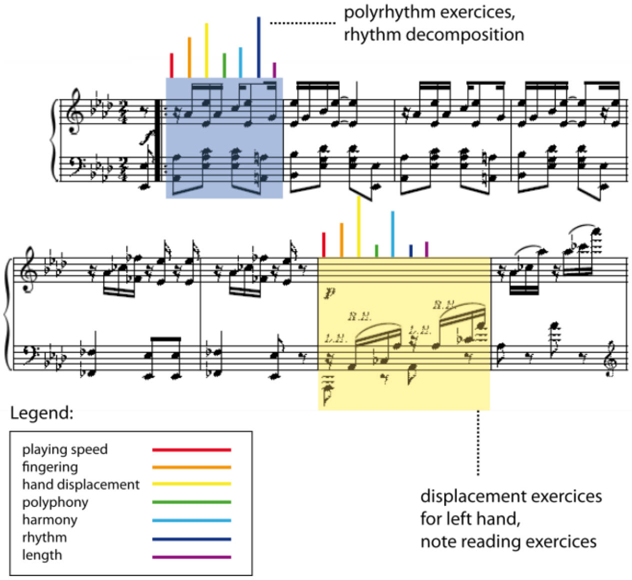


Figure 7. Difficulty analysis and recommendations on a digital score

Identified patterns are associated to corresponding exercises to guide the learner. In this case, most musical exercises consists in decomposition and repetition of subparts of the pattern. For instance, in the case of an arpeggio, the latter will be extended to the whole keyboard and repeated part by part, by adding a new note every ten repetitions. This process can easily be computed as suggested by Figure 9. These exercises can be adapted to the occurrence of the pattern by observing its tonal and rhythmic features.

But at anytime, the annotation's owner and teachers can modify it in order to improve the given explanation with textual and video commentaries, symbols and tags. Users can also invalidate the generated annotation if they consider it as being inappropriate. In this case, the motive for the suppression should be specified. This data will be later used to determine the reasoning error in order to improve the next generated annotations. All in all, automatically generated annotations should be clearly stated as such to users, for example with different colors, to avoid any confusion between public and private knowledge, validated or not by experts, annotations from users with unidentified level, computed or personalized knowledge. That is why, at each step of investigation (structure, difficulty, similar annotations retrieval), a degree of certainty is calculated in order to indicate to the user the reliability of the resulting annotation. Users can then choose to only display annotations which exceed 90% of certainty, or which have been approved by a teacher.

Indeed, filters play an important role on @-MUSE to insure an appropriate personalization of the annotation platform according to each registered user. Figure 8 illustrates this filter system. Thus, users can constitute their own libraries of personalized scores, which are instances of the original score containing all created annotations.

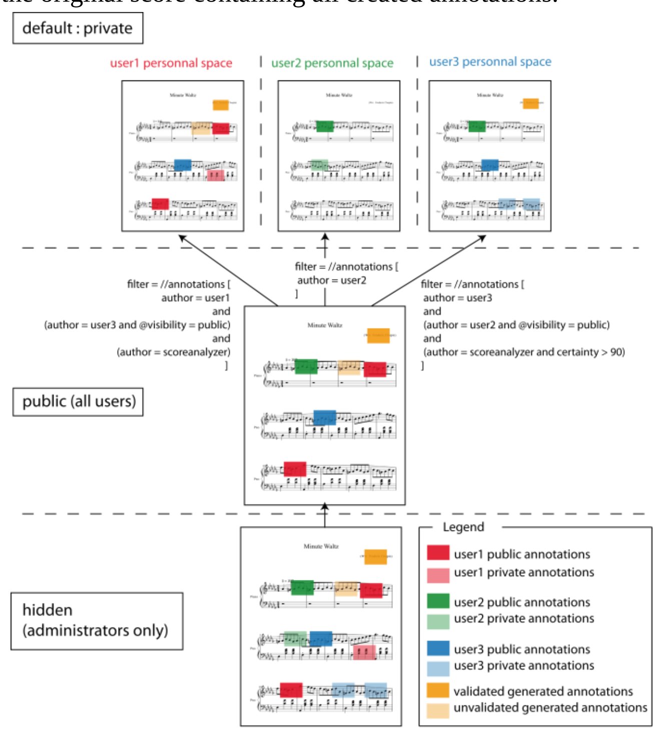


Figure 8. @-MUSE annotations filter system

Inferred playing knowledge can also be used to suggest new pieces to a musician, by analyzing the pieces which are present in his library. Identified criteria for piece learning recommendation are:

- The taste of the learner: if a user profile presents a tendency to play mostly one specific genre (e.g. classical period), two propositions can be made: either play another piece of the same style (not recommended by teachers, as in an academic curriculum, addressing all genres is necessary), or either suggest to discover a modern style one.

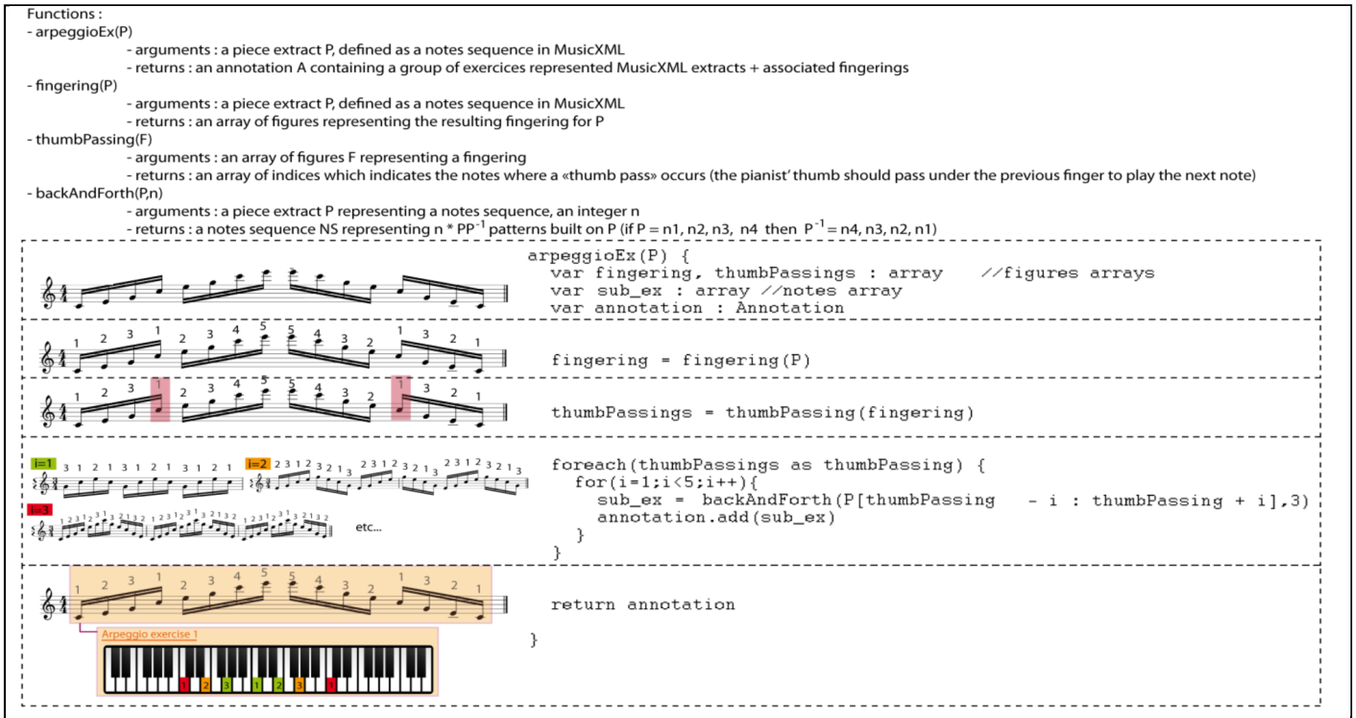


Figure 9. Pseudocode algorithm generating progressive arpeggio exercises

- The difficulties encountered by the learner: studies (e.g. Chopin's Études) constitute recognized works to overcome characteristic difficulties. Once these works have been correctly tagged, they can serve as suggestions to allow students to progress on the identified points. To identify difficulties encountered by a student, we analyze annotations created by this student, or which concern his recordings. To go further, systems which directly analyze performances in real time, such as Apple™ GarageBand® music learning module, need to be studied on advanced pieces.

In the next part, we present an implementation of the algorithms we proposed, in the form of an @-MUSE module called Score Analyzer.

E. Score Analyzer

The criteria presented in the previous sections have been implemented in a Web application called Score Analyzer [40]. This module is integrated to @-MUSE as a Web service in order to automatically evaluate a piece level and identify its difficult parts and advice apprentice musicians on their technical points.

Score Analyzer's engine takes any well-formed MusicXML file as input and parses it to extract knowledge exploitable from a performer point of view. For the time being, it targets pianists, but its main frame and several of its algorithms can directly be applied to other instruments. Following the scheme we detailed previously, the context of the piece is briefly analyzed (title, composer) and a few statistics are displayed (Figure 10). Then, main parts of the piece are identified, and lastly, difficulty estimations are given for each criteria identified in part C. At the time this paper is written, work is still in progress to display the results directly on the score under the form of annotations, in order to enhance the user experience.

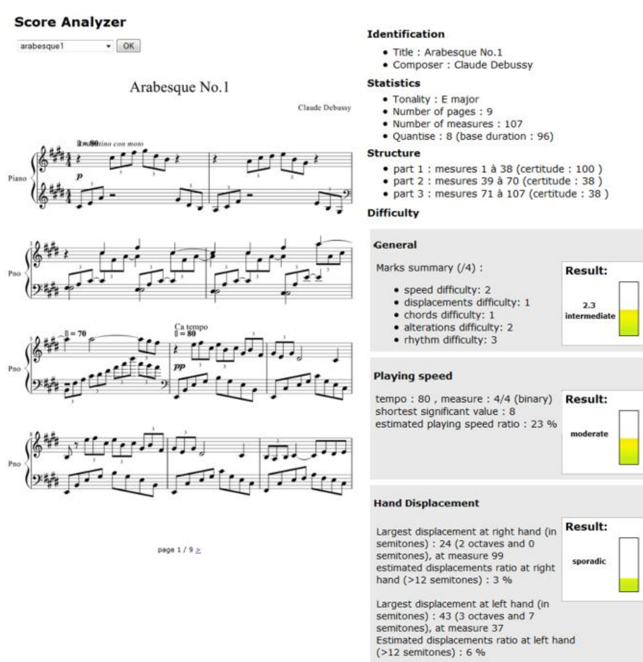


Figure 10. Score Analyzer's interface

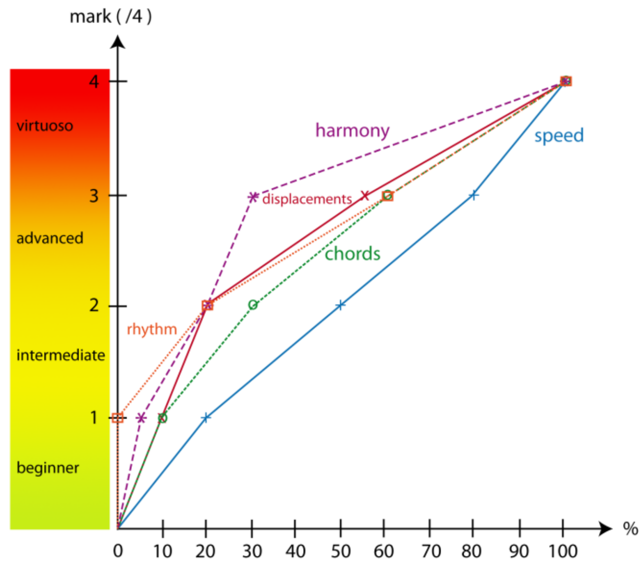


Figure 11. Evolution of marks in function of percentages for each criterion

To make the results more readable from a user point of view, percentages output from the formula given in Table II were replaced by marks, from 1 (beginner) to 4 (virtuoso), expressing the estimated levels of difficulty. Figure 11 details the defined slots for each criterion. These curves were calibrated on a sample of piano pieces commonly used in music schools and representative of various classical genres and levels. For each criterion, we dispose of at least one piece known to maximize its result. As such, the larger the spectrum of calibration pieces is, the more reliable the results are. Indeed, most of the criteria do not have a linear distribution (Figure 11), which constitutes a pianistic reality. For example, chords presence is considered as high starting from 60% occurrences on a given piece, as pieces constituted of only chords remain seldom cases. Therefore, most pieces concentrate around the center of Figure 11 graph. Easy ones occupy its left down corner, and difficult ones its top right one.

The synchronization between both hands is also taken into account. For instance, if each hand obtains a mark of 2 for the displacements criterion, then the global difficulty mark for this criterion will be 3, as playing with both hands will create an additional difficulty.

The protocol to evaluate the accuracy of our system simply consists in comparing results from Score Analyzer and pianists estimations. To do so, we use two distinct sources. The first one consists in difficulty estimations from the Free-scores online music community [39]. These estimations result from user comments and thus correspond to the experience of a large population of musicians. The second one consists in a precise evaluation of each piece (under the form of a questionnaire) by two experimented piano teachers from Reunion Island Conservatory of Music. This second source favors a qualitative approach. On Figure 12, we give the results of this experimentation on a corpus of ten representative piano pieces. This corpus was elaborated to cover a large range of genres and levels, and is regularly studied in music schools.

The main result of this first evaluation is that the estimations provided by Score Analyzer globally correspond to those of the majority of musicians. As such, its usability within an annotation platform containing a large collection of scores such as @-MUSE is relevant. Punctually, some gaps may be noticed between Score Analyzer results and human evaluations, as seen on Figure 12 for Bach's Invention n°1. In this case, this is due to the absence of counterpoint evaluation, which is one of Bach's

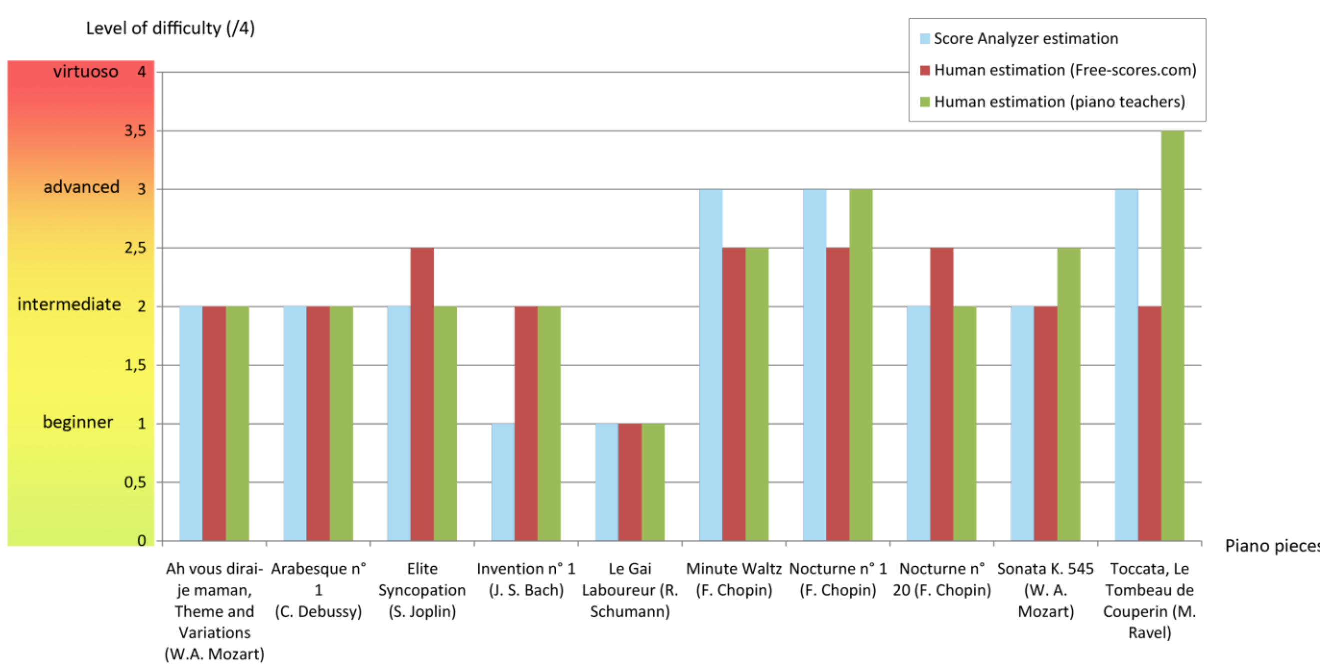


Figure 12. Comparison of difficulty estimations on ten piano pieces

characteristic and which can be particularly tricky to carry out for beginners. To implement this feature, more work need to be done on voices detection (see Part B). We also notice that this protocol induces a few biases. The first one is the lack of estimations for some pieces, thus reducing the objectiveness of the difficulty assessment. Typically, the Toccata from Ravel is clearly an advanced/virtuoso piece, but it was inappropriately marked by the only user who commented it. Hence the gap we can note on Figure 12. This type of cases points up Score Analyzer's interest on new untreated pieces. The second one is the users level. The population of Free-scores community is very heterogeneous, and as such, some users comments are only valid for their level, which is not always appropriate to approach the chosen piece (i.e. a beginner comments an intermediate piece as advanced for his level, and vice versa).

Of course, these biases may correspond to real experiences from users, as each musician approaches a piece differently, with his own skill, culture, feelings and motivation. In this frame, the final purpose of Score Analyzer is to provide an objective advice by informing a user if he chose a piece too difficult for him (a common case in musical education, but also a motive to progress), suggesting appropriate pieces to progress, and guiding him through the sight-reading of new pieces, by indicating difficult parts. On scores databases, Score Analyzer's results could be pointed up when the difficulty level of a piece has not been entered or do not present enough estimations to be relevant.

Thus, to ensure a more reliable human evaluation, we also questioned piano teachers (Figure 12). Most of their assessments correspond to Free-scores and Score Analyzer ones. However, we notice that Score Analyzer results correspond more to teachers' estimations rather than to Free-scores community ones, thus confirming the relevance of our criteria. As such, we dispose of a quantitative validation (lot of answers, less reliability), as well as a qualitative one (few answers, high reliability) on the evaluation of the global difficulty of a piano piece by Score Analyzer.

To go further into details, we also evaluate the quality of the calculation for each criterion. Indeed, our purpose is not only to indicate the level of difficulty of the piece, but also to find in what it is difficult (or not). To do so, we once again compare Score Analyzer results to pianists' assessments. Figure 13 presents an example of such an evaluation on three pieces of different levels (easy, intermediate and advanced).

Despite a few special cases, the estimations globally match (gap $\leq$ 0.5). This experimentation underlines the slight underrating of the Fingering and Rhythm criteria by Score Analyzer. Indeed, teachers evaluate these parameters with more factors than Score Analyzer does for the time being. For instance, rhythms difficulties do not include only the occurrence of many awkward rhythmic patterns in the piece, but also the required stability (for example, the multiple notes repetitions in the Toccata from Ravel) and strictness. Some of these parameters cannot be computed for

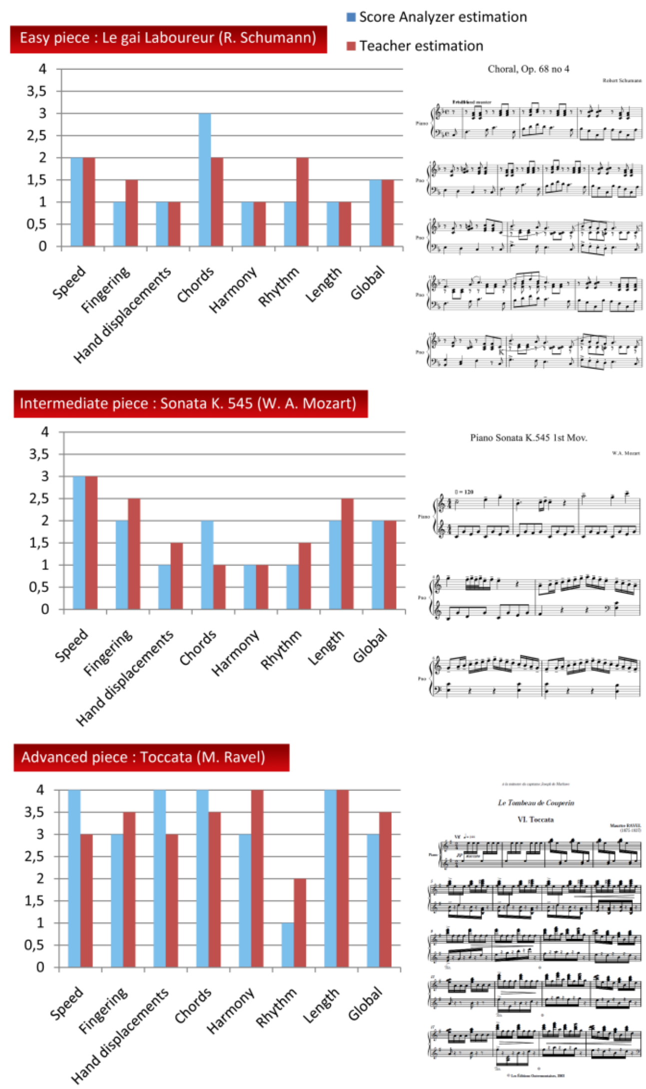


Figure 13. Comparison of difficulty estimations per criteria on three piano pieces

the moment, either because of their nature (expression depiction difficulty), or their complexity (specific patterns dependence). Thus, these first results also give us leads to enhance our calculations.

Working with musicians enabled us to confirm Score Analyzer's first result, but also to raise its main limit concerning high-level works: musicality consideration. Indeed, any Music Information Retrieval (MIR) system is limited as it can only consider processable data (audio/video signal, notes, tempo, text). As for expression and feelings, this remains an issue only noticeable, in multiple ways, to humans. Still, some leads about how a piece should "sound" could be suggested to beginners by analyzing styles, composers, direction texts and nuances, as well as previous annotations on similar pieces. This would constitute an interesting and challenging perspective for our platform.

V. Conclusion and perspectives

In this paper, we have proposed a methodology (Sign Management), a model (Iterative Sign Base System) and some inference methods (score-mining) to build an instrumental e-learning platform called @-MUSE. This platform allows teachers and learners to create music lessons dynamically with the assistance of a semi-automatic pieces annotator. These lessons can evolve according to the users' needs by submitting contextual exercises to them, in the form of multimedia annotations. These exercises are generated from the original score based on the identification of remarkable parts and their playability. Users can then give their point of view on the generated annotations but also add new ones, thanks to a dedicated symbols library as well as a multimedia capture module. The more knowledge is created on the platform, the more detailed the lessons will be, thanks to the emerging network effect resulting from the semantic linking of the various resources.

To generate relevant annotations, we have particularly insisted on the importance of finding difficulties within a score. To do so, we have presented Score Analyzer, a module of @-MUSE enabling automatic evaluation of piano pieces difficulty. Score Analyzer's first results have been presented and validated by confronting them to pianists' assessments.

Different perspectives are considered for this work. Concerning Score Analyzer, the presented experimentations suggests several leads to enhance the difficulty estimation, the main one being a further analysis of the genre and composer of the piece to better study the adapted playing style. Once again, this requires a close collaboration with professional music teachers and musicologists. Of course, a larger MusicXML pieces base would also allow to improve our criteria. We also intend to study in detail their applicability to other instruments and types of performances (chamber music, orchestration, etc). But what really constitutes the next challenge in this project is to distinguish what type of expressive knowledge can be automatically explicitated on a score. Indeed, extracting purely expressive features (emotion, intensity, rubato) from a score remains a tough task, as it rarely includes the basic information to do so. Moreover, imposing "rules" in musicality is a delicate task, as it can lead to conformism. The method we recommend is thus to analyze high level pre-annotated scores and research implicit rules based on the genre of the considered work (for instance, in classical music, it is conventional to soften the end of a phrase). Elements of fuzzy logic would then allow us to balance the relevance of an "expressive" annotation according to the context of the piece.

As for @-MUSE development, our ongoing work is to deliver an interface adapted to tablet devices (Figure 14), which would allow to use our platform directly in front of the instrument, guaranteeing an experience close to a traditional music lesson. Once these modules are merged, the @-MUSE project will give birth to a real e-community dedicated to music practice, and not only to music consumption. As such, the collaborative aspects of such a platform need to be studied to approach music learning under an entertaining angle, for instance by proposing specific group performances (Global Sessions [34]) and game features. Finally, as implied by our platform's name, learning music should first and foremost be a pleasure.

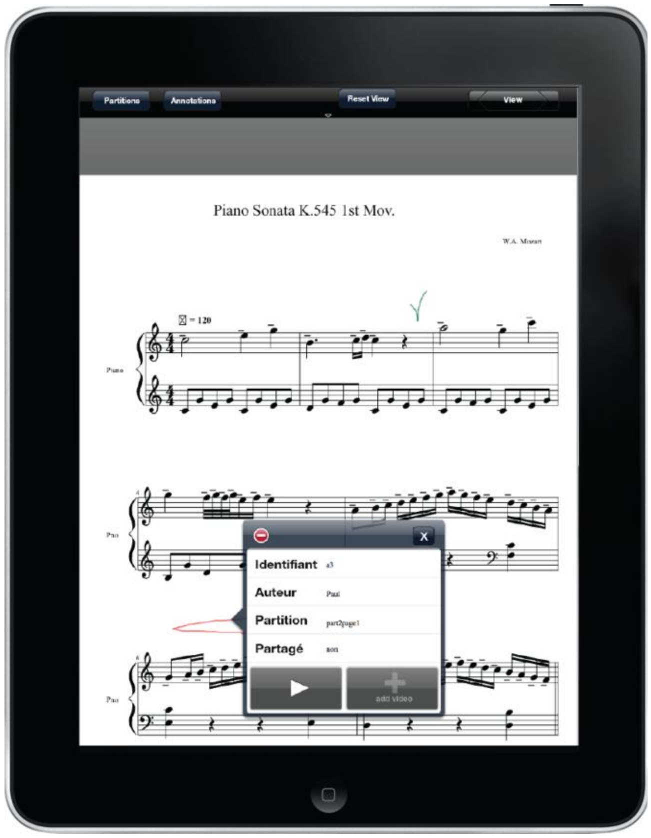


Figure 14. @-MUSE tactile interface

Acknowledgment

The authors thank Marie-Claude Equoy and Maïté Cazaubon (piano teachers at Reunion Island Conservatory of Music) for their expertise on piano education issues, as well as Paul Sébastien (trainee at the University of Reunion Island) for the development and test of @-MUSE tactile interface.

References

[1] V. Sébastien, D. Sébastien, N. Conruyt, "Dynamic Music Lessons on a Collaborative Score Annotation Platform", The Sixth International Conference on Internet and Web Applications and Services , ICIW 2011, St. Maarten, Netherlands Antilles, 2011, pp. 178-183.

[2] L. Abrouk, H. Audeon, N. Cullot, C. Davy-Rigaux, Z. Faget, D. Gross-Amblard, H. Lee, P. Rigaux, A. Tacaille, E. Gavignet, and V. Thion-Goasdoue. The Neuma Project: Towards On-line Music Score Libraries. In Intl. Workshop on Exploring Musical Information Spaces, WEMIS'09, 2009.

[3] G. Castan, M. Good, and P. Roland, "Extensible Markup Language (XML) for Music Applications: An Introduction", The Virtual Score: Representation, Retrieval, Restoration, MIT Press, Cambridge, MA, pp. 95-102, 2001.

[4] C.-C. Lin, “An Intelligent Virtual Piano Tutor”, National Chung Cheng University 2006.

[5] N. Conruyt and D. Grosser, “Knowledge management in environmental sciences with IKBS: application to Systematics of Corals of the Mascarene Archipelago”, Selected Contributions in Data Analysis and Classification, Series: Studies in Classification, Data Analysis, and Knowledge Organization, Springer, ISBN: 978-3-540-73558-8, 2007, pp. 333-344.

[6] N. Conruyt, O. Sébastien, V. Sébastien, D. Sébastien, D. Grosser, S. Caldéroni, D. Hoarau, and P. Sida, “From Knowledge to Sign Management on a Creativity Platform, Application to Instrumental E-learning”, 4th IEEE International Conference on Digital Ecosystems and Technologies , DEST 2010, IEEE Press, 2010, pp. 367-374.

[7] N. Conruyt, O. Sébastien, and V. Sébastien, “Living Lab in practice: the case of Reunion Creativity Platform for Instrumental e-Learning”, 13th International Conference on Interactive Computer Aided Learning, ICL 2010, September 15-17, Hasselt, Belgium, 2010.

[8] C. L. Hanon, “The Virtuoso Pianist in Sixty Exercises”, 1873.

[9] A. Al Kasimi, E. Nichols, and C. Raphael, “A simple algorithm for automatic generation of polyphonic piano fingerings”, 8th International Conference on Music Information Retrieval, September 23rd-27th, Vienna, Austria, 2007.

[10] K.J. Kim and C.J. Bonk, “The Future of Online Teaching and Learning in Higher Education”, Educause Quarterly, vol. 29, 2006, pp. 22-30.

[11] O. Lartillot, “Une analyse musicale automatique suivant une heuristique perceptive”, 3ème Conférence Internationale Francophone sur l’Extraction et la Gestion des Connaissances, EGC 03, Lyon, France, 2003.

[12] O. Lassila and R. R. Swick, “Resource Description Framework (RDF) Model and Syntax”, W3C specification, 1998.

[13] R. Parncutt, J. A. Sloboda, M. Raekallio, E. F. Clarke, and P. Desain. “An Ergonomic Model of Keyboard Fingering for Melodic Fragments”, Music Perception: An Interdisciplinary Journal Vol. 14, No. 4, 1997, pp. 341-382.

[14] Y. Raimond, S. Abdallah, M. Sandler, and F. Giasson, “The Music Ontology”, Proceedings of the International Conference on Music Information Retrieval, ISMIR, 2007.

[15] Y. Raimond, “A distributed Music Information System”, Ph.D. Thesis, Department of Electronic Engineering, Queen Mary, University of London, November 2008.

[16] P. Rigaux, “Neuma Ontology Specification”, Project Neuma Report, Lamsade-CNRS, ANR-08, 2008.

[17] O. Sébastien, N. Conruyt, and D. Grosser, “Defining e-services using a co-design platform: Example in the domain of instrumental e-learning", Journal of Interactive Technology and Smart Education, Vol. 5, issue 3, ISSN 1741-5659, Emerald Group Publishing Limited, 2008, pp. 144-156.

[18] V. Sébastien, D. Sébastien, and N. Conruyt, “A collaborative platform model for digital scores annotation”, 3rd Annual Forum on e-Learning Excellence in the Middle East, Dubaï, 2010.

[19] V. Sébastien, D. Sébastien, and N. Conruyt, “An Ontology for Musical Performances Analysis. Application to a Collaborative Platform dedicated to Instrumental Practice”, The Fifth International Conference on Internet and Web Applications and Services, ICIW 2010, Barcelona, 2010, pp. 538-543.

[20] J. A. Sloboda, E. F. Clarkeb, R. Parncutt, and M. Raekallio, “Determinants of Finger Choice in Piano Sight-Reading”, Journal of Experimental Psychology: Human Perception and Performance, Volume 24, Issue 1, 1998, pp. 185-203.

[21] D.R. Stammen and B. Pennycook, “Real-time Recognition of Melodic Fragments using the Dynamic Timewarp Algorithm”. ICMC Proceedings, 1993, pp. 232-235.

[22] M. Stanley, R. Brooker, and R. Gilbert, “Examiner Perceptions of Using Criteria in Music Performance Assessment”. Research Studies in Music Education, June 2002, vol. 18, issue 1, pp. 46-56.

[23] T. Li, M. Ogihara and G. Tzanetakis, “Music Data Mining”, CRC Press, July 2011.

[24] M. A. Winget, “Annotations on musical scores by performing musicians: Collaborative models, interactive methods, and music digital library tool development”, Journal of the American Society for Information Science and Technology, 2008

[25] http://e-guitare.univ-reunion.fr, visited on the 23/01/2012.

[26] http://www.guitar-pro.com, visited on the 23/01/2012.

[27] http://www.apple.com/ilife/garageband/#basic-lessons, visited on the 23/01/2012.

[28] Flash Interactive Guitar Saloon: http://e-guitare.univ-reunion.fr/figs, visited on the 23/01/2012.

[29] http://musicbrainz.org, visited on the 23/01/2012.

[30] http://icking-music-archive.org, visited on the 23/01/2012.

[31] http://youtube.com, visited on the 23/01/2012.

[32] http://www.ehow.com/, visited on the 23/01/2012.

[33] http://creativecommons.org/, visited on the 23/01/2012.

[34] http://www.youtube.com/watch?v=ZTOmYLTitGg, visited on the 23/01/2012.

[35] http://www.last.fm, visited on the 23/01/2012.

[36] http://www.finalemusic.com, visited on the 23/01/2012.

[37] http://musescore.org, visited on the 23/01/2012.

[38] http://musescore.com/sheetmusic, visited on the 23/01/2012.

[39] http://www.free-scores.com/, visited on the 23/01/2012.

[40] http://e-piano.univ-reunion.fr/tests/ScoreAnalyser/readScore.php, visited on the 23/01/2012.

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