CTRQ 2014

The Seventh International Conference on Communication Theory, Reliability, and Quality of Service

ISBN: 978-1-61208-322-3

February 23 - 27, 2014

Nice, France

CTRQ 2014 Editors

Petre Dini, Concordia University, Canada/ China Space Agency Center, CHINA

CTRQ 2014

Foreword

The Seventh International Conference on Communication Theory, Reliability, and Quality of Service (CTRQ 2014), held between February 23rd-27th, 2014 in Nice, France, continued a series of events focusing on the achievements on communication theory with respect to reliability and quality of service. The conference also brought onto the stage the most recent results in theory and practice on improving network and system reliability, as well as new mechanisms related to quality of service tuned to user profiles.

The processing and transmission speed and increasing memory capacity might be a satisfactory solution on the resources needed to deliver ubiquitous services, under guaranteed reliability and satisfying the desired quality of service. Successful deployment of communication mechanisms guarantees a decent network stability and offers a reasonable control on the quality of service expected by the end users. Recent advances on communication speed, hybrid wired/wireless, network resiliency, delay-tolerant networks and protocols, signal processing and so forth asked for revisiting some aspects of the fundamentals in communication theory. Mainly network and system reliability and quality of service are those that affect the maintenance procedures, on the one hand, and the user satisfaction on service delivery, on the other hand. Reliability assurance and guaranteed quality of services require particular mechanisms that deal with dynamics of system and network changes, as well as with changes in user profiles. The advent of content distribution, IPTV, video-on-demand and other similar services accelerate the demand for reliability and quality of service.

We take here the opportunity to warmly thank all the members of the CTRQ 2014 Technical Program Committee, as well as the numerous reviewers. The creation of such a high quality conference program would not have been possible without their involvement. We also kindly thank all the authors who dedicated much of their time and efforts to contribute to CTRQ 2014. We truly believe that, thanks to all these efforts, the final conference program consisted of top quality contributions.

Also, this event could not have been a reality without the support of many individuals, organizations, and sponsors. We are grateful to the members of the CTRQ 2014 organizing committee for their help in handling the logistics and for their work to make this professional meeting a success.

We hope that CTRQ 2014 was a successful international forum for the exchange of ideas and results between academia and industry and for the promotion of progress in the field of communication theory, reliability and quality of service.

We are convinced that the participants found the event useful and communications very open. We also hope the attendees enjoyed the charm of Nice, France.

CTRQ Advisory Committee:

Eugen Borcoci, Politehnica University of Bucharest, Romania
Joel Rodrigues, Instituto de Telecomunicações / University of Beira Interior, Portugal
Pascal Lorenz, University of Haute Alsace, France
Raj Jain, Washington University in St. Louis, USA

CTRQ 2014

Committee

CTRQ Advisory Committee

Eugen Borcoci, Politehnica University of Bucharest, Romania
Joel Rodrigues, Instituto de Telecomunicações / University of Beira Interior, Portugal
Pascal Lorenz, University of Haute Alsace, France
Raj Jain, Washington University in St. Louis, USA

CTRQ 2014 Technical Program Committee

Mandicou Ba, Université de Reims Champagne-Ardenne, France
Dirk Bade, University of Hamburg, Germany
Jasmina Barakovic Husic, BH Telecom – Sarajevo, Bosnia and Herzegovina
Roberto Beraldi, "La Sapienza" University of Rome, Italy
Eugen Borcoci, Politehnica University of Bucharest, Romania
Bruno Checcucci, Perugia University, Italy
Laurent Ciavaglia, Alcatel-lucent, Italy
Javier Del Ser Lorente, TECNALIA Research & Innovation, Spain
Michel Diaz, LAAS, France
Manfred Droste, Universität Leipzig, Germany
Ali El Masri, Troyes University of Technology -Troyes, France
Andras Farago, The University of Texas at Dallas - Richardson, USA
Alexandre Fonte, Polytechnic Institute of Castelo Branco, Portugal & Centre for Informatics and Systems of the University of Coimbra (CISUC) , Portugal
Tulsi Pawan Fowdur, University of Mauritius, Mauritius
Julio César García Alvarez, Universidad Nacional de Colombia Sede Manizales, Colombia
Bogdan Ghita, University of Plymouth, UK
Marc Gilg, Université de Haute Alsace, France
Rita Girão Silva, DEEC-FCTUC / INESC-Coimbra, Portugal
Antti Hakkala, University of Turku, Finland
Bjarne J. Helvik, NTNU, Norway
Robert Ching-Hsien Hsu, Chung Hua University, Taiwan
Mohsen Jahanshahi, Islamic Azad University - Central Tehran Branch, Iran
Brigitte Jaumard, Concordia University, Canada
Sokratis K. Katsikas, University of Piraeus, Greece
Wojciech Kmiecik, Wroclaw University of Technology, Poland
Michal Kucharzak, Wroclaw University of Technology, Poland
Archana Kumar, Delhi Institute of Technology & Management - Haryana, India
Mikel Larrea, University of the Basque Country UPV/EHU, Spain
Pascal Lorenz, University of Haute Alsace, France
Malamati Louta, University of Western Macedonia - Kozani, Greece
Zoubir Mammeri, IRIT - Paul Sabatier University - Toulouse, France

Table of Contents

QoS Provisioning Through Bandwidth Granting Scheme for Wireless Networks

Kuokkwee Wee, Tze Hui Liew, Shih Yin Ooi, Siew-chin Chong
Faculty of Information Science and Technology
Multimedia University
Melaka, Malaysia
Email: wee.kuok.kwee@mmu.edu.my, thliew@mmu.edu.my, syooi@mmu.edu.my, chong.siew.chin@mmu.edu.my

***Abstract*— Quality of Service (QoS) is an essential element in modern wireless networks such as WiFi, Long Term Evolution (LTE) and Worldwide Interoperability for Microwave Access (WiMAX). The QoS provisioning of a wireless network can be secured in the scheduling, admission control, bandwidth granting and queuing components that reside at the MAC layer. This paper focuses on the bandwidth granting mechanism for uplink traffic transmission. Until now, not much study has been done by researchers on improvements of this particular mechanism although it is an important component in QoS framework. Through our experiments, it is discovered that the network performance could be further improved by introducing custom-designed mechanisms in the bandwidth granting process. Another thing to note is that not all the common scheduling algorithms are appropriate to be implemented since the information on the bandwidth granting process is very limited. Furthermore, the design of bandwidth granting mechanism must be simple and fast. Thus, by taking into account these limitations and concerns, two mechanisms have been proposed and evaluated in this study. The traditional and typical approach used in bandwidth granting scheme is bench-marked and compared with our proposed mechanisms. The simulation results show that our proposed mechanisms outperform the conventional approach.**

Keywords- scheduling;bandwidth granting;quality of service; 4G; wireless network

I. INTRODUCTION

Worldwide Interoperability for Microwave Access (WiMAX) is an example of broadband wireless access (BWA) and defined by IEEE in its IEEE 802.16 standards [1][2]. The IEEE 802.16 standard is further divided into two categories which are the IEEE 802.16d (fixed WiMAX) and the IEEE 802.16e (mobile WiMAX) [2]. The recent development in the WiMAX standard has allowed many service providers to adapt this technology as the alternate solution for last-mile delivery.

Theoretically, broadband access to an area blanketed by a radius of 31 miles can be covered using the WiMAX technology [1], [2]. This distance is achieved when using WiMAX for line-of-sight (LOS) backhaul service. However, for a deployment in the urban environment, it is difficult to achieve LOS between the receiver and the BS. In NLOS WiMAX, the signals arriving may come from reflected paths, scattered energy and some diffracted propagation paths. Hence, the area size is significantly lesser with only a radius of 3 to 5 miles covered.

Its higher bandwidth and large area coverage compared to other BWAs makes WiMAX a suitable solution for many applications. Some examples of these applications include high quality Voice over Internet Protocol (VOIP), Video on Demand (VoD) and Internet Protocol Television (IPTV) services.

The ability to be able to support multimedia applications such as IPTV and VoD is a challenging task faced by Internet Service Providers (ISP) worldwide. The Quality of Service (QoS) assurance for these multimedia applications is usually implemented from the application layer to the physical layer. Unlike other layers, the layer 2 (or the MAC layer) is varied from one technology to another. It is highly dependent on the medium access technology, especially when using wireless as the medium of access.

As one of the prominent 4G technologies, WiMAX is designed to support all kind of services with QoS assurance. In general, the MAC layer is responsible to assure the quality of packet delivery. There are many components in the MAC layer that assist in QoS provisioning. Some examples include the Call Admission Control (CAC), uplink/downlink schedulers, ranging, fragmentation and defragmentation, classifier and MAC management entity. The entire MAC architecture for subscriber station (SS) and base station (BS) is shown in Fig. 1.

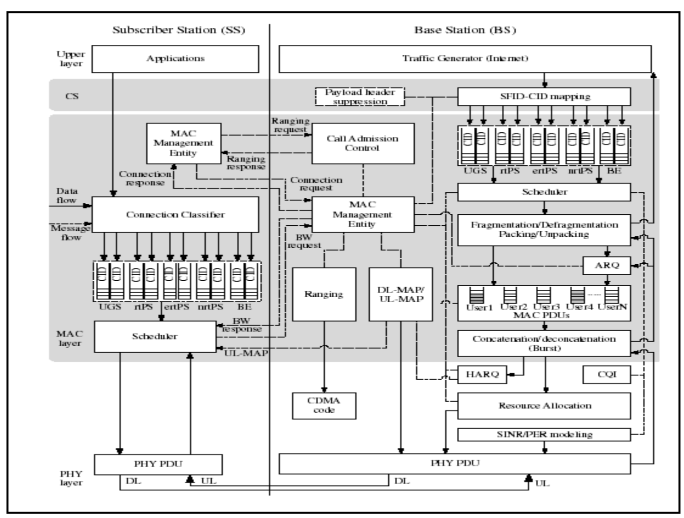

Fig. 1. MAC architecture for IEEE 802.16

Incoming traffic is categorized into five service classes in IEEE 802.16 [1], [2]. The five service classes are the Unsolicited Grant Service (UGS), Extended Real-time Polling Service (ertPS), Real-Time Polling Service (rtPS), Non-Real Time Polling Service (nrtPS), and Best Effort (BE). UGS traffic is aimed for VOIP service without silence suppression while ertPS service is for VOIP with silence suppression. Meanwhile, rtPS is designed for real-time Internet application such as VoD and online gaming. Both nrtPS and BE are targeted on non-real time services. Specifically, nrtPS is focused on the non-real time services that require bandwidth in variable sizes.

The main contribution of this study is addressing the issues in bandwidth granting scheme at base station in WiMAX network, which has been overlooked by many researchers. Unlike uplink scheduler or downlink scheduler, information is very limited during bandwidth granting scheme. BS is unlikely to know the queue status, incoming packet rate or packet delay from the bandwidth requested message obtained from SS. Thus, most of the recent scheduling techniques are unable to work in bandwidth granting protocol. Proportional bandwidth granting schemes have been proposed and investigated in this study.

This paper is organized as follows. Section II presents the bandwidth request/granting process of the IEEE 802.16e and Section III describes the proposed bandwidth granting mechanisms. Section IV depicts the simulation experiment environments and discusses the simulation results. Lastly, conclusions and future plans are presented in Section V.

II. BANDWIDTH REQUEST/GRANTING PROCESS

Despite there being several QoS components in the MAC layer, our research is only focused towards the bandwidth granting mechanism for uplink transmission. The bandwidth granting process happens only at the BS when a SS is requesting bandwidth for its uplink transmission, whereby uplink transmission is defined as the transfer of packet from a SS to the BS. The uplink transmission is much more complicated than the downlink transmission because the information required by the uplink transmission is very limited compared to the downlink transmission. Local information such as the queue size is always available in the downlink transmission [3] but not for the uplink.

Before an uplink transmission can commence, a SS is required to send a bandwidth request message to the BS. The bandwidth request message can be sent either explicitly or implicitly. In the explicit approach, the bandwidth request is attached and embedded in the data message while in the implicit approach, the bandwidth request is the only message sent to the BS. Another common bandwidth request approach is the contention based [4] approach. Contention based approach in the IEEE 802.16 is identical to the contention approach used in the IEEE 802.11. See TABLE I for the eligibility of bandwidth request for different service classes. The IEEE 802.16 standard allows the bandwidth request to be done on per connection basis or per station basis. The bandwidth request per station is claimed to be more efficient due to the low management message usage [5].

TABLE I UPLINK REQUEST RULES

UL request/grant	UGS	rtPS	nrtPS	BE
Implicit		√	√	√
Explicit		√	√	√
Contention based			√	√

Upon receiving a bandwidth request message, the BS stores the message in a queue based on its arrival time. The bandwidth request messages are processed by the bandwidth granting module before a new frame starts. This bandwidth granting process is very challenging since the information available is limited [6], [7]. From the bandwidth request messages received, the BS only have the information on the amount of bandwidth needed by the SS and the service class requirements. The situation becomes worse when the requests are from homogeneous application or the requesters are having similar QoS requirements. In this context, the BS can only depend on the amount of bandwidth needed during the bandwidth granting process. Due to the above concerns, many scheduling algorithms commonly used in uplink/downlink scheduler are not suitable in the bandwidth request module. For instance, strict priority policy used in downlink scheduler [7], [8] is meaningless when the requests are from homogeneous applications. Weighted round robin [10], deficit round robin [11], weighted deficit round robin [12], worst-case fair weighted fair queuing [13], earliest deadline first [14] and other packet information based scheduling approaches from [15], [16] and [17] are not able to be implemented in bandwidth granting module because of the unavailability of packet and queue information from the SS. Moreover, the bandwidth granting scheme is always neglected by many researchers and the most typical algorithm used in the bandwidth granting scheme is first-come-first-serve basis [9], [18]. Hence, a custom-designed mechanism for the uplink bandwidth granting is needed and essential. We observed that the bandwidth granting mechanism which is not getting adequate attention of researchers has an important role in QoS provisioning of an IEEE 802.16e network. Experiment results have proven that the network performance (throughput and jitter) can be improved by introducing custom-designed bandwidth granting mechanism.

III. PROPORTIONAL BANDWIDTH GRANTING SCHEMES

The proposed bandwidth granting schemes consist of i) proportional byte based (PBB) and ii) proportional physical slot based (PPSB). In general, the amount of granted bandwidth of a SS depends on the following:

- Total amount of bandwidth request
- Individual amount of bandwidth request
- Total amount of available bandwidth

Total amount of bandwidth request is the summation of all individual bandwidth requests from each SS at the current cycle. Meanwhile, individual amount of bandwidth request is referring to the bandwidth request received by BS for each SS. Last but not least, total amount of available bandwidth is given by the vacancy slots, which are ready for transmission in the next cycle.

The proportional byte based approach in bandwidth granting mechanism, PBB, is first presented in this research. The PBB mechanism keeps the individual bandwidth request in the byte format, which is its original value extracted from the bandwidth request message. The amount of granted bandwidth is calculated according to the percentage of occupancy by an individual bandwidth request. More bandwidth is given to those requesters who have higher bandwidth demands without consider any channel condition or modulation scheme. Unlike the approach used by [9] and [18], all requesters will be given some amount of the available bandwidth in PBB. At least some portions of the bandwidth will be granted to every SS and hence, starvation of SS can be avoided. This approach intended is to give fairness to all of the requested too. The formula for PBB in bandwidth granting is as in (1).

$$BW_i = \left\lfloor \frac{BR_i}{\sum BR} BW \right\rfloor \quad (1)$$

where BW_i is the amount of granted bandwidth and BR_i is the amount of individual bandwidth request for the ith request respectively. $\sum BR$ represents the total amount of the bandwidth requests while for the BW, this is the total available bandwidth.

The second mechanism, PPSB is similar to PBB but it also takes into account the concern in [19], whereby the conversion from bandwidth request in byte to physical slot causes extra unused bandwidth to be allocated. In order to overcome this issue, PPSB converts the amount of bandwidth request, the total amount of bandwidth and the available bandwidth from byte to physical slot before any bandwidth allocation calculation. The conversion of byte to physical slot requires the Channel Quality Indicator (CQI) data for each SS. Overall, a better CQI is able to carry more data as compared to a lower CQI. PPSB mechanism also provides a more accurate allocation when there are different wireless network conditions in a scenario. As known, poor CQI requires more physical slots to send the same amount of data compared to those SSs in a better CQI network. The byte to physical slot conversion is based on (2).

$$\mathrm{BRPhySlot} = \frac{\mathrm{BRbyte}}{\mathrm{bytePS}} \quad (2)$$

where *BRPhySlot* is the bandwidth request in physical slot unit, *BRbyte* is the bandwidth request in byte unit and the *bytePS* reflects the amount of data in bytes that can be carried by a physical slot, which depends on Uplink Interval Usage Code (UIUC) as in TABLE II. In general, a higher UIUC index will result more bytes per slot. Maximum 27 bytes per physical slot is defined in [3].

TABLE II UIUC INDEX AND BYTE PER PHYSICAL SLOT

UIUC Index	Number of Byte per Slot
1	6
2	9
3	12
4	15
5	18
6	24
7	27

Upon the conversion, the amount of granted bandwidth is then calculated by using (3). Through this approach, the loss during the conversion may be identified. PPSB is also designed for network scenario whereby more than one wireless conditions or modulation and coding schemes are found in a WiMAX network.

$$\mathrm{BWPS_i} = \left\lfloor \frac{\mathrm{BRPhySlot}_i}{\sum \mathrm{BRPhySlot}} \mathrm{BWPhySlot} \right\rfloor \quad (3)$$

where $BWPS_i$ is the amount of granted bandwidth in physical slot unit and $BRPhySlot_i$ is the amount of individual bandwidth request in physical slot for the ith request respectively. $\sum BRPhySlot$ represents the total amount of the bandwidth requests whereas the *BWPhySlot* is the total available bandwidth in physical slot.

IV. SIMULATION EXPERIMENTS

A. Simulation Model and Environment

The simulation experiments were conducted using Qualnet network simulator version 5.0. The network scenario is a single cell with a BS and 10 SSs as in Fig. 2. All the SSs are located 150 meter away from the BS. The network simulation parameters for our experiment are as presented in TABLE III.

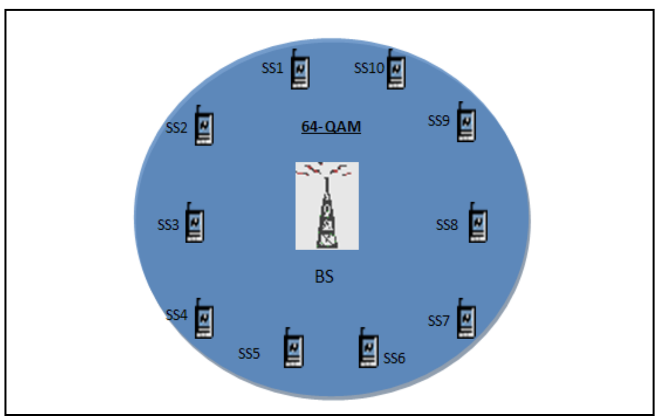

Fig. 2. Simulation scenario

 ISBN: 978-1-61208-322-3

TABLE III SIMULATION PARAMETERS

Simulation Parameters	
PHY	OFDMA
Bandwidth	20MHz
Cyclic Prefix, FFT Length	1/8, 2048
UL/DL Frame Length	10ms
Modulation	16,64-QAM
Antenna Type	Omni-directional
Simulation Duration	0 - 90s
Wait UCD/DCD timeout interval	25s
UCD/DCD	5s

In order to validate the performance of the proposed bandwidth granting mechanisms in a homogenous environment, some slight modifications on the incoming traffic which base on [18] is used. Only rtPS traffic is selected and it is examined in our experiments. The performance of the proposed granting mechanisms in handling the traffic from the same service class is to be observed in our study. Furthermore, rtPS is designed for video and multimedia transmission, whereby both are more challenging and demanding. All the SSs were equipped with 2 uplink traffic with a 0.8 Mbps and a 1.2 Mbps traffic load respectively. Both implicit and explicit mechanisms were allowed in the bandwidth request mechanism.

The experiments were simulated in a network scenario, whereby no different modulation scheme and coding or no different wireless network condition were created. A network scenario with only one type of wireless network condition, the 64 QAM was simulated. Our intention in this context was to test the efficiency of the proposed bandwidth granting mechanisms in the IEEE 802.16e network.

Our proposed bandwidth granting mechanisms is benchmarked against the common bandwidth granting mechanism (CBGM) approach used by [9], [18] and the statistical approach (SA) from [20]. The network performance metrics; throughput, delay, and jitter, are makred as the major assessment elements in this study. We also assumed that there will be no new incoming traffic during the simulation.

B. Simulation Results and Discussions

Fig. 3 shows the total average end-to-end throughput for CBGM, PPSB, PBB and SA over the simulation time. The SA produces the highest throughput for the first 20 seconds of the simulation time. During the first 20 seconds, bandwidth requests for all SSs are at a moderate level, the reserved bandwidth is significant to be distributed fairly by using SA approach [20]. However, the performance of SA is overtaken by PPSB and PBB after the 30 seconds mark. The proposed PPSB and PBB always perform better than CBGM in this study. PBB mechanism achieves 1.29% higher total throughput in average as compared to CBGM. The gap of difference between the PBB and CBGM becomes smaller along the simulation time. Also, it is observed that there is no significant difference between the PPSB and PBB in terms of throughput. These results indicate that byte to physical slot conversion issue is negligible in proportional approach when all the SSs are within the same modulation scheme and coding.

The total average end-to-end delay is presented in Fig. 4. CBGM has the best performance for the first 30 seconds. For 40 seconds and onwards, SA has the lowest latency. There is a very significant different between CBGM and other approaches at the 10 seconds mark. The difference between PBB/PPSB and CBGM gets lesser as the simulation time passes. It averages about 12% higher delay between PBB/PPSB and CBGM at the 90 seconds mark. Similar to the findings of the above, both PPSB and PBB do not have significant difference in the delay performance. For instance, PBB is only 0.5% better than PPSB at the 90 seconds mark. Also, SA is the best performance for this category except during the first 30 seconds.

From Fig. 5, for the total average end-to-end jitter, it is observed that both the PBB and PPSB mechanisms have improved the jitter performance of IEEE 802.16e networks, regardless of the simulation time. It is also observed that the SA has the worst results in this context. The jitter is improved from the range of 4.6% (at 10 seconds) to 2.1% (at 90 seconds) when comparing the PPSB with CBGM. This phenomenon reflects that the PPSB and PBB mechanism spare some bandwidth to every SS instead of one or two greedy SS. Greedy SS requests and consumes a majority of the available bandwidth than others. The network becomes less healthy when the number of greedy SS increases. It is observed that the PPSB and PBB have an impact in controlling the greedy SS but it is not significant enough.

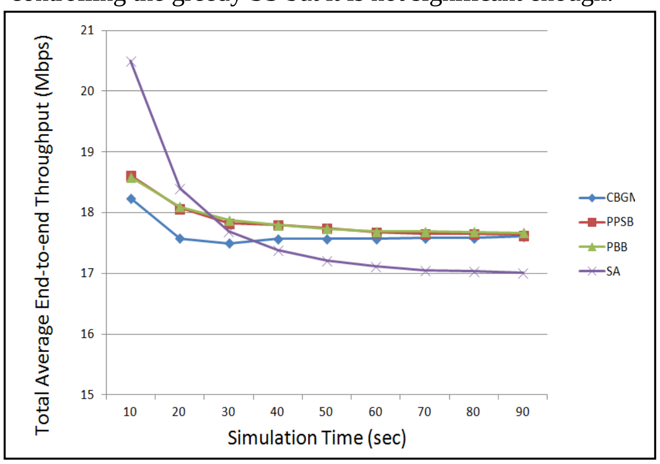

Fig. 3. Total average end-to-end throughput

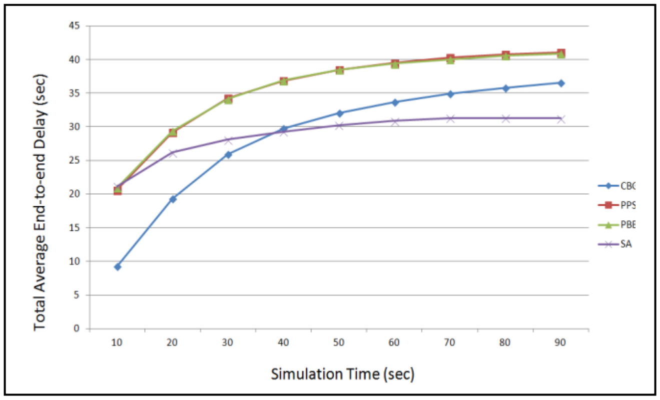

Fig. 4. Total average end-to-end delay

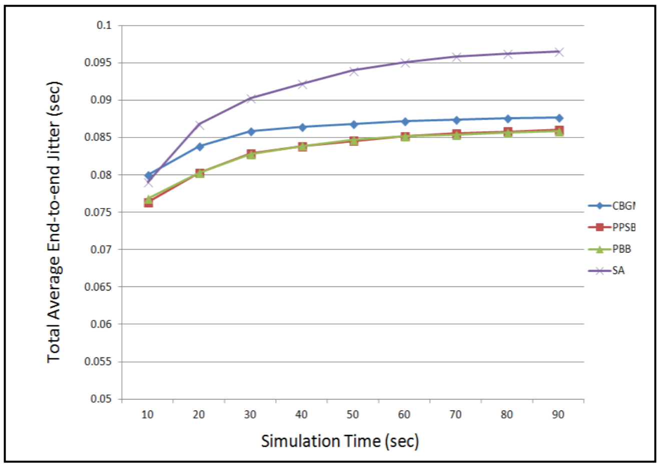

Fig. 5. Total average end-to-end jitter

V. CONCLUSION AND FUTURE DIRECTION

PPSB and PBB have the better average throughput and jitter, as observed from the simulation results. However, both PPSB and PBB failed to improve the delay, which is one of the crucial performance metrics for rtPS traffic. The poor delay result cannot be compromised for real-time application although a better result in throughput and jitter could be achieved. In conclusion, the PPSB and PBB are not suitable for real-time traffic. PPSB and PBB are for non-real-time applications (nrtPS traffic) that do not have delay stringency and only target on the throughput. We also observed that some mechanisms to regulate the greedy bandwidth requesters and custom-designed bandwidth granting scheme is urgently needed, especially for homogenous application.

From the results obtained, the proposed PBB and PPSB will be tested again by using nrtPS traffic, which does not have delay requirement. The modulation scheme and coding is another important consideration in carrying out a research in wireless networks. Hence, more network scenarios in different modulation schemes and codings will be created for a more extensive testing in the future.

REFERENCES

[1] IEEE, IEEE Standard for Local and Metropolitan Area Networks Part 16: Air Interface for Fixed Broadband Wireless Access Systems, 2004.

[2] IEEE, IEEE Standard for Local and metropolitan area networks--Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems, 2009.

[3] IEEE, IEEE Standard for Air Interface for Broadband Wireless Access Systems, 2012.

[4] J. F. Borin and N. Da Fonseca, “Uplink Scheduler and Admission Control for the IEEE 802.16 Standard,” in Global Telecommunications Conference 2009 GLOBECOM 2009 IEEE, 2009, pp. 1–6.

[5] W. K. New, K. Wee Y. Wee, and C.-O. Wong, “WiMAX: Performance Analysis and Enhancement of Real-time Bandwidth Request,” IAENG International Journal of Computer Science, vol. 40, no. 1, 2013, pp. 20–28.

[6] S. Z. Tao and A. Gani, “Intelligent Uplink Bandwidth Allocation Based on PMP Mode for WiMAX,” Proc. 2009 International Conference on Computer Technology and Development, 2009, pp. 86–90.

[7] Kuokkwee Wee, R. Mardeni, S. W. Tan, and S. W. Lee, “QoS Prominent Bandwidth Control Design for Real Time Traffic in IEEE 802.16e Broadband Wireless Access,” The Arabian Journal for Science and Engineering, in press.

[8] J. Sun, Y. Yao, and H. Zhu, “Quality of service scheduling i broadband wireless access systems,” Proc. 2006 IEEE 63rd Vehicular Technology Conference, 2006, vol. 3, pp. 1221–1225.

[9] K. K. Wee and S. W. Lee, “Priority Based Bandwidth Allocation Scheme for WiMAX Systems,” Proc. 2nd IEEE International Conference on Broadband Network & Multimedia Technology, 2009, pp. 15–18.

[10] M. S. Arhaif, “Comparative study of scheduling algorithms in WiMAX,” International Journal of Scientific & Engineering Research, vol. 2, no. 2, 2011, pp. 1–7.

[11] C.-H. H. C.-H. Hsieh, T.-J. W. T.-J. Wu, and H.-T. C. H.-T. Chern, “Bandwidth Control Protocol in WiMAX Network,” Proc. Parallel and Distributed Processing with Applications ISPA 2010 International Symposium on, 2010, pp. 342–349.

[12] C. Cicconetti, L. Lenzini, E. Mingozzi, and C. Eklund, “Quality of service support in IEEE 802.16 networks,” IEEE Network, vol. 20, no. 2, 2006, pp. 50–55.

[13] H. J. E. Blanco and I. P. P. Parra, “Evaluation of scheduling algorithms in WiMAX networks,” in ANDESCON 2010 IEEE, 2010, pp. 1–3.

[14] A. Chakchai, S. Raj, and J. K. Abdel, “Resource allocation in IEEE 802.16 mobile WiMAX,” in Orthogonal Frequency Division Multiple Access, 2010.

[15] A. Esmailpour and N. Nasser, “Dynamic QoS-Based Bandwidth Allocation Framework for Broadband Wireless Networks,” IEEE Transactions on Vehicular Technology, vol. 60, no. 6, 2011, pp. 2690–2700.

[16] M. Andrews, K. Kumaran, K. Ramanan, A. Stolyar, P. Whiting, and R. Vijayakumar, “Providing quality of service over a shared wireless link,” IEEE Communications Magazine, vol. 39, no. 2, 2001, pp. 150–154.

[17] C. Tian and D. Yuan, “A novel cross-layer scheduling algorithm for IEEE 802.16 WMAN,” Proc. International Workshop on Cross Layer Design, 2007, pp. 70-73.

[18] J.-M. Liang, J.-J. Chen, Y.-C. Wang, and Y.-C. Tseng, “A Cross-Layer Framework for Overhead Reduction, Traffic Scheduling, and Burst Allocation in IEEE 802.16 OFDMA

Networks," IEEE Transactions on Vehicular Technology, vol. 60, no. 4, 2011, pp. 1740–1755.

[19] X. Bai, A. Shami, and Y. Ye, "Robust QoS Control for Single Carrier PMP Mode IEEE 802.16 Systems," IEEE Transactions on Mobile Computing, vol. 7, no. 4, 2008, pp. 416–429.

[20] Kuokkwee Wee, W. K. New, Y. Wee, and C.-O. Wong, "Intensive Bandwidth Request and Handling Design in PMP," International Journal of Computer Science and Network Security, vol. 12, no. 2, 2012, pp. 27–32.

[21] K. Wee, M. Roslee, S. W. Tan, and S. W. Lee, "Statistical Approach In Bandwidth Granting Process for IEEE 802.16 Networks," Proc. 8th International Conference on Wireless Communications Networking and Mobile Computing, IEEE Press, 2012, pp. 1-4.

Blocking Probabilities in Multi-Service Systems with Preemptive Scheduling

Shuna Yang, Norvald Stol
Department of Telematics
Norwegian University of Science and Technology, NTNU
Trondheim, Norway
email: {shuna, norvald.stol}@item.ntnu.no

***Abstract*—This paper investigates blocking probabilities in multi-service communication systems, in which the preemptive scheduling is adopted to implement service differentiation. A novel approximation model is proposed. In contrast with existing multi-dimensional Markov model, which focuses on analyzing the small system with only two service classes and results in non-closed form expressions of blocking probabilities, our model has three major advantages: 1) it is applicable to analyzing a general multi-service scenario, independent of the number of service classes and resources; 2) the closed form expressions of blocking probabilities can be derived directly; 3) this model shows excellent extensibility for analyzing larger system which supports more service classes or common resource units. The analytical values are compared with simulation results for two- and three-service systems. Results show that the proposed model provides a high degree of accuracy in the blocking probabilities under different scenarios.**

Keywords-Markov chain; Preemptive Scheduling; blocking probability; service differentiation .

I. INTRODUCTION

Driven by increasing communication needs worldwide, a wide variety of services and applications will be brought into the future communication networks. Some of them have comparable demands to today's services, while some demands much more strict requirements in terms of bandwidth and time delay [1]. In order to meet the diverse service demands, the scheduler (in routers or switches) has to deploy efficient handling schemes to serve the different applications in different ways. In the past the programmers resorted to a rigid, pre-determined order for execution of different services, so that the corresponding service times could be predicted in advance [2]. Unfortunately these cyclic executive methods result in programs that are hard to understand and maintain because the code for logically independent tasks is interleaved. In order to guarantee the service of the safety-sensitive applications as well as simplify the task processing on large schedulers, preemptive scheduling approaches attract notable research efforts [3, 4].

In this paper we consider multi-service communication systems which integrate different kinds of applications together (some of them are safety-sensitive applications while some are safety-nonsensitive services). Central to these systems is a service facility with multiple common shared resource units (which may be interpreted according to the application under consideration as communication channels [5], computer memory sectors [6], time slots in a TDM bus [7], wavelength channels in an OPS/OBS (optical packet/burst switched) network [8, 9], etc.) and a service discipline of preemptive scheduling. That is, each type of service class is given a fixed priority and an interrupt mechanism is executed. Each class is served according to its assigned priority and the being served user can be preempted/interrupted by the higher priority arriving users in case of no available resource units. Otherwise it occupies the required resource unit for the duration of its service time.

The performance of multi-service systems with preemptive scheduling can be evaluated by the existing multi-dimensional Markov model, which is built based on a variant of the multi-dimensional Erlang blocking model. References [8]-[10] give a detailed discussion about the blocking probabilities in two-service systems (in this case applied to an OPS/OBS network) using this model. However, the existing research focuses on two-dimensional Markov models, which can only be used for analyzing small systems supporting only two kinds of service classes. For the larger system which supports more service classes or common resource units, the model will become very complex and computationally much harder to solve. The reason is that it will introduce an excessive number of states/parameters (i.e. $O(N^R)$ states and parameters, where N is the number of shared resource units and R the number of supported service classes) [11]. Another major limitation is that no closed form expressions of blocking probabilities can be derived. Hence, the existing multi-dimensional Markov models have limited applicability in modeling multi-service systems which have practical value.

In this paper we propose a novel approximation model to analyze the performance of multi-service systems with preemptive scheduling. By using conditional partitioning method, this model builds multiple levels of one-dimensional Markov chains. Each level presents all possible service states of the corresponding service class in the system. The blocking probabilities can be calculated level by level and their closed form expressions are derived directly. Compared with the existing multi-dimensional Markov model, the proposed model has several significant advantages: 1) it is applicable to analyzing the general multi-service system, independent of the number of the supported service classes and resource units; 2) the closed form expression of the blocking probability for any service class can be derived directly and separately; 3) by using the one-dimensional Markov chains to calculate blocking probabilities, the computational complexity is decreased

 ISBN: 978-1-61208-322-3

dramatically; 4) this model shows excellent extensibility for analyzing the larger system which supports more service classes and resources. Besides the model of the general multi-service case, we also give the concrete models of the two- and three-service system cases in this paper.

The rest of this paper is organized as follows. Section II presents the operation of preemptive scheduling for the studied system. Section III proposes the approximation model and derives the closed form expressions of blocking probabilities; it also gives two concrete models of two- and three-service systems. Both analytical and simulation results are given in Section IV. Section V concludes the paper.

II. The Preemptive Scheduling

As shown in Fig. 1, we consider the system with a capacity of N common resource units. The system services R (R is an integer) mutually independent classes of users: class 1 has the highest priority and class R has the lowest priority. For $1 \leq i \leq R$, class i users are assumed to arrive according to a Poisson process with arrival rate φ_i. Meanwhile, a class i user has a request size of one resource unit and an exponentially distributed holding time with mean value of μ_i^{-1}. Thus, the average traffic offered to the system by a class i arrival process is equal to: $A_i = \varphi_i/\mu_i$.

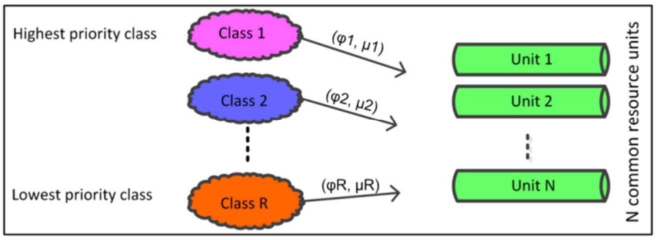

Figure 1. The traffic model of the studied system.

Fig. 2 presents the detailed operation of the preemptive scheduling when a new user arrives. All available resource units are shared among R different classes. As long as there exist available resources, the new arriving user is served directly independent of its priority. However, if all resources are occupied, this new arriving user should check its priority with that of the being served users. As illustrated in Fig. 2, we assume the lowest priority of the being served users in the system is i ($1 \leq i \leq R$) and the priority of this new user arrival is j. If $j \geq i$, this new user arrival will be blocked directly. If $j < i$, it will preempt/interrupt the service of class i user and takes over the respective resource unit for its own use. When i is equal to 1, all the resources are occupied by the highest priority class 1 users. Then all the new arrivals will be blocked and no preemption/interruption will happen.

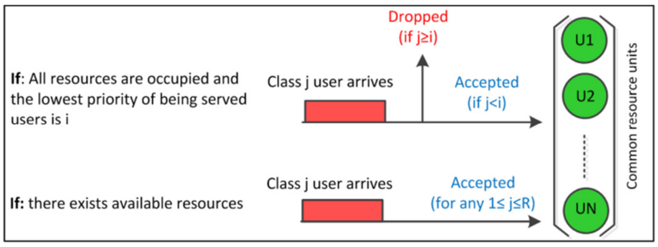

Figure 2. The operation of the preemptive scheduling.

III. Analytical Model

In this section, we present the approximation model to study the blocking in a multi-service communication system with preemptive scheduling. The traffic model is shown in Section II. In this part we first build the analytical model of a general R-service system and present the detailed derivation of the closed form expressions of blocking probabilities. Then we give the concrete models of the two- and three-service case systems to clarify its construction and calculation.

A. The model of the general R-service system

We model the number of the resource units occupied by each class as a continuous time Markov chain. For the R-service communication system, according to the priority of each class, the model is built from the $1st$ /top to the Rth/bottom level as shown in Fig. 3. Each level presents all possible service states of the corresponding class. The $1st$/top level gives all states of the class 1 users while the Rth/bottom level presents all states of the class R. In Fig. 3, state $i_k (0 \leq i \leq N, 1 \leq k \leq R)$ denotes that i resource units are currently serving class k users. Note that class 1 has the highest priority and class R has the lowest priority, the number of common resource units is equal to N.

For class 1 with the highest priority, blocking only happens when all resources are currently occupied by other class 1 users, hence the system can be modeled as a $M/M/N/N$ loss system as shown in the $1st$/top level.

For any state i_1 ($0 \leq i_1 < N$) of class 1 in the $1st$/top level, it indicates that i_1 resources are currently serving class 1 users. Due to the higher priority of class 2 compared with classes from 3 to R, the remaining $(N - i_1)$ resources can be used for serving class 2 users. Accordingly, level 2 has a respective conditional one-dimensional Markov chain whose maximum state is $(N - i_1)$ to denote the service states of class 2. However, when i_1 is equal to N, i.e., all resources are held by class 1 users, no resource can be accessed by class 2. Hence level 2 has N conditional one-dimensional Markov chains corresponding to the different states $(i_1, 0 \leq i_1 < N)$ of class 1.

For any state i_1, $i_2$in the $1st$/top and $2nd$ level, $i_1 + i_2 < N$, there exists a conditional one-dimensional Markov chain whose maximum state is $(N - i_1 - i_2)$ in the third level, i.e., in current i_1, i_2 resource units are busy serving class 1 and 2 users respectively. Also due to the higher priority of class 3 compared with classes from 4 to R, $(N - i_1 - i_2)$ resources can be used for serving class 3 users. Considering all possible combinations of i_1, i_2 and $i_1 + i_2 < N$, level 3 has $N * (N + 1)/2$ conditional one-dimensional Markov chains.

Using the iterative method, we build all conditional one-dimensional Markov chains in each level. Note that Fig. 3 shows only one generic one-dimensional Markov chain in each level. However, when $N > 1$, except for level 1, the other levels have more than one one-dimensional Markov chain, i.e., one for each possible combination of the states in higher levels. When calculating the blocking probabilities, conditional probability principles are used to weigh and sum contributions from each one-dimensional Markov chain in each level. Note that this model only needs to increase R or N when modeling the larger system with more classes or resources, thus the proposed model shows excellent extensibility compared with multi-dimensional Markov models.

In Fig. 3, for each conditional one-dimensional Markov chain except that of level 1, the outgoing transition probability of the last state must be adjusted to take into account arrivals of higher priority users. For instance, for any one-dimensional Markov chain on level k, the last state $(N - Z_k)$ denotes that $(N - Z_k)$ common resources are currently serving class k users while Z_k resources are held by the higher priority users. Since all N resources are currently occupied, the being served class k users can be interrupted/preempted if higher priority users arrive during their holding time. Because of the lower priority of class k compared with classes from 1 to $(k-1)$, $\wedge_k$ and Z_k of the kth level in Fig. 3 are defined as:

$$\wedge_k = \sum_{j=1}^{k-1} \varphi_j \,, \qquad Z_k = \sum_{j=1}^{k-1} i_j \,. \tag{1}$$

Hence, $\wedge_R$ and Z_R in the Rth level can be written as $\wedge_R = \sum_{j=1}^{R-1} \varphi_j$, $Z_R = \sum_{j=1}^{R-1} i_j$.

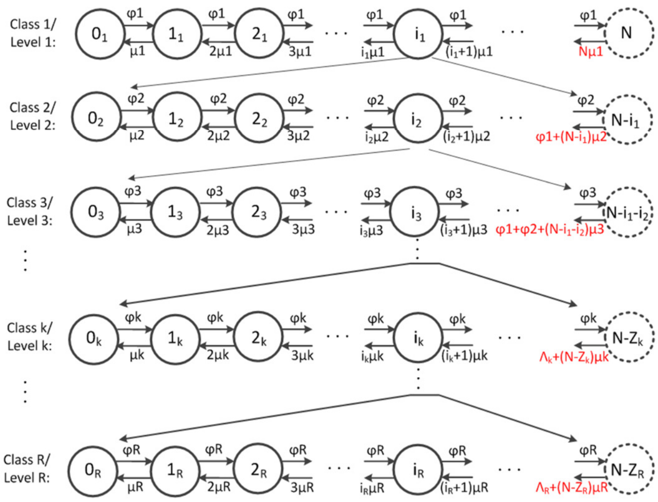

Figure 3. The model of the R-service system with capacity of N common resource units.

According to the model in Fig. 3, the blocking probability of each class can be calculated level by level. Due to the preemptive scheduling, for any class k $(1 \le k \le R)$, its blocking probability $b(k)$ only depends on the traffic pattern of classes with equal or higher priority, while not influenced by the performance of classes with lower priority. We can derive the blocking probability of each class from the highest to the lowest priority/level. In the following analysis, we use $Q_k(i_k)$ to denote the probability of state i for class k, where i resources are currently busy serving class k users.

As illustrated in Fig. 3, the blocking probability $(b(1))$ of class 1 is given directly by the $M/M/N/N$ loss formula [12]:

$$b(1) = Q_1(N) = \frac{A_1^N/N!}{\sum_{v=0}^{N} A_1^v/v!} \tag{2}$$

For any service class k $(1 < k \le R)$, $b(k)$ consists of two parts: one is the $b(k)$ introduced by the new class k user arrivals which are blocked directly; the other is the $b(k)$ given by the being served class k users which are preempted/interrupted by higher priority users. The former happens in all states where all resources are occupied by users whose priority is equal to or higher than k. According to preemptive scheduling, the new class k arrival will be blocked directly. We call these states block states. The latter happens in states where all resources are occupied by users of which the lowest priority is equal to k. In these states these being served class k users will be preempted if higher priority users arrive during their service time. We call these states preemption states. Note that the block states include all preemption states. As discussed above, in preemption states the arrival intensity of all higher priority classes is equal to $\sum_{j=1}^{k-1} \varphi_j$, and the arrival intensity of class k users is φ_k in all states of the studied system. In the following analysis, we use $Q_{k,block}$, $Q_{k,preempt}$ to denote the probabilities of block states and preemption states respectively. We also introduce Q_{all} to denote the probability of all possible service states of the studied system, which is equal to 1. Hence the blocking probability of the service class k is

$$b(k) = \frac{\varphi_k * [Q_{k,block}]}{\varphi_k * Q_{all}} + \frac{\sum_{j=1}^{k-1} \varphi_j * [Q_{k,preempt}]}{\varphi_k * Q_{all}}$$

$$= Q_{k,block} + \frac{\sum_{j=1}^{k-1} \varphi_j}{\varphi_k} * [Q_{k,preempt}] \,. \tag{3}$$

In order to calculate $Q_{k,block}$ and $Q_{k,preempt}$, we have to find all possible block and preemption states and their corresponding probabilities for service class k.

Block states consist of k different cases, we use $Q_{k,block,v}$ $(1 \le v \le k)$ to denote the probability of each case for class k.

$1st$ case: all resources are currently held by only class 1 when a new class k user arrives. The probability is

$$Q_{k,block,1} = Q_1(N). \tag{4}$$

$2nd$ case: all resources are currently occupied by the users whose lowest priority is 2 when a new class k user arrives. So,

$$Q_{k,block,2} = \sum_{i_1=0}^{N-1} Q_1(i_1) Q_2(N - i_1). \tag{5}$$

v-th case $(3 \le v \le k)$: all resources are currently occupied by the users of which the lowest priority is v. Then new class k arrival users will be blocked. Using the iterative method, the respective probability is obtained as

$$Q_{k,block,v} = \left\{ \prod_{j=1}^{v-1} \left[\sum_{i_j=0}^{N-\sum_{l=1}^{j-1} i_l - 1} Q_j(i_j) \right] \right\} Q_v\left(N - \sum_{j=1}^{v-1} i_j\right). \tag{6}$$

where $Q_j(i_j)$ $(1 \le j \le v)$ can be derived by node equations of the corresponding one-dimension Markov chain in level j,

$$\begin{cases} Q_j(i_j) * \varphi_j = Q_j([i+1]_j) * i * \mu_j \,, & 0 \le i \le N - \sum_{d=1}^{j-1} i_d - 2 \\ Q_j(i_j) * \varphi_j = Q_j([i+1]_j) * [\wedge_j + (N - R_j) * \mu_j], & i = N - \sum_{d=1}^{j-1} i_d - 1 \\ \sum_{i=1}^{N-\sum_{d=1}^{j-1} i_d} Q_j(i_j) = 1 \,, & \end{cases} \tag{7}$$

After getting the value of $Q_{k,block,v}$,$Q_{k,block}$ can be obtained:

$$Q_{k,block} = \sum_{v=1}^{k} Q_{k,block,v} \tag{8}$$

$Q_{k,preempt}$ denotes the probability of preemption states that all resources are occupied and the lowest priority of users being served is k. Then the being served class k users can be preempted/interrupted by the higher priority class arrivals. Note that on preemption states, all the new class k arrivals will be blocked directly, the preemption states belong to one case of block states for class k (i.e., $v = k$ of block states). Hence,

$$Q_{k,preempt} = Q_{k,block,k} \tag{9}$$

Substituting formulas (8) and (9) into (3), we obtain the closed form expression of $b(k)$, which are expressed by φ_j, μ_j $(1 \le j \le k)$ directly. Note that one-dimensional Markov chains are used to calculate the blocking probability of each class. The corresponding computation complexity is reduced dramatically compared with solving a multi-dimensional Markov chain. Next we will give the concrete models for two- and three-service systems, both of which clarify the detailed construction, blocking calculations and the excellent extensibility of the proposed model.

B. Example I: the model of the two-service system

As shown in Fig. 4, the model of the two-service system is built as two levels of one-dimensional Markov chains. The $1st$ level shows an $M/M/N/N$ Erlang loss model, which presents all service states of class 1 users. For any state i_1 $(0 \le i_1 < N)$ of class 1, the $2nd$ level has a respective one-dimensional Markov chain, which gives all possible states of class 2 when i_1 resource units are busy serving class 1 users currently. Hence the level 2 has N different one-dimensional Markov chains.

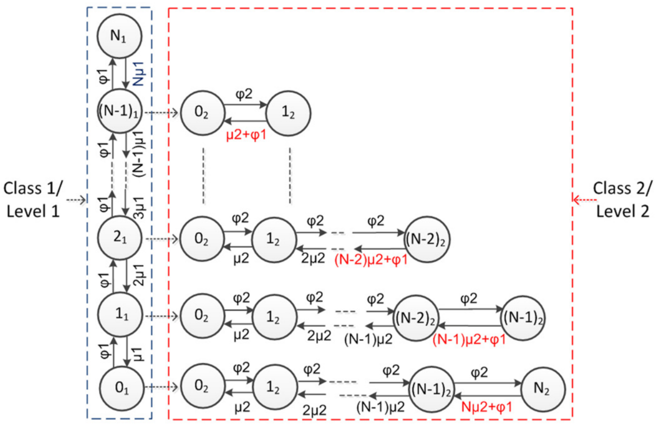

Figure 4. The model of the two-service system with capacity of N common resource units

When calculating the blocking probabilities, we use the closed form expressions directly. For class 1, its $b(1)$ is given by Erlang loss formula (i.e., formula (2)). For class 2, according to formulas (4) and (5), $Q_{2,block,1} = Q_1(N_1)$, $Q_{2,block,2} = \sum_{i_1=0}^{N-1} Q_1(i_1)Q_2(N-i_1)$.

Since formulas (3), (8) and (9),

$$b(2) = Q_1(N_1) + \frac{(\varphi_1+\varphi_2)}{\varphi_2} * \sum_{i=0}^{N-1} Q_1(i_1) * Q_2(N-i_1) \tag{10}$$

where $Q_1(i_1)$, $Q_2(i_2)$ are directly obtained by formula (7).

C. Example II: the model of the three-service system

Fig. 5 shows the analytical model of the three-service system. Compared with the model of two-service system in Fig. 4, this model has one more level of one-dimensional Markov chains, which presents the service states of class 3 corresponding to all possible combinations of the states in first two levels. It is noticeable that, when we extend the model of two-service system into a new model of three-service system, we only need to add one more level of one-dimensional Markov chains. In addition, due to the preemptive scheduling, we also only need to calculate the blocking of the additional class 3, while the blocking expressions of the first two classes are not affected and kept unchanged. These shows the excellent extensibility of the proposed model.

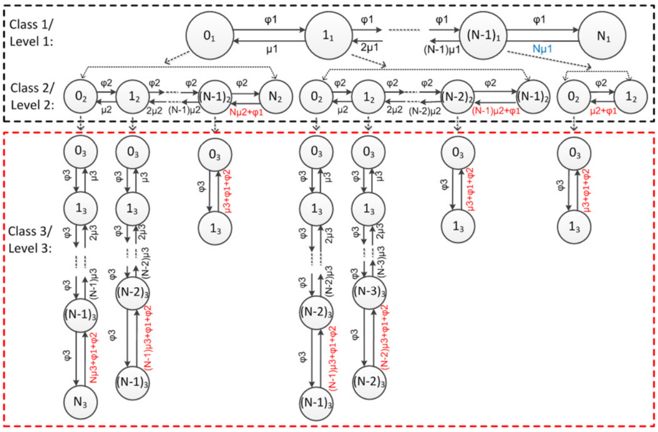

Figure 5. The model of the three-service system with capacity of N common resource units

For class 3 with the lowest priority, formulas (4)-(6) imply, $Q_{3,block,1} = Q_1(N_1)$, $Q_{3,block,2} = \sum_{i_1=0}^{N-1} Q_1(i_1)Q_2(N-i_1)$, $Q_{3,block,3} = \sum_{i_1=0}^{N-1}\sum_{i_2=0}^{N-i_1-1} Q_1(i_1) * Q_2(i_2) * Q_3(N-i_1-i_2)$.

According to formulas (3), (8) and (9),

$$b(3) = Q_1(N_1) + \sum_{i_1=0}^{N-1} Q_1(i_1)Q_2(N-i_1) + \left(\frac{\varphi_1+\varphi_2+\varphi_3}{\varphi_3}\right) * \sum_{i_1=0}^{N-1}\sum_{i_2=0}^{N-i_1-1} Q_1(i_1) * Q_2(i_2) * Q_3(N-i_1-i_2). \tag{11}$$

where $Q_1(i_1)$, $Q_2(i_2)$, $Q_3(i_3)$ are given by equations (7).

IV. SIMULATION AND ANALYTICAL RESULTS

In this section we evaluate the accuracy of the proposed model by simulations. Two- and three-service scenarios are considered. The simulator was built in the Discrete Event Modeling on Simula (DEMOS) software [13]. Ten independent simulations were performed for each parameter setting. For all simulation results we have plotted the error-bars giving the results with 95% confidence. The analytical results are obtained using formulas (1)-(11).

A. Two-service system

We consider a two-service communication system with 32 common resource units ($N = 32$). The total traffic ($A = A_1 + A_2$) offered by two service classes is varied from 0.1 to 1 ($0.1 \le A \le 1$). We use $S1$, $S2$ to denote the relative load value of two classes ($S1 = A_1/A$, $S2 = A_2/A$), and let $T1$, $T2$ to denote their mean holding times ($T1 = 1/\mu_1$, $T2 = 1/\mu_2$). In this this we consider the same mean holding times of different service classes ($T1 = T2 = 1.184 * 10^{-6}$ s).

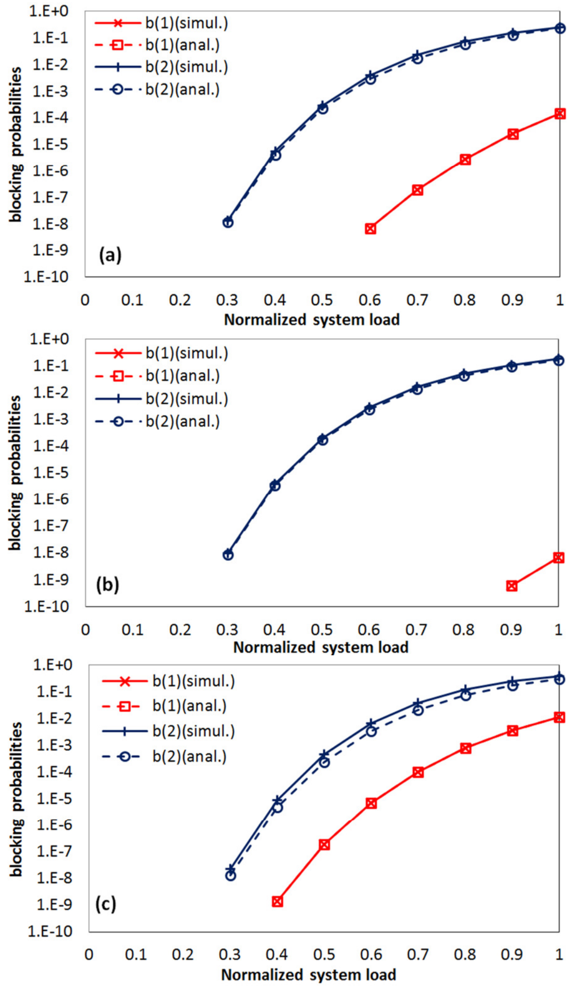

Figure 6. The blocking probabilities in a two-service system for different load allocations ($T1 = T2 = 1.184 * 10^{-6}$ s). (a). the same relative load values ($S1 = S2 = 0.5$). (b). the different relative load values ($S1 = 0.3, S2 = 0.7$). (c). the different relative load values ($S1 = 0.7, S2 = 0.3$).

Fig. 6 shows the blocking probabilities of two classes under different load allocations. We first keep $S1 = S2$ in Fig. 6(a), then change them as $S1 = 0.3, S2 = 0.7$ in Fig. 6(b) and $S1 = 0.7, S2 = 0.3$ in Fig. 6(c). Both simulation and analytical results are shown. The most important observation is that the analytical values approximate the simulation results very well under different system scenarios. We also observe that for class 1, both results overlap with each other completely for the same system load. This validates the accuracy of the analytical model, at least for the calculation of $b(1)$. Meanwhile, $b(1)$ only depends on the value of $S1$, it increases as $S1$ grows, as shown in Fig. 6(a) and (c). And it diminishes as $S1$ decreases, as shown in Fig. 6(a) and (b). This can be explained by the closed form expression of $b(1)$ in formula (2). Furthermore, although the analytical results of class 2 are very close to that of simulation for certain system load, it always produces a little smaller value, especially when $S1$ is larger than $S2$, as shown in Fig. 6(c). The reason is that when we consider the preemptive scheduling in analytical model, we use the arrival rate of class 1 to approximate its preemption probability on class 2 in the respective state. This can be seen from the outgoing transition probability of the last state in each Markov chain of level 2, as shown in Fig. 4. However, this approximation is not accurate, and the corresponding discrepancy will increases as the relative arrival rate of class 1 increases. Hence under the same system load Fig. 6(c) has larger discrepancy than Fig. 6(a) and (b), in which the discrepancy is so small and can be neglected.

B. Three-service system

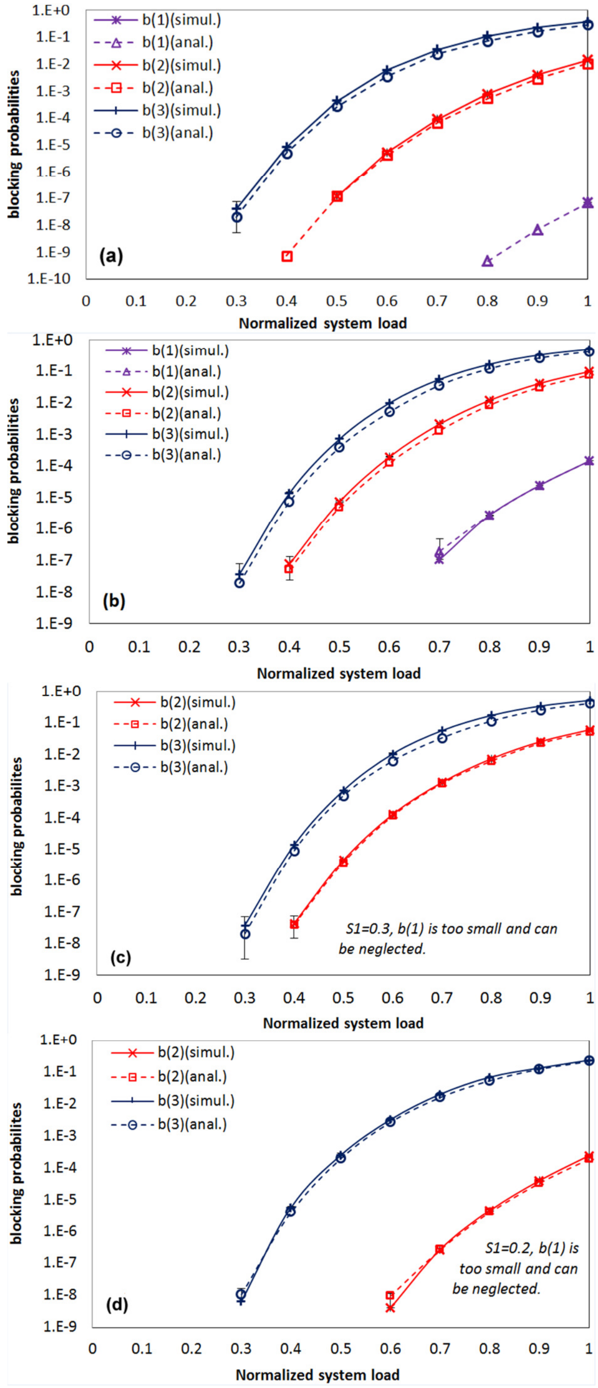

Figure 7. The blocking probabilities in a three-service system for different load allocations ($T1 = T2 = 1.184 * 10^{-6}$). (a). the same relative load values ($S1 = S2 = S3 = 1/3$). (b). the different relative load values ($S1 = 0.5, S2 =$

$0.3, S3 = 0.2$). (c). the different relative load values ($S1 = 0.3, S2 = 0.5, S3 = 0.2$). (d). the different relative load values ($S1 = 0.2, S2 = 0.3, S3 = 0.5$).

In this subsection we evaluate the proposed approximation model under a three-service scenario, i.e., class 1 has the highest priority and class 3 has the lowest priority. Same as in part A of this Section, we use $S1$, $S2$ and $S3$ ($S1 = A_1/A$, $S2 = A_2/A$, $S3 = A_3/A$, $A = A_1 + A_2 + A_3$) to denote the relative load value of three classes and let $T1, T2, T3$ ($T1 = T2 = T3 = 1.184 * 10^{-6}$ s) to denote their mean holding times. Fig. 7(a), (b), (c) and (d) present the results under different parameter settings. Both simulation and analytical results are shown. In order to further evaluate the accuracy of the propose model and get the influence of the different load allocations of three classes on the performance of the studied system, we first keep $S1 = S2 = S3$, the results are shown in Fig. 7(a), and then change the value of $S1{:}S2{:}S3$ as 5:3:2 (Fig. 7(b)), 3:5:2 (Fig. 7(c)) and 2:3:5 (Fig. 7(d)).

A number of observations can be done based on Fig. 7. The most important one is that the analytical values approximate the simulation results very well under different scenarios. This verifies the high accuracy of the proposed analytical model. We also observe that the analytical model always provides accurate $b(1)$ values for all consideration scenarios. Note that the discrepancies of class 1 in Fig. 7(b) and (d) resulted from the limited simulation times. Meanwhile, the $b(1)$ values only depend on the traffic load of class 1. As shown in Fig. 7(c) and (d), the $b(1)$ values are too small and can be neglected when $S1$ is not larger than 0.3. However, they will of course increase as $S1$ grows. In Fig. 7(b), when $S1 = 0.5$, $b(1)$ increases and the corresponding values are clearly shown. This can be explained by formula (2), the value of $b(1)$ is dominated by $S1$ for a constant N. For class 2 and 3, their blocking probabilities are only affected by the traffic pattern of the classes with same and higher priority, while not affected by the lower priority classes. Due to the operation of preemptive scheduling, the larger relative load value of one certain service class will lead to higher blocking probabilities of lower priority service classes. However, it cannot influence the blocking probabilities of higher priority service classes. As shown in Fig. 7(b) and (c), when $S1$ is increased to 0.5 while $S2$ is still 0.3, we can see the value of $b(2)$ increases over two orders of magnitude. However, comparing Fig. 7(b) and (d), when $S3$ is increased to 0.5 while keeping $S2$ unchanged, $b(2)$ decreases a lot due to the corresponding decrease in S1. For the lowest priority class 3 the blocking probability depends on the total load value of all other classes, independent of their relative load allocations. As shown in Fig. 7(b) and (c), we can see that the $b(3)$ value is kept the same under the same system load, even if the allocations of $S1$, $S2$ are different.

In addition, same as discussed in part A of this Section, although the analytical values approximate the simulation results very well under different scenarios, the proposed analytical model always produces smaller values for class 2 and 3. The reason is we made an important approximation for this model: for one service class, its preemption probability is equal to the arrival intensity of all higher priority classes. Hence this model offers high degree accurate blocking probabilities for the studied three-service system under small $S1$. Otherwise, it produces smaller values for both class 2 and 3, as shown in Fig. 7(a) and (b). Meanwhile, for class 3 with lowest priority, the corresponding discrepancy decreases as $(S1 + S2)$ diminishes. As shown in Fig. 7(d), when $S1 + S2 \geq 0.5$, the discrepancy is so small and can be neglected.

V. CONCLUSION

In this paper, we propose a novel analytical approximation model to investigate the performance of multi-service communication systems with preemptive scheduling. By using the conditional partitioning method, the proposed model builds multiple levels of one-dimensional Markov chains. Each level presents all possible service states of one service class. The corresponding blocking probability is calculated using the one-dimensional Markov chains of all higher levels as well as its own level. Its closed form expression can be derived directly and is shown in the paper. We also give the concrete models for both the two- and three-service case systems. Furthermore, the proposed model is evaluated by simulations. Both two- and three-service scenarios are considered. The results show that this model provides satisfactory approximation results under different scenarios. An additional observation is that for the lowest priority service class, its blocking probability depends on the total load value of all higher priority classes, independent of the allocation of their relative load values.

REFERENCES

[1] N. Stol, C. Raffaelli and M. Savi, "3-Level Integrated Hybrid Optical Network (3LIHON) to Meet Future QoS Requirements," *IEEE GLOBECOM 2011*, Houston, Texas, USA, Dec. 2011.

[2] C.J. Fidge, "Real-time schedulability tests for preemptive multitasking," *Computer Science*, vol. 14, no. 1, pp. 61-93, 1998.

[3] K. Lakshmanan, R. Rajkumar and J.P. Lehoczky, "Partitioned fixed-priority preemptive scheduling for multi-core processors," *In Proceedings of the 21st Euromicro Conference on Real-time Systems,* Dublin, 2009.

[4] S. Vestal, "Preemptive scheduling of multi-criticality systems with varying degrees of execution time assurance," *In Proceedings of the 28th IEEE Internationsl Real-time Systems Symposium,* Tucson, 2007.

[5] A. A. Fredericks, "Congestion in blocking systems-a simple approximation technique," *The Bell System Technical Journal*, vol. 59, no. 6, pp.805-827, 1980.

[6] E. Arthurs, J.S. Kaufman, Sizing a message store subject to blocking criteria, *In Proceeding of performance of data communications systems and their applications*, Amsterdam, 1981.

[7] J.W. Roberts, V. Mocci and I. Virtamo. "Broadband Network Teletraffic," Final Report of Action, Springer, Berlin, 1996.

[8] H. Øverby, N. Stol. "Evaluation of QoS differentiation Mechanism in Asynchronous Bufferless Optical Pakcet-Switched Networks," *IEEE Communication Magazine*, vol.44, pp. 52-57, 2006.

[9] H. Øverby, N. Stol. "Providing QoS in OPS/OBS networks with the Preemptive Drop Policy," *In Proceedings of the 3rd International Conference on Networking (ICN),* vol. 1, pp. 312-319, 2004.

[10] B. Kim, S. Lee, Y. Choi and Y. Cho, "An efficient preemption-based channel scheduling algorithm for service differentiation in OBS networks," *Computer Communications*, vol. 29, pp. 2348-2360, 2006.

[11] M. Stasiak, M. Glabowski, "A simple approximation of the link model with reservation by a one-dimensional Markov chain," *IEEE Performance Evaluation,* vol. 41, pp. 195-208, 2000.

[12] H. Akimaru, K. Kawashima, "Teletraffic theory and applications," Springer-Verlag, 1993.

[13] G. Birtwistle. "Demos-a system for Discrete Event Modelling on Simula," University of Sheffield, England S1 4DP, 2003.

Resource Allocation Method Based on QoE for Multiple User Types

Tatsuya Yamazaki
Graduate School of Science and Technology
Niigata University
Niigata, Japan
Email: yamazaki@ie.niigata-u.ac.jp

Takumi Miyoshi
Graduate School of Engineering and Science
Shibaura Institute of Technology
Saitama, Japan
Email: miyoshi@shibaura-it.ac.jp

***Abstract*—Both of Quality of Service (QoS) and Quality of Experience (QoE) are defined to specify the degree of service quality. In some sense, QoE includes subjective evaluation from the users as an extension of QoS. Therefore, feedback from QoE to QoS control might realize user-oriented resource allocation in network services. Utility functions are sometimes used to assign a bridging role between QoS and QoE. Although the user characteristic has variety, a single utility function was used in previous studies in most cases. Moreover, the QoS control, i.e., the resource allocation, by making use of the utility functions are hardly studied yet. In this paper, multiple user types which have respective utility functions are considered. Respective utility functions are acquired from real experiments. Then a resource allocation method is proposed to reflect each user type satisfaction based on the utility functions. The simple case is studied and the resource allocation method is derived analytically.**

Keywords*—*Quality of Experience, Quality of Service, utility function, resource allocation, user type.

I. INTRODUCTION

Network infrastructure is a requisite for our business and ordinary life and it provides us Web service, video streaming, Social Network Service (SNS), video meeting, and so on. Available network throughput is increasing owing to the progress of technologies, while user demand for network capability is also growing year in and year out. Therefore, adequate network resource allocation is one of the problems that network operators consider [1], where maximization of the user satisfaction with minimum cost is a goal.

In order to attain this goal, a quality of experience (QoE) based approach is regarded as a promising way to introduce the user satisfaction [2]-[4]. Compared with quality of service (QoS), QoE includes more subjective factor and presents more comprehensive evaluation. In other words, concept of QoE is suitable to reflect the degree of user satisfaction, since it is defined as “the overall acceptability of an application or service, as perceived subjectively by the end-user” in [5]. A popular way to obtain QoE evaluation is a subjective experiment such as the mean opinion score (MOS).

Several studies use utility functions in order to implement QoE factors [2]-[4],[6],[7]. The term “utility” is initially defined as the total satisfaction received from consuming a good or service in economics. Its concept can be extended to the network service and it is used to allocate resources in connection with economic approaches from the viewpoint of the users. Schroeder *et al.* [4] used the game theory, where they proposed an auction algorithm to determine the resource allocation. Ogino *et al.* [6] defined user terminal utility functions and allocated terminal resources by negotiation regarding QoS. These previous studies, however, assume the same utility function for all of the users and the application case of this assumption is considered to be rare.

Reference [3] categorized the users into three types and proposed the utility function that estimates the user satisfaction for different applications. Simulations were carried out to compute the user satisfaction. Application of the utility function to the resource allocation problem is still an open issue. Yamazaki *et al.* [8] also presented categorization of the users into four groups. The categorization seems to be realistic and credible because it is based on the real experimental data. The QoS control, i.e., the resource allocation, by the utilization of QoE factors remains as a future study.

In this paper, it is assumed that there are multiple user types, which have respective utility functions for a particular service. QoE evaluation differs from each other for different utility functions. A novel resource allocation method is proposed to reflect each user type satisfaction based on the utility functions. The simple case is studied and the resource allocation method is derived analytically.

The rest of this paper is organized as follows. Section II introduces the notation used in this paper and the problem setting. In Section III, the utility function is defined as the function mapping QoS parameters to the degree of user satisfaction and concrete utility functions are presented. In Section IV, two types of users are considered as a simple case and analyzed solutions are derived for such a case study. Section V presents computation results in order to evaluate the analyzed solutions and Section VI concludes this paper.

II. NOTATION AND PROBLEM STATEMENT

The symbols and variables used hereafter are explained in this section.

The situation considered in this paper is that the users are at the same task requiring the same traffic on a network segment. The number of all users is N_{ALL} and the users can be categorized into N_{TYPE} ($1 \leq N_{TYPE} \leq N_{ALL}$) user types. N_n ($n = 1, 2, \ldots N_{TYPE}$) is the number of users who belong

 ISBN: 978-1-61208-322-3

to the user type n. N_{ALL}, N_{TYPE} and N_n are integers.

$$N_{ALL} = \sum_{n=1}^{N_{TYPE}} N_n. \tag{1}$$

The ratio of the user number in the user type n to the number of all users is defined as R_n.

$$R_n = N_n/N_{ALL} \quad \text{for} \quad n = 1, 2, \ldots N_{TYPE}. \tag{2}$$

The utility function is specified for each user type. The utility function is the function that maps a QoS parameter or QoS parameters to the degree of user satisfaction. For the task, each user downloads a file of size S_{DATA} (bits) and the users have to utilize the same network link of bandwidth B_{ALL}, which is the bottleneck of communication in the problem. Hereinafter, the network bandwidth is considered as the QoS parameter and it is related with the waiting time of data download that affects QoE evaluation. B_n is introduced to denote the bandwidth allocated to one user in the user type n. Hence, the waiting time of data download for a user in the user type n is defined as T_n,

$$T_n = S_{DATA}/B_n \quad \text{for} \quad n = 1, 2, \ldots N_{TYPE}. \tag{3}$$

It is also assumed that there is a bandwidth allocation function in the network segment using, for instance, the software defined networking (SDN) technology such as OpenFlow [9]. Namely the network can assign a necessary network capacity for each user. The problem is how to determine the network resource allocation B_n using the utility functions related with user QoE evaluation.

III. Utility Functions

As above-mentioned, the utility function is considered to be the function mapping a QoS parameter or QoS parameters to the degree of user satisfaction. In general, the utility functions are difficult to obtain.

Regarding the problem concerned, the bandwidth allocation B_n influences the user satisfaction. The smaller B_n , the longer the download waiting time becomes. Since human beings are sensitive to time [10], the user satisfaction might be ruled by the waiting time. Therefore, the experiments to measure the user QoE are executed to get the utility functions under the controlled delay time.

The outline of the experiments is as follows. The experimental respondents are asked to solve simple four arithmetic operations (hereafter, they are called questions) on PC. A Web application presents the questions in sequence. After the respondent solves one question, random time delay ranging between 0 and 12 seconds is set before the next question is presented. The time delay is set as integer.

The respondents are classified into two groups. One group is instructed to solve the questions as fast and correctly as possible. It is assumed that the respondents are in busy or emergent situation, so this group is called the busy user group.

A movie playing window is provided for the other group on each PC. They are instructed to be relaxed and permitted to watch the movie during the resolution of the questions. It is assumed that the respondents in this group are in relaxed situation, so this group is called the relaxed user group.

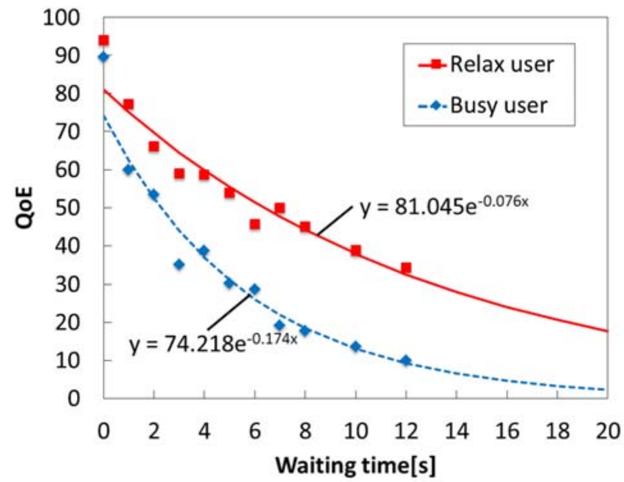

Fig. 1. The utility functions obtained by the user QoE measurement experiments.

After the calculation, the respondents are requested to answer the question "how long did you feel the transition period from one question to the next question?". The answer is recorded by means of the visual analog scale (VAS) method which uses a continuous scale in conformity with ITU-R Recommendation. BT.500-11 [11]. In the VAS method, there are five equidistant ranks of degrees on the definite length of line on the inquiry score sheet. The respondent answers his/her evaluation by marking a point on the line and the evaluation measurement of length is converted to normalized scores in the range 0 to 100. The number of respondents whose ages ranged from 18 to 23 was 31 (4 women and 27 men).

Fig. 1 is the utility functions that shows relations between the loaded time delay (waiting time) and the average values of VAS (QoE) for the busy user and relaxed user groups. From Fig. 1, different tendency is observed between two groups apparently.

Khan and Toseef proposed more generic user utility functions for real-time and non-real-time applications with respect to both technical and non-technical attributes [3]. Their utility functions were computed based on simulations. On the contrary, the more realistic utility functions are obtained through the experiments in this paper. Analysis of the utility functions in [3] and this paper is futher study.

IV. Case Study: Two User Type Analysis

In order to deduce a bandwidth allocation method, a simple case of two user types is considered. The bandwidth allocation method utilizes the utility functions to attain adaptive resource allocation. Simplification of the problem statement is just to limit the user types as two, that is all users are categorized into N_1 or N_2. To make it clearly understandable, two user types are assumed to be the busy user and relaxed user types. Thus N_1 and N_2 are denoted as N_B (the busy type) and N_R (the relax type) respectively. All of the notation in Section II are the same otherwise expressing B as the busy type and R

as the relax type. Moreover, the utility functions in Section III can be applied for the busy and relax types.

The utility functions in Fig. 1 are expressed as follows,

$$U_B(t) = C_B e^{-Q_B t}, \tag{4}$$

$$U_R(t) = C_R e^{-Q_R t}, \tag{5}$$

where C_B, Q_B, C_R and Q_R are the constant values shown in Fig. 1. Using (3) that is the relationship between the waiting time and the allocated bandwidth, (4) and (5) is transformed as follows,

$$U_B(B_B) = C_B e^{-Q_B \frac{S_{DATA}}{B_B}}, \tag{6}$$

$$U_R(B_R) = C_R e^{-Q_R \frac{S_{DATA}}{B_R}}. \tag{7}$$

Then the average utility of all users is described as $\overline{U_{ALL}}$.

$$\overline{U_{ALL}} = U_B(B_B)R_B + U_R(B_R)R_R, \tag{8}$$

where R_B and R_R are the ratios of the busy and relaxed user numbers to the number of all users respectively.

A parameter k is introduced to control balance of the utility values of two user types.

$$U_R(B_R) = k \times U_B(B_B). \tag{9}$$

When $k = 1$, both the busy and relaxed users experience the same degree of satisfaction. If k is set as 0.8, it means the relaxed users receive 20% less degree of satisfaction than the busy users.

From (5) and (9), the followings are derived.

$$kC_B e^{-Q_B \frac{S_{DATA}}{B_B}} = C_R e^{-Q_R \frac{S_{DATA}}{B_R}}. \tag{10}$$

Finally,

$$\frac{Q_B}{B_B} - \frac{Q_R}{B_R} = \frac{1}{S_{DATA}} \ln(k\frac{C_B}{C_R}). \tag{11}$$

On the other hand, summation of the bandwidth shared by each user becomes the total bandwidth.

$$B_B \cdot N_B + B_R \cdot N_R = B_{ALL}. \tag{12}$$

From (11) and (12), B_B is deduced by eliminating B_R,

$$\frac{Q_B}{B_B} - \frac{N_R \cdot Q_R}{B_{ALL} - B_B \cdot N_B} = \frac{1}{S_{DATA}} \ln(k\frac{C_B}{C_R}). \tag{13}$$

The right side of (13) can be regarded a constant value and it is replaced as C',

$$C' = \frac{1}{S_{DATA}} \ln(k\frac{C_B}{C_R}). \tag{14}$$

Using (14), (13) is regarded as a quadratic equation of B_B. Then the following equation is derived for B_B,

$$B_B = \frac{V(N_B) \pm \sqrt{V(N_B)^2 - 4C'N_B Q_B B_{ALL}}}{2C'N_B} \quad (N_B \neq 0), \tag{15}$$

where

$$V(N_B) = (Q_B - Q_R)N_B + Q_R N_{ALL} + C'B_{ALL}. \tag{16}$$

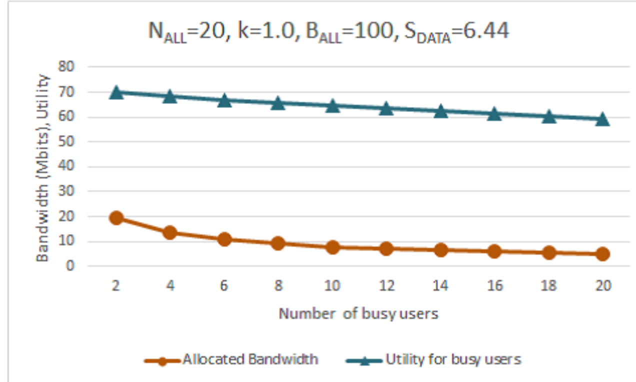

Fig. 2. Computation results ($N_{ALL} = 20, k = 1.0, B_{ALL} = 100, S_{DATA} = 6.44$).

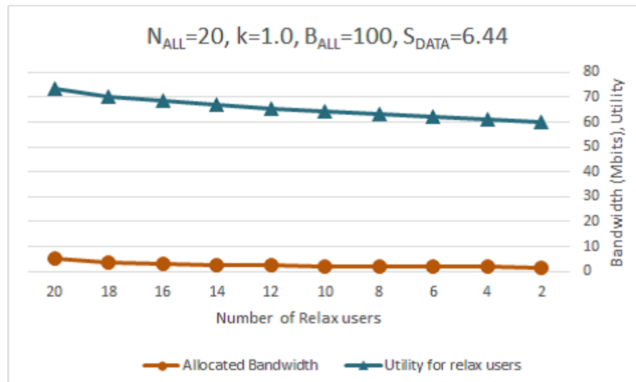

Fig. 3. Computation results ($N_{ALL} = 20, k = 1.0, B_{ALL} = 100, S_{DATA} = 6.44$).

In the same way, B_R is derived as

$$B_R = \frac{-W(N_R) \pm \sqrt{W(N_R)^2 + 4C'N_R Q_R B_{ALL}}}{2C'N_R} \quad (N_R \neq 0), \tag{17}$$

where

$$W(N_R) = (Q_R - Q_B)N_R + Q_B N_{ALL} - C'B_{ALL}. \tag{18}$$

From (15) and (17), the amounts of allocated bandwidth are calculated for the busy and relaxed user types. k is the parameter to control the QoE degrees of two user types.

V. RESULTS

This section evaluates the analyzed results obtained in Section IV.

In the first place, the case of $N_{ALL} = 20$ is presumed. The task is that each user downloads a Web content whose size is 6.44 Mbits. The size is an average value of five contents actually retrieved from a news Web site. It is also presumed that $B_{ALL} = 100.0$ (Mbits/s) and $k = 1.0$. It means a small office is assumed and the same utility is assigned to all users in the busy and relaxed user types.

Figs. 2 and 3 present allocated bandwidths and attained user utilities when the rates of busy and relaxed users change. In Fig. 2, the allocated bandwidths and the utilities for the busy users are shown, where the horizontal axis indicates the

Copyright (c) IARIA, 2014. ISBN: 978-1-61208-322-3

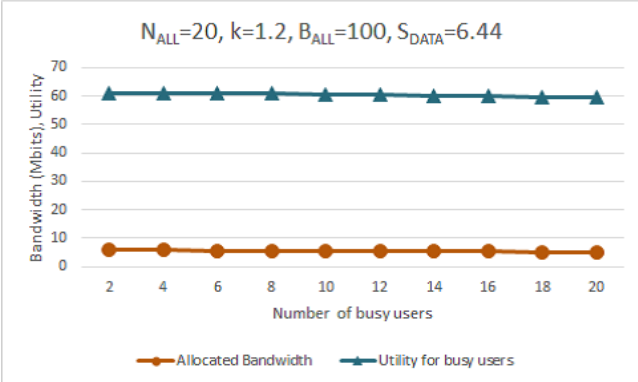

Fig. 4. Computation results ($N_{ALL} = 20, k = 1.2, B_{ALL} = 100, S_{DATA} = 6.44$).

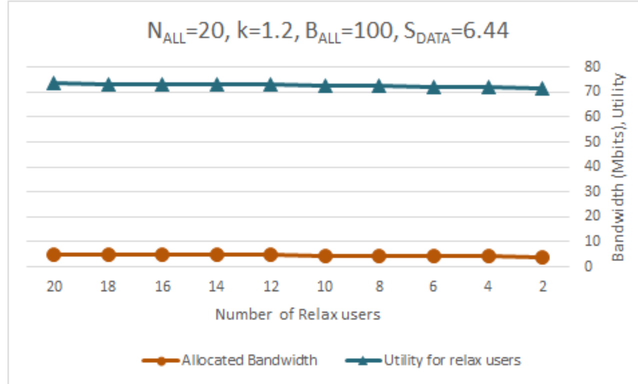

Fig. 5. Computation results ($N_{ALL} = 20, k = 1.2, B_{ALL} = 100, S_{DATA} = 6.44$).

number of the busy users and the vertical axis indicates both the allocated bandwidth in Mbits and the utility simultaneously. Fig. 3 presents the same results but from the viewpoints of the relaxed users. The horizontal axis indicates the number of the relaxed users in descending order, because it is corresponding to the ascending order of the horizontal axis in Fig. 2. The indications of the axes are set in the same way for the following figures.

From Figs. 2 and 3, it is shown that the same utilities are attained at any rate of the users since the utility balance control parameter k is set to 1.0. The bandwidth allocated to the relaxed users is rather stable, while the busy users are somewhat greedy since a small number of busy users tend to occupy the bandwidth.

Under the same conditions, the utility balance control parameter k is changed into 1.2, that is the utility for the relaxed users is set 20% more than the busy users. The results are shown in Figs. 4 and 5. Actually the utilities are always set higher for the relaxed users. Still the utilities for the busy users are kept around 60 which is not so bad value for waiting. It is noted that these conditions are very stable for any rate of the users. Fairness might be attained at these conditions.

Regarding fairness, Jain *et al.* [12] studied the fairness definition widely and expressed that fairness implies equal allocation of resources. It should be noticed that they dealt with QoS-level resource allocation fairness and an allocation metric differs among researchers. At QoE-level, the definition of fairness can be extended and the user utility is selected as the allocation metric.

Next, the parameters are set as more considerable values to evaluate the larger scale case in the user number and the network size. The scale is extended as $N_{ALL} = 100$, $B_{ALL} = 100.0$ (Mbits/s) and $S_{DATA} = 200.0$ (Mbits).

Figs. 6 and 7 present the results of $k = 1.0$, that is the case of equally-balanced utilities. Note that scaling of the vertical axes is different, because the busy users are greedy. The utilities of two user types are, however, kept to be even.

Next, the utility balance is changed as $k = 1.2$. The relaxed users' utilities are increased by 20% and the results are shown in Figs. 8 and 9. The allocated utilities are stable in spite of the user type ratios, while the allocated bandwidths decrease

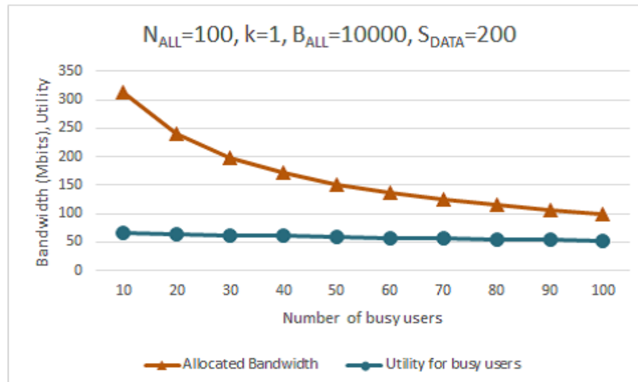

Fig. 6. Computation results ($N_{ALL} = 100, k = 1.0, B_{ALL} = 10000, S_{DATA} = 200$).

as the number of the busy users increases. Although it can be said that fairness is kept for two user types, the characteristics of the user types are not considered so well.

Finally, the computation results for $k = 0.8$ are presented in Figs. 10 and 11. These results tell that the priviledged utilities for the busy users are protected compared with the other results. The utilities are provided from 52.4 to 70.4 for the busy users, though it might be difficult to read the values from Fig. 10. One critical point is that too much bandwidth

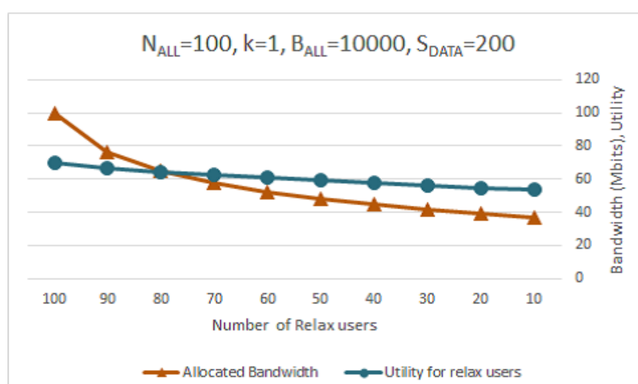

Fig. 7. Computation results ($N_{ALL} = 100, k = 1.0, B_{ALL} = 10000, S_{DATA} = 200$).

 ISBN: 978-1-61208-322-3

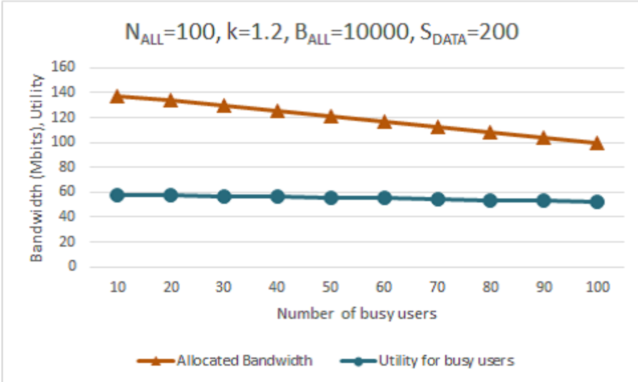

Fig. 8. Computation results ($N_{ALL} = 100, k = 1.2, B_{ALL} = 10000, S_{DATA} = 200$).

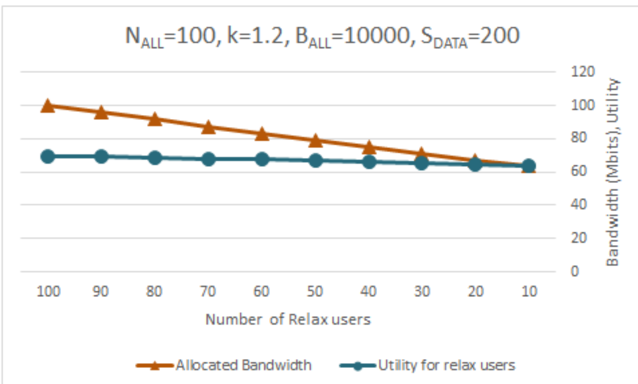

Fig. 9. Computation results ($N_{ALL} = 100, k = 1.2, B_{ALL} = 10000, S_{DATA} = 200$).

allocation might occur in the case of small number of the busy users.

VI. Conclusion

The utility functions should be different for each user type, where the utility function is defined as the function mapping QoS parameters to the degree of user satisfaction (QoE). Based on this premise, a novel resource allocation method using QoE factors is proposed in this paper. The detailed analysis is carried out for the case of two user types. By making use of the analyzed solutions, the resource allocation is derived for each user type and the utilities attained for the users are correctly derived. The analysis uses the user utility functions obtained from real user experiments. Several computation results are presented to prove the correctness of the solutions.

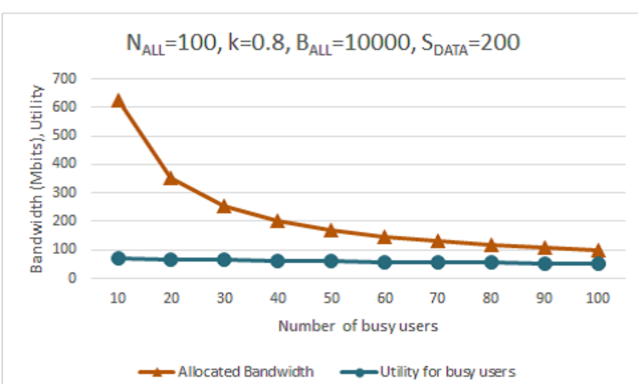

Fig. 10. Computation results ($N_{ALL} = 100, k = 0.8, B_{ALL} = 10000, S_{DATA} = 200$).

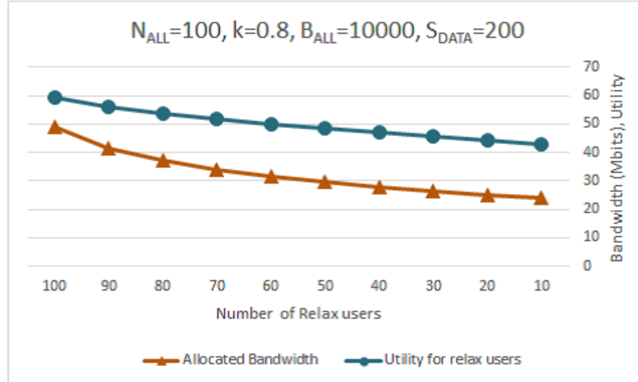

Fig. 11. Computation results ($N_{ALL} = 100, k = 0.8, B_{ALL} = 10000, S_{DATA} = 200$).

Future works include introduction of more general analysis for the cases of various user types. Moreover, implementation of the proposed method by e.g. the SDN technology is needed to evaluate adaptability of the proposed method to real situations.

References

[1] T. M. Stoenescu and D. Teneketzis, "Decentralized resource qllocation mechanisms in networks: realization and implementation," in Advances in Control, Communication Networks, and Transportation Systems, pp. 225-263, Birkhauser, Boston, 2005.

[2] P. Ameigeiras, J. J. Ramos-Munoz, J. Navarro-Ortiz, P. Mogensen, and J. M. Lopez-Soler, "QoE oriented cross-layer design of a resource allocation algorithm in beyond 3G systems," Computer Communications, 33, pp. 571-582, March 2010.

[3] M.A. Khan and U. Toseef, "User utility function as Quality of Experience (QoE)," the Tenth International Conference on Networks (ICN 2011), pp. 99-104, Jan. 2011.

[4] D. Schroeder, A. El Essaili, E. Steinbach, Z. Despotovic, and W. Kellerer, "A Quality-of-Experience driven bidding game for uplink video transmission in next generation mobile networks," IEEE International Conference on Image Processing (ICIP 2012), Orlando, Florida, USA, Sept. 2012.

[5] ITU-T Recommendation P.10/G.100 Amendment 1, "New appendix I - definition of Quality of Experience (QoE)," Jan. 2007.

[6] N. Ogino, M. Kosuga, T. Yamazaki, and J. Matsuda, "A model of adaptive QoS management platform based on cooperation of layered multi-agents," GLOBECOM'99 (Global Telecommunications Conference), pp. 406-413, Rio de Janeiro, Brazil, Dec. 1999.

[7] V. Gazis, N. Houssos, A. Alonistioti, and L. Merakos, "On the complexity of "always best connected" in 4G mobile networks", IEEE Semiannual Vehicular Technology Conference (VTC Fall 2003), Orlando, Florida, USA, 2003.

[8] T. Yamazaki, M. Eguchi, T. Miyoshi, and K. Yamori, "A service quality coordination model bridging QoS and QoE," 20th International Workshop on Quality of Service (IEEE/ACM IWQoS 2012), Coimbra, Portugal, June 2012.

[9] P. Georgopoulos, Y. Elkhatib, M. Broadbent, M. Mu, and N. Race, "Towards network-wide QoE fairness using OpenFlow-assisted adaptive video streaming," Proc. of the Workshop on Future Human-Centric Multimedia Networking (FhMN 2013) (an ACM SIGCOMM workshop), Hong Kong, Aug. 2013.

[10] O. Huber and U. Kunz, ”Time pressure in risky decision making: effect on risk defusing,” Psychology Science, vol. 49, no. 4, pp. 415-426, 2007.

[11] ITU-R Recommendation: BT.500-11, ”Methodology for the subjective assessment of the quality of television pictures,” 2002.

[12] R. Jain, D. Chiu, and W. Hawe, ”A quantitative measure of fairness and discrimination for resource allocation in shared systems,” DEC, Tech. Rep., 1984.

An Algorithm for Combinatorial Entropy Coding

Stephan Bärwolf

Integrated Communication Systems Group
Ilmenau University of Technology
Ilmenau, Germany
stephan.baerwolf@tu-ilmenau.de

***Abstract*—Entropy coding (esp. order-0) was one of the first techniques for lossless data compression, dating back to the invention of modern information theory. Over such a long period of time different schemes were invented and entropy coding has experienced various improvements: Huffman published its minimal tree structured codes and then Witten, Neal and Cleary presented a scheme leading to even better results.**
While entropy compression is still used today in most of recent compression schemes, it has not lost its significance. This paper presents an encoding and its corresponding decoding algorithm not using trees or intervals to do entropy compression. Instead it derives permutations from the input which are mapped to natural numbers. Furthermore this paper gives an impression about the compression performance by comparing some first results with well known entropy compression schemes.

***Keywords*- entropy; coding; data compression**

I. INTRODUCTION

Today, nearly all lossless and even lossy compression schemes are using at least a build-in order-0 entropy coder. Because normally entropy coding is very easy to understand and very effective in compressing non uniform distributed inputs, it is a preferred "last-stage" compression technique in such schemes. Since Shannon posted his first ideas about compression, known as Shannon-Fano, in his famous paper [3], different concepts for entropy compression have emerged: Some years after Shannon, David Huffman [4] improved Shannons scheme. Still using the same concept of binary trees, Huffman changed the way of constructing the tree structure and proved it to be optimal. Finally, in 1987 a paper [5] was published, which clearified an algorithm leading to nearly always better compression results than Huffman. This breakthrough was done by using successive bisection of an interval instead of trees to generate code words.

As already mentioned, even if today's compression schemes use more advanced algorithms, one of the previous mentioned entropy coders (often Huffman) is still part of them. For example, the Microsoft LZX [7] extends the idea of LZ77 [6] by utilizing entropy compression for match-lengths and -positions via Huffman codes.
An extreme example of entropy compression used today are the Burrows-Wheeler transform (BWT) [8], its bijective version BWTS [9], and other sort transforming modifications [11]. Since the BWTs are only "transformations", the whole compression effect is done (after some intermediate processing stages) in one final entropy compression stage [10].

This paper presents a different concept for order-0 entropy compression by mappings of permutations to enumerations and vice versa. Instead of using trees or intervals, the compression results of the presented technique therefore should never be worse than the one compared to arithmetical coding.

The paper is structured as following. A simple algorithm for encoding is presented and discussed in the first section. After this section the same is done for the decoding. In section 4, some first results are presented by using the usual compression corpora ([14],[15],[16]). Finally, the paper will be closed with a conclusion/future work chapter.

II. ENCODING

The idea of encoding an input word "I" over the finite, non-empty alphabet "A" ($I \in A^*, (a_0, a_1, \ldots, a_{d-1}) \in A, a_0 < a_1 < \cdots < a_{d-1}$), is to enumerate its represented permutation under its given symbol frequencies α $([\alpha[a_0], \alpha[a_1], \ldots, \alpha[a_{d-1}]] = \alpha \in (\mathbb{N} \cup \{0\})^k, \alpha[a_i] = |I|_{a_i}, n = |I| = \sum_{k=0}^{d-1} \alpha[a_k])$.
Furthermore a_k will be synonymous with k.

Because such an enumeration would be bounded by a multinomial coefficient (1), the enumeration could be stored with only $log_2(\binom{n}{\alpha[\,]})$ instead of $n \cdot log_2(d)$ bits.

$$\frac{(\sum_{k=0}^{d-1} \alpha[k])!}{\prod_{k=0}^{d-1} \alpha[k]!} = \frac{n!}{\prod_{k=0}^{d-1} \alpha[k]!} = \binom{n}{\alpha[\,]} < d^n \tag{1}$$

For example, the word "mississippi" leads to enumeration 32592 (out of $\frac{11!}{4! \cdot 1! \cdot 2! \cdot 4!} = 34650$), where $\alpha["i"] = 4, \alpha["m"] = 1, \alpha["p"] = 2, \alpha["s"] = 4$, see table I.
A different word may produce the same result, if its permutation is the same and just its symbol frequencies α are different: The word "MISSISSIPPI" leads to same enumeration 32592.

 ISBN: 978-1-61208-322-3

TABLE I. ENUMERATIONS FOR A GIVEN α

enumeration	word
0	iiiimppssss
1	iiimippssss
...	
99	mppiisisiss
100	imppsiisiss
...	
1999	iippisisssm
...	
32591	imssissippi
32592	mississippi
32593	msisissippi
...	
34649	ssssppmiiii

Algorithm 1 shows a way to efficiently calculate such an enumeration.

First for every symbol $a = a_i$ (see line 5) a separate, partial enumeration ("$code_a$") is generated by just taking symbols larger or equal to a_i into account.
Because permutations of smaller symbols $a_j, a_j < a_i$ have already been processed, they are ignored in further iterations. Therefore, $code_a$ is a sum of binomial coefficients $\binom{n}{m}$ for every position where "a" occurs in "I". Therefore, "n" is the number of symbols (till the current processed position) greater or equal to "a" and "m", the count of already processed occurrences of "a".
This "outer" loop is done backwards in order to enable forward decoding.

The final output result, "$code$", is the combined value of all "$code_a$".

$$base_a = \prod_{i=0}^{a-1} \binom{n - \sum_{l=0}^{i-1} \alpha[l]}{\alpha[i]}, base_0 = 1 \tag{2}$$

$$\begin{aligned} code &= code_0 + \sum_{a=1}^{d-1} code_a \cdot \prod_{i=0}^{a-1} \binom{n - \sum_{l=0}^{i-1} \alpha[l]}{\alpha[i]} \\ &= \sum_{a=0}^{d-1} code_a \cdot base_a \end{aligned} \tag{3}$$

III. DECODING

In order to decode a given enumeration, first the original symbol frequencies α must be known to the decoder. In a practical application, this is either already known (due to special constructions), or has to be transmitted to the decoder separately.
Within the further text it is assumed to know the correct α before decode.

If α is known, then $n = |I|$ can be retrieved efficiently, because $n = \sum_{k=0}^{d-1} \alpha[k], d = |\alpha|$.
Knowing α also enables the decoder to calculate each $base_a$ using (2) and therefore each $code_a$ using (3). Resolving the equation (3) for $code_a$ leads to (4).

$$code_a = \left\lfloor \frac{code \bmod (\prod_{i=0}^{a} base_i)}{\prod_{i=0}^{a-1} base_i} \right\rfloor \tag{4}$$

Algorithm 1 Derive an enumeration (code) from an (input) word

Require: $msg \Leftarrow$ to be encoded message ($msg = I$)
Require: $\alpha \Leftarrow$ frequency of each character (byte) in the message
Require: $n \Leftarrow$ size of decoded message ($n = \sum_{a=0}^{255} \alpha[a]$)

```
1:  // initialize output:
2:  code ⇐ 0
3:  bytes_done ⇐ 0
4:  // process position of every value "a" individually
5:  for a = 255 down to 0 do
6:      bytes_done ⇐ bytes_done + α[a]
7:      bytes_relevant ⇐ 0
8:      bytes_processed ⇐ 0
9:      code_a ⇐ 0
10:     code ⇐ code * binom(bytes_done, α[a])
11:     // loop over message, track positions of value "a",
        ignore smaller values
12:     for msg_position = 0 to (n − 1) do
13:         if msg[msg_position] ≧ a then
14:             if msg[msg_position] = a then
15:                 bytes_processed ⇐ bytes_processed + 1
16:                 code_a ⇐ code_a + binom(bytes_relevant, bytes_processed)
17:             end if
18:             bytes_relevant ⇐ bytes_relevant + 1
19:         end if
20:     end for
21:     code ⇐ code + code_a
22: end for
23: return code
```

To retrieve a position for symbol "a" from $code_a$, the biggest position possible (with a binomial coefficient still fitting into $code_a$) has to be successively substracted, as indicated in line 17 of algorithm 2. Thanks to $\alpha[a]$ the decoder already knows the right count of positions to decode.
Because such decoded positions were positions in the set of unprocessed symbols (symbols greater or equal) they have to be transformed into a global array index (lines 19 to 26).

Finally, all indices of the "msg" array have been processed and msg contains the decoded "I".

 ISBN: 978-1-61208-322-3

Algorithm 2 Retrieve word from its enumeration

Require: $code \Leftarrow$ to be decoded number
Require: $\alpha \Leftarrow$ frequency of each character (byte) in decoded message
Require: $n \Leftarrow$ size of decoded message ($n = \sum_{a=0}^{255} \alpha[a]$)
// initialize output:
$msg \Leftarrow$ each byte filled with value 255, $|msg| = n$
$bytesleft \Leftarrow n$
// process permutation of every char individually:
for $a = 0$ **to** 255 **do**
  // retrieve permutation code for positions of value a
  $code_a \Leftarrow code \bmod \binom{bytesleft}{\alpha[a]}$
  // update code for succeeding permutations
  $code \Leftarrow code\ div\ \binom{bytesleft}{\alpha[a]}$
  $position \Leftarrow bytesleft$
  // to track the unprocessed indices within msg:
  $msg_{position} \Leftarrow n$
  $msg_{unprocessed} \Leftarrow bytesleft$
  // start decoding positions where value "a" occurs in message
  for $k = \alpha[a] - 1$ **down to** 0 **do**
    $maxpos \Leftarrow position$
    $position \Leftarrow$ find biggest m with $k \leqq m < maxpos$ and $\binom{m}{k+1} \leqq code_a$
    $code_a \Leftarrow code_a - \binom{m}{k+1}$
    // place value "a" into msg at unprocessd position "position"
    while $msg_{unprocessed} > maxpos$ **do**
      $msg_{position} \Leftarrow msg_{position} - 1$
      if $msg[msg_{position}] = 255$ **then**
        $msg_{unprocessed} \Leftarrow msg_{unprocessed} - 1$
      end if
    end while
    $msg[msg_{position}] \Leftarrow a$
  end for
  $bytesleft \Leftarrow bytesleft - \alpha[a]$
end for
return msg

IV. FIRST RESULTS

The results in table II (resp. table III) show the compressed size and the percentage to the original size of the Huffman-, arithmetical- and the presented scheme.
In order to use a set of representative files, the Canterbury [16] (resp. Calgary [15]) corpus was used.
There were no other preprocessings than the direct entropy compression with the mentioned schemes.
In all schemes it was assumed that decoders will have full apriori information about α and used this apriori information at the beginning of encoding.

For generating Huffman compressed files, the open source "libhuffman" [17] was used. This encoder operates in two phases, the first one scans the input in order to construct an optimal tree. The second phase uses this tree to compress the input byte by byte. For "libhuffman" the function storing the extra information about α (from phase one) was commented out in order to preserve comparability.

For generating arithmetical compressed files, the algorithm from Witten et al. [5] was used. It was slightly adapted to avoid using END-symbols and to work in two phases like "libhuffman" does. As in "libhuffman", the arithmetical encoding used the knowledge about α from the beginning, but also did not store any information about α to the output.

The algorithm of the presented scheme always stored $log_2(\binom{original\_size}{\alpha[\ ]})$ bit (ceiled up to the next full byte) as output. Again no information about α was put to the output.

From both tables it can be seen, that the presented scheme always is the best compressing one.

V. CONCLUSION

This paper presented a different concept for order-0 entropy compression, where mappings from permutations to enumerations are used. The paper presented also an algorithm for encoding and for decoding. First compression results were compared to two well established encoders, indicating promising compression performance since the presented scheme always is the most performant one.
Since it can be shown, that at most $(n + d) \cdot log_2(n + d) - n \cdot log_2(n) - d \cdot log_2(d)$ additional bits ($n = |I|, d = |\alpha|$) are required to be transmitted to the decoder for decoding the correct α, the presented coding scheme seems to have no issues affecting decodability.

The presented technique also offers promising possibilities for future work on this field:
Since binomial coefficients can be precalculated into some kind of cache or table, the scheme nearly hasn't any slow multiplication. Because the last remaining multiplication in line 10 of algorithm 1 could be replaced by much faster logical left-shifting - no multiplications or even divisions are necessary at all.
Furthermore and because of commutative multiplication/addition, the algorithms outer- and inner loops are good candidates for parallelization with nearly no synchronization needs and therefore nearly full speedup.

ACKNOWLEDGMENTS

The author would like to thank the "Deutsche Telekom Stiftung" Germany [18] for its funding and for its ongoing efforts, actions and activities to support education and thus research and science. Furthermore the author also would like to thank the team of the Integrated Communication Systems Group at TU-Ilmenau for its general support at all times.

TABLE II. THE CANTERBURY CORPUS [16] COMPRESSION PERFORMANCE

filename	org. size (bytes)	huffman size	huffman %	arithmetical size	arithmetical %	presented size	presented %
alice29.txt	152089	87688	57.66	86837	57.10	86788	57.06
asyoulik.txt	125179	75806	60.56	75235	60.10	75187	60.06
cp.html	24603	16199	65.84	16082	65.37	16035	65.17
fields.c	11150	7026	63.01	6980	62.60	6936	62.21
grammar.lsp	3721	2170	58.32	2155	57.91	2126	57.14
kennedy.xls	1029744	462532	44.92	459971	44.67	459779	44.65
lcet10.txt	426754	250565	58.71	249071	58.36	249008	58.35
plrabn12.txt	481861	275585	57.19	272936	56.64	272880	56.63
ptt5	513216	106551	20.76	77636	15.13	77563	15.11
sum	38240	25645	67.06	25473	66.61	25353	66.30
xargs.1	4227	2602	61.56	2589	61.25	2559	60.54

TABLE III. THE CALGARY CORPUS [15] COMPRESSION PERFORMANCE

filename	org. size (bytes)	huffman size	huffman %	arithmetical size	arithmetical %	presented size	presented %
README	2479	1492	60.19	1483	59.82	1457	58.77
bib	111261	72761	65.40	72330	65.01	72273	64.96
book1	768771	438374	57.02	435043	56.59	434981	56.58
book2	610856	368300	60.29	365952	59.91	365877	59.90
geo	102400	72556	70.86	72274	70.58	72117	70.43
news	377109	246394	65.34	244633	64.87	244555	64.85
obj1	21504	16051	74.64	15989	74.35	15868	73.79
obj2	246814	194096	78.64	193144	78.25	192971	78.18
paper1	53161	33337	62.71	33113	62.29	33058	62.18
paper2	82199	47615	57.93	47280	57.52	47228	57.46
paper3	46526	27275	58.62	27132	58.32	27084	58.21
paper4	13286	7860	59.16	7806	58.75	7768	58.47
paper5	11954	7431	62.16	7376	61.70	7334	61.35
paper6	38105	24023	63.04	23861	62.62	23808	62.48
pic	513216	106551	20.76	77636	15.13	77563	15.11
progc	39611	25914	65.42	25743	64.99	25687	64.85
progl	71646	42982	59.99	42720	59.63	42668	59.55
progp	49379	30214	61.19	30052	60.86	30000	60.75
trans	93695	65218	69.61	64800	69.16	64734	69.09

REFERENCES

[1] D. Salomon, “Data Compression The Complete Reference,” 4th ed. London: Springer, 2007.

[2] D. J. C. MacKay, “Information Theory, Inference, and Learning Algorithms,” version 6.0 Cambridge University Press, June 2003, pp. 32.

[3] C. E. Shannon, “A Matematical Theory of Communication,” Reprinted with corrections from The Bell System Technical Journal, vol. 27, October 1948, pp. 379–423, 623–656.

[4] D. A. Huffman, “A method for construction of minimum-redundancy codes,” Proceedings of the I.R.E., September 1958, pp. 1098–1101.

[5] I. Witten, R. Neal, and J. Cleary, “Arithmetic Coding for Data Compression,” Communications of the ACM, vol. 30, no. 6, June 1987, pp. 520–540.

[6] A. Lempel and J. Ziv, “A Universal Algorithm for Sequential Data Compression,” IEEE transactions on information theory, May 1977, pp. 337–343.

[7] Microsoft, “Microsoft LZX Data Compression Format,” version 4.71.410.0 Microsoft Cabinet SDK, , March 1997.

[8] M. Burrows and D. J. Wheeler, “A block-soring Lossless Data Compression Algorithm,” System Research Center, Palo Alto, USA, research report, May 1994.

[9] J. Gil and D. A. Scott, “A Bijective String Sorting Transform,” Israel/USA, July 2009.

[10] J. Abel, “Improvements to the Burrows-Wheeler Compression Algorithm: After BWT Stages,” preprint Düsburg-Essen, Germany, March 2003.

[11] M. Kufleitner, “On Bijective Variants of the Burrows-Wheeler Transform.” Prague, Czech Republic: Proceedings of PSC 2009, pp. 65–79.

[12] A. Salomaa, “Public-Key Cryptography,” 2nd ed. Finland: Springer, 1990.

[13] M. Lothaire, “Combinatorics on words,” Reading, Massachusetts: Addison-Wesley, 1983.

[14] R. Arnolds and T. Bell, “A corpus for the evaluation of lossless compression algorithms.” Christchurch, NZ: University of Canterbury.

[15] T. C. Bell and I. Witten, “Calgary compression corpus,” [retrieved: December, 2013]. [Online]. Available: ftp://ftp.cpsc.ucalgary.ca/pub/projects/text.compression.corpus/

[16] M. Powell and T. Bell, “The Canterbury corpus,” [retrieved: December, 2013]. [Online]. Available: http://corpus.canterbury.ac.nz/

[17] D. R. Richardson, “libhuffman - An Open Source Huffman Coding Library in C,” [retrieved: December, 2013]. [Online]. Available: http://huffman.sourceforge.net/

[18] http://www.telekom-stiftung.de

A Novel Aggregation Approach Based on Cooperative Agents and Self-Stabilizing Clustering for WSNs

Mandicou Ba*, Olivier Flauzac*, Rafik Makhloufi†, Leïla Merghem-Boulahia‡, Florent Nolot* and Ibrahima Niang§
* Université de Reims Champagne-Ardenne (URCA), Laboratoire CReSTIC - SysCom EA 3804, France
{mandicou.ba, olivier.flauzac, florent.nolot}@univ-reims.fr
† Ecole Nationale des Ponts et Chaussées (ENPC), Laboratoire CERMICS - SOWG, France
makhlour@cermics.enpc.fr
‡ Université de Technologies de Troyes, Laboratoire ICD - ERA (UMR CNRS 6279), France
leila.boulahia@utt.fr
§ Université Cheikh Anta Diop (UCAD), Laboratoire d'Informatique de Dakar (LID) Sénégal
iniang@ucad.sn

***Abstract*—In this paper, we present novel ideas for an intelligent aggregation approach based on cooperative agents and a self-stabilizing clustering scheme for Wireless Sensor Networks (WSNs). This aggregation solution aims at optimizing communications by combining an efficient self-stabilizing clustering approach and an intelligent cooperative aggregation process. On the one hand, the clustering approach reorganizes the network nodes into groups of nodes (i.e., clusters) so that it minimizes communications between nodes, their cluster-head and the Sink. On the other hand, the aggregation approach reduces communications by merging and sending only pertinent information to the Sink, in a distributed way.**

***Keywords-Wireless Sensor Networks*; *Self-stabilizing Clustering*; *Aggregation*; *Cooperative Agents*;**

I. INTRODUCTION

Due to their good properties and wide applications, Wireless Sensor Networks (WSNs) have known a growing interest in both industrial and academic fields. They are used for many applications like: medical, scientific, military, environmental, security, etc. In a WSN, sensor nodes have a limited energy in their battery due to their size. This energy is consumed by three main operations: sensing, communication and processing. Communication of messages is the one consuming the highest quantity of energy for a sensor. Thus, saving communication power is crucial in a WSN. It is stated in [6] that the necessary energy power to transmit a 1 KB message over a distance of 100 meters is approximately equivalent to the execution of 3 million CPU instructions by a 100 MIPS/W processor. Consequently, to extend network lifetime it is more urgent in these networks to optimize communications than processing. It is also stated in the literature that organizing the network into groups of nodes enables the control of the network communications and thus saves energy. From this, several clustering approaches are proposed. These approaches are used in the design of energy-aware routing mechanisms

Several clustering approaches are proposed in the literature and used, for example, in the case of a WSN for routing collected information to a base station (Sink). However, most of them are based on state model [10], [11], [13], [14], so they are not realistic in the context of WSNs compared to message-passing based clustering ones [7], [8], [17], [18]. Moreover, approaches in the last category are generally highly costly in terms of exchanged messages, while in the case of WSNs clustering aims at optimizing communications and energy consumption. Thus, in Ba et *al.* [2], we have proposed a self-Stabilizing Distributed Energy-Aware k-hops Clustering protocol for ad hoc networks (SDEAC). SDEAC prolongs the network lifetime by minimizing the energy consumption involved in the exchanged of messages. Furthermore, in terms of energy consumption in WSNs, data aggregation has been presented as a particularly useful capability for routing [1], [6], [12], [16], [21]. However, existing self-stabilizing clustering solutions like [7], [8], [10], [11], [13], [14], [17], [18] do not provide data aggregation mechanisms based on their clustering algorithms.

Therefore, by considering the fact that in a WSN communication of messages is by far the most important source of energy consumption, we propose in this paper a novel aggregation approach that optimizes these communications by combining an efficient self-stabilizing clustering scheme proposed in Ba et *al.* [2] and an adaptation of the intelligent cooperative aggregation process proposed in Sardouk et *al.* [19] [1]. Thus, on the one hand, the clustering approach reorganizes the network nodes into groups of nodes (i.e., clusters) so that it minimizes communications between nodes, their cluster head noted CH and the Sink. On the other hand, the aggregation approach reduces communications (in terms of number of messages and their size) by merging and sending only pertinent information to the Sink, in a distributed

[1]These works [2], [3], [4], [5], [19] are supported by Regional Council of Champagne-Ardenne and European Regional Development Fund through the CPER CapSec ROFICA project

TABLE I
THEORETICAL COMPARISON OF SDEAC WITH OTHER APPROACHES BASED ON STATE MODEL

	Stabilizing Time	Memory space per neighbor	Neighborhood
SDEAC	$n+2$	$\log(2n+k+3)$	1 hop
Datta *et al.* [11]	$O(n)$, $O(n^2)$	$O(\log(n))$	k hops
Caron *et al.* [10]	$O(n*k)$	$O(log(n)+log(k))$	k+1 hops

way. Thus, collected information about the environment is processed locally by nodes on the route before it reaches the Sink. By reducing the amount of information, important power consumption could be saved. In addition, this strategy deals with the density of the network around each node by eliminating redundancy of information.

The remainder of the paper is organized as follows. Section II summarizes SDEAC protocol. In Section III, we explain our ideas for the establishment of an intelligent cooperative multi-agent aggregation approach by using SDEAC protocol. Section IV presents the validation framework used to evaluate performances of our aggregation approach. Finally, Section V concludes this paper and presents our working perspectives.

II. SDEAC: SELF-STABILIZING DISTRIBUTED ENERGY-AWARE CLUSTERING

In [2], we have presented SDEAC: a self-Stabilizing Distributed Energy-Aware k-hops Clustering approach for ad hoc networks. SDEAC is based on message-passing model and structures the network into non-overlapping clusters with diameter at most equal to $2k$. This structuring does not require any initialization. Furthermore, it is based only on information from neighboring nodes at distance 1 contrary to other self-stabilizing clustering algorithms. SDEAC uses a unique type of message to discover the neighborhood of a node at distance 1 and to structure the network into non-overlapping *k-hops* clusters.

On the one hand, we have validated SDEAC through a formal proof and simulations in [2]. We have proved that a legal configuration (i.e., stabilization) is reached after at most $n+2$ transitions and each node u requires $(\Delta_u+1)*\log(2n+k+3)$ memory space, where n is the number of network nodes and Δ_u is the degree of u. As illustrated in Table I, we have formally compared SDEAC with self-stabilizing clustering approaches based on state model [10], [11]. Furthermore, in Ba et *al.* [3], we have compared SDEAC with one of the most referenced papers on self-stabilizing solutions based on message-passing model [17]. This shows that SDEAC reduces communication cost and energy consumption by a factor of at least 2.

On the other hand, in Ba et *al.* [4], we have proposed an evaluation study of SDEAC's cluster-head election criteria in the context of WSNs with energy constraint. For this, we consider the most important criteria: node's identity, residual energy and degree. SDEAC optimizes energy consumption and then prolongs the network lifetime by minimizing the number of messages involved in the construction of clusters and by minimizing stabilization time. It also offers an optimized structure for routing. Furthermore, SDEAC is generic and complete. It can be easily used for constructing clusters according to multiple criteria in the election of cluster-heads.

Moreover, SDEAC is fault tolerance and adapted to topological changes. In fact, in [5], a fault-tolerance mechanism has been proposed and simulation results have shown that that after faults occurrence, the re-clustering cost is minimal compared to the clustering cost.

Therefore, in this work, clusters provided by SDEAC are used for an intelligent cooperative multi-agent aggregation approach in order to reduce communications by merging and sending only pertinent information to the Sink, in a fully-distributed way.

III. INTELLIGENT CLUSTER-BASED COOPERATIVE MULTI-AGENT AGGREGATION APPROACH

We propose an intelligent aggregation approach based on the clustering mechanism SDEAC [2] described above and on a cooperative Multi-Agent System (MAS) [19]. This aggregation approach enables the communication of only pertinent sensors information to the base station (i.e., Sink) in fully-distributed way.

A. Aggregation scenario

Basically, in this approach, each sensor that collects important information sends it in the direction of its CH and then, the CH sends the aggregate in the direction of the Sink. Each sensor node that receives information from one or multiple sensors will process it by merging and combining it with its local information in order to minimizing the amount of information. Then, it sends the resulting aggregate to its intermediate node (i.e., relay node that knows the next node in route to the CH or to the Sink) in the direction of the Sink. An autonomic agent is installed on each sensor node in order to control it and to define its behavior by analyzing collected information and making decisions. This agent uses routing information to establish a one-hop situated view (or situadness) [9] and thus to maintain a knowledge base. Each agent will decide by itself according to a given strategy if it cooperates or not with other agents. By using a situated view, each node reduces the amount of maintained information by a view limited only to its direct neighbors.

When a sensor collects information about its environment, its agent decides if this one is important. If so, it initiates the aggregation process by sending a cooperation request to its neighboring agents. If a neighbor decides to cooperate, it sends its local information (i.e., sensed by its sensor) to the requesting agent. If the node that receives the cooperation request is the default intermediate node, it does not answer to the requesting node. But, in order to save time, while

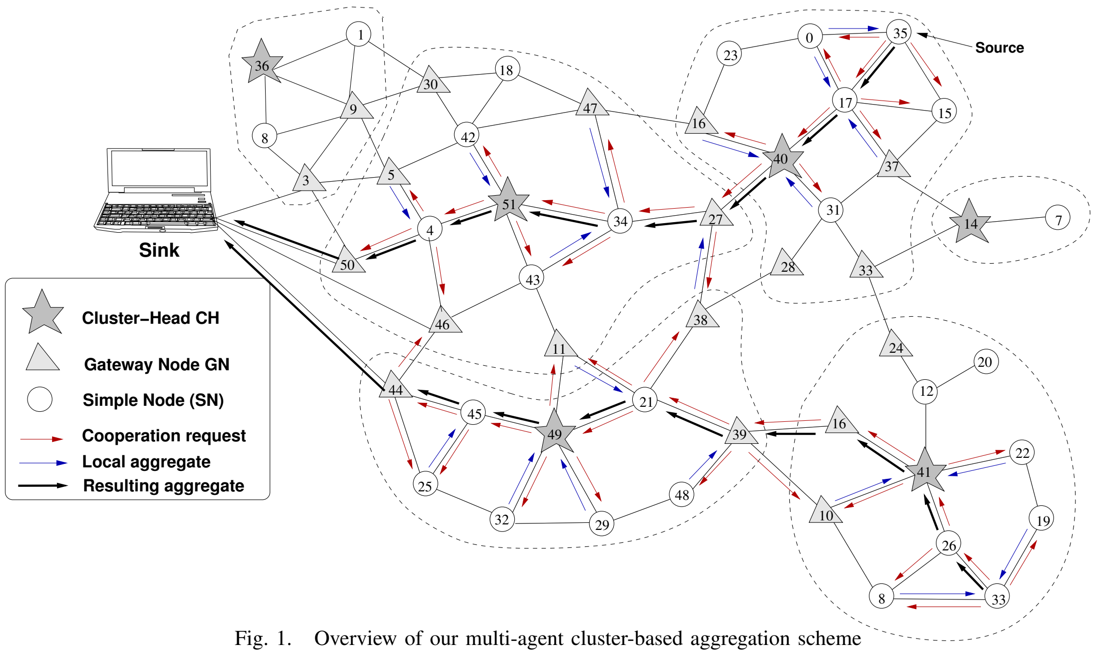

Fig. 1. Overview of our multi-agent cluster-based aggregation scheme

waiting for the aggregate of the requesting node, the relay agent directly sends a cooperation request to its neighbors.

Once the requesting node receives local state information from its neighbors, it merges the given data and eliminates redundancy before sending the result (aggregate) to its default intermediate node. The latter merges the received aggregate with the gathered information of its neighbors to compute a new aggregate and then it sends the result to its intermediate node. The aggregation process continues until the sensed information reaches the Sink.

Figure 1 illustrates one scenario of the proposed cluster-based aggregation scheme. Here, a simple source node 35 sends its sensed information to its CH (40) through node 17 and the CH sends the resulting partial aggregate in the direction of the Sink. Each node on the path aggregates data.

B. Cooperative agents

We define a distributed multi-agent cooperation strategy considering different parameters in the decision-making process [19]. These parameters are: importance level of collected information (I), level of energy power on sensors (E), position of nodes in the network (P) and network density (D). For example, when the battery power on a node is critical (i.e., less than a certain threshold) the agent of this node refuses to cooperate and will use this energy only for sensing and sending its own information.

Thus, according to the value of these parameters, an autonomic agent selects by itself the best behavior and decides to cooperate or not in the aggregation process. For this, it computes a coefficient R that denotes the relevance of cooperation like shown in equation 1.

$$R = w_e \times E + w_d \times \frac{1}{D} + w_p \times P + w_i \times I \qquad (1)$$

TABLE II
EXAMPLE OF NEIGHBORHOOD TABLE

Routing table of node 51				
Node_ID	Cluster_ID	Status	Dist_CH	Dist_Sink
4	51	SN	1	2
34	51	SN	1	4
42	51	SN	1	4
43	51	SN	1	4

where w_e, w_d, w_p and w_i are the influence factors (i.e., weights) for, respectively, energy, density, position and information importance.

C. Communication policy

The communication policy used in our aggregation is an adaptation of the routing protocol Dynamic Source Routing (DSR) [15] that limits its communications in the case of a WSN. Our approach consists in two operations: route discovery and route maintenance. Note that initially, each node knows how to reach its CH through the reutilization of clustering information. Thus, CHs need to know the route to the Sink. For this, the route discovery phase allows CHs to obtain information about their direct neighborhood and about the default intermediate node in the direction of the Sink. Each node on the path between a CH and the Sink will learn its intermediate node. Then, the route maintenance phase enables nodes to have the last updates about their neighborhood. Table II illustrates an example of a neighborhood table of a node 51 on the path between CH 40 and the Sink (Figure 1).

 ISBN: 978-1-61208-322-3

IV. Validation of the Proposed Approach

In order to validate our aggregation approach, we plan to carry out a simulation campaign under the *OMNeT++* simulator [20] when we have implemented SDEAC protocol. For this, we implement our intelligent cooperative multi-agent aggregation approach described in Section III and we validate it according to three scenarios:

i Fully-decentralized aggregation approach: aggregation is done on each node on the path to the Sink;

ii Partially-decentralized aggregation approach: aggregation is done only on cluster-heads;

iii Non-aggregated information: sensed information is directly sent to the sink without intermediary merging (aggregation).

Thus, in the case of non-aggregation scenario, during each sensing cycle, each node in the cluster sends its information to the Sink through its cluster-head. As soon as a cluster-head receives an information from a node in its cluster, it forwards this message directly to the Sink through its path. At opposed, in the case of aggregation on each cluster-head, the cluster-head collects all messages from all nodes in its cluster and aggregates data. Thus, during each sensing cycle, only one message containing all information from other cluster's nodes is send to the Sink by each cluster-head.

To validate these three scenarios (i.e., *(i), (ii) and (iii)*), we plan to evaluate our approach according to the most used criteria under the same clustering approach and testing framework:

- Communication cost: in terms of number of messages and their size;
- Energy consumption: total and distribution of battery power consumption of sensors;
- End-to-end delay: necessary time to send information from source nodes to the sink;
- Network lifetime: time until i nodes of the network die (empty battery), where we vary the value of i.

V. Conclusions and Perspectives

In this paper, we have presented novel ideas for the proposal of an intelligent aggregation approach based on cooperative agents and a self-stabilizing clustering for WSNs. The goals of this intelligent aggregation approach is to reduce communications by merging and sending only pertinent information to the Sink, all this in a distributed way.

Currently, we are working on the validation of our protocol in *OMNeT++* simulator in order to evaluate its communication cost in terms of messages, energy consumption, end-to-end delay and the network lifetime.

As future work, we plan to compare our aggregation approach with other existing aggregation solutions.

References

[1] K. Akkaya, M. Younis, and M. Youssef. Efficient aggregation of delay-constrained data in wireless sensor networks. In Proceedings of the ACS/IEEE 2005 International Conference on Computer Systems and Applications, pp. 904–909, 2005.

[2] M. Ba, O. Flauzac, B. S. Haggar, F. Nolot, and I. Niang. Self-stabilizing k-hops clustering algorithm for wireless ad hoc networks. In 7th ACM International Conference on Ubiquitous Information Management and Communication (IMCOM'13), pp. 38:1–38:10, 2013.

[3] M. Ba, O. Flauzac, R. Makhloufi, F. Nolot, and I. Niang. Comparison between self-stabilizing clustering algorithms in message-passing model. In 9th International Conference on Autonomic and Autonomous Systems (ICAS'13), pp. 27–32, 2013.

[4] M. Ba, O. Flauzac, R. Makhloufi, F. Nolot, and I. Niang. Evaluation study of self-stabilizing cluster-head election criteria in wsns. In 6th International Conference on Communication Theory, Reliability, and Quality of Service (CTRQ'13), pp. 64–69, 2013.

[5] M. Ba, O. Flauzac, R. Makhloufi, F. Nolot, and I. Niang. Fault-tolerant and energy-efficient generic clustering protocol for heterogeneous wsns. International Journal On Advances in Networks and Services, pp. 231–245, 2013.

[6] S. J. Baek, G. de Veciana, and X. Su. Minimizing energy consumption in large-scale sensor networks through distributed data compression and hierarchical aggregation. IEEE Journal on Selected Areas in Communications, pp. 1130–1140, 2004.

[7] A. Bui, A. Kudireti, and D. Sohier. A fully distributed clustering algorithm based on random walks. In 8th IEE International Symposium on pages Parallel and Distributed Computing (ISPDC '09), pp. 125–128, 2009.

[8] A. Bui, A. Kudireti, and D. Sohier. A random walk based clustering with local recomputations for mobile ad hoc networks. IEEE International Symposium on Parallel Distributed Processing, Workshops and Phd Forum (IPDPSW'10), pages 1–8, 2010.

[9] T. Bullot, R. Khatoun, L. Hugues, D. Gaïti, and L. Merghem-Boulahia. A situatedness-based knowledge plane for autonomic networking. International Journal of Network Management, pp 171–193, 2008.

[10] E. Caron, A. K. Datta, B. Depardon, and L. L Larmore. A self-stabilizing k-clustering algorithm for weighted graphs. Journal of Parallel and Distributed Computing, pp. 1159–1173, 2010.

[11] A. K. Datta, S. Devismes, and L. L. Larmore. A self-stabilizing O(n)-round k-clustering algorithm. In 28th IEEE International Symposium on Reliable Distributed Systems (SRDS '09), pp. 147–155, 2009.

[12] E. Fasolo, M. Rossi, J. Widmer, and M. Zorzi. In-network aggregation techniques for wireless sensor networks: a survey. IEEE Wireless Communications, pp. 70–87, 2007.

[13] C. Johnen and L. H. Nguyen. Robust self-stabilizing weight-based clustering algorithm. Theoretical Computer Science, pp. 581–594, 2009.

[14] Colette Johnen and LeHuy Nguyen. Self-stabilizing construction of bounded size clusters. In International Symposium on Parallel and Distributed Processing with Applications (ISPA '08), pp. 43–50, 2008.

[15] D. B. Johnson, D. A. Maltz, and J. Broch. Ad hoc networking. chapter DSR: The Dynamic Source Routing Protocol for Multihop Wireless Ad Hoc Networks, pp. 139–172. 2001.

[16] R. Makhloufi, G. Doyen, G. Bonnet, and D. Gaïti. A survey and performance evaluation of decentralized aggregation schemes for autonomic management. International Journal of Network Management (IJNM), pp 29p, 2013. To appear.

[17] N. Mitton, E. Fleury, I. G. Lassous, and S. Tixeuil. Self-stabilization in self-organized multihop wireless networks. In 25th IEEE International Conference on Distributed Computing Systems Workshops (ICDCS'05), pp. 909–915, 2005.

[18] N. Mitton, A. Busson, and E. Fleury. Self-organization in large scale ad hoc networks. In Proceedings of the Mediterranean ad hoc Networking Workshop (MedHocNet'04), 2004.

[19] A. Sardouk, Rana Rahim-Amoud, L. Merghem-Boulahia, and D. Gaïti. Power-aware agent-solution for information communication in wsn. Telecommunication Systems, pp. 329–338, 2011.

[20] A. Varga and R .Hornig. An overview of the OMNeT++ simulation environment. In Proceedings of the 1th IEEE International Conference on Simulation Tools and Techniques for Communications, Networks and Systems (SIMUTools), pp. 60:1–60:10, 2008.

[21] Y. Wen, T. A. F. Anderson, and D. M. W. Powers. On energy-efficient aggregation routing and scheduling in IEEE 802.15.4-based wireless sensor networks. Wireless Communications and Mobile Computing, 2012.

